Journal of Cases on Information Technology
Volume 26 • Issue 1

Social Recommender System Based on CNN Incorporating Tagging and Contextual Features

Muhammad Alrashidi, Universiti Teknologi Malaysia, Malaysia*

Ali Selamat, Universiti Teknologi Malaysia, Malaysia

https://orcid.org/0000-0001-9746-8459

Roliana Ibrahim, Universiti Teknologi Malaysia, Malaysia

Hamido Fujita, Universiti Teknologi Malaysia, Malaysia

ABSTRACT

The Internet's rapid growth has led to information overload, necessitating recommender systems for personalized suggestions. While content-based and collaborative filtering have been successful, data sparsity remains a challenge. To address this, this article presents a novel social recommender system based on convolutional neural networks (SRSCNN). This approach integrates deep learning and contextual information to overcome data sparsity. The SRSCNN model incorporates user and item factors obtained from a neural network architecture, utilizing features from item titles and tags through a CNN. The authors conducted extensive experiments with the MovieLens 10M dataset, demonstrating that the SRSCNN approach outperforms state-of-the-art baselines. This improvement is evident in both rating prediction and ranking accuracy across recommendation lists of varying lengths.

KEYWORDS

CNN, Deep Learning, Recommender Systems, Social Networks, Social Recommendation

1. INTRODUCTION

The rapid growth of the Internet has not only contributed to making life easier but has also generated an enormous volume of information, leading to the issue of information overload. It has become challenging for individuals to select their desired items that align with their preferences from the vast array of choices available. To address this, recommender systems (RSs) play a crucial role in alleviating information overload and enhancing user experiences. RSs are intelligent tools and methodologies designed to propose items that are likely to capture the interest of a particular user. These recommendations frequently play a role in various decision-making processes, such as helping users decide which products to purchase, which songs to listen to, and which articles to read. The term

DOI: 10.4018/JCIT.335524

*Corresponding Author

"item" commonly refers to the suggestions provided by the system. The system typically presents personalized recommendations in the form of ranked lists. These recommendations are generated by predicting the most suitable items or services for an individual based on their preferences (Ricci et al., 2022).

Three main types of RSs are commonly employed: content-based filtering, collaborative filtering (CF), and hybrid approaches (Da'u & Salim, 2020). These methods rely on the fundamental principle that individuals with similar past preferences are likely to make comparable future choices. CF recommenders specifically utilize data about user-item interactions to formulate their suggestions. Additionally, there exists a subset of RSs that incorporate contextual information related to users and/or items to enhance their recommendations. Among these types, content-based recommenders stand out as a distinct category within RSs. Moreover, certain hybrid RSs amalgamate insights from both user and item perspectives to provide comprehensive recommendations (Batmaz et al., 2019).

Despite the effectiveness of CF methods in various scenarios, the issue of data sparsity remains a significant hurdle (Da'u & Salim, 2020) (Batmaz et al., 2019). This challenge, stemming from an insufficient number of user ratings for each item, often hampers the performance of conventional approaches. Another notable concern is the cold start problem (Shokeen & Rana, 2020a), which significantly impacts both new users and new items. Providing meaningful recommendations without access to relevant historical data proves to be challenging. While novel users can receive suggestions for popular items, this approach falls short of delivering truly personalized recommendations.

Addressing these challenges necessitates the incorporation of supplementary data such as user profiles, product descriptions, and information from social platforms (Sun et al., 2019). Several research projects have explored the collection of additional data, and one approach for tackling data scarcity involves extracting insights from review texts (L. Zheng et al., 2017). In instances where adequate historical user-item interaction data is lacking, leveraging contextual information becomes pivotal in making informed inferences about user preferences. Individuals' preferences can be aggregated through social networks, allowing for the deduction of their tastes. Further data sources, like item tags or categories, can be directly employed to comprehend user preferences. For instance, a user's favorite movie genres or music album categories can shed light on the types of content they prefer (Sun et al., 2019). Contextual factors, such as timing, location, or tags, possess the capacity to influence an individual's perception of a particular item (Chen & Li, 2019). Tag-related data, on the other hand, supplements existing datasets by elucidating item attributes and capturing user preferences through tagging behaviors (J. Zhang et al., 2019b). Consequently, integrating supplementary information into the recommendation process emerges as a viable solution to tackle the challenges posed by data sparsity and the cold start problem in RSs.

While incorporating supplementary data often enhances the accuracy of RSs, it is important to note that this additional data is frequently more sparse compared to the user-item rating matrix (Ahmadian et al., 2022). In contrast, matrix representations connecting users, items, and tagged data possess high dimensions due to the typical abundance of users, products, and tags in real-world scenarios. Consequently, adding this extra information to the recommendation process can make techniques a lot more complicated, which could make them less useful for large-scale systems (Z. Zhang et al., 2017).

Deep learning techniques have demonstrated effectiveness in RSs by leveraging textual attributes like reviews and item descriptions within deep learning models (L. Zheng et al., 2017) (Yin et al., 2017) (Tuan & Phuong, 2017) (Catherine & Cohen, 2017) (Yao et al., 2017). However, challenges persist in terms of the sparsity issue within the user-item rating matrix and handling the additional data, which could potentially render the learned latent factors ineffective. Moreover, the efficacy of extracting valuable features from diverse data sources critically influences the success of recommendation techniques in terms of enhancing prediction accuracy. In light of these considerations, a potential solution to the data sparsity problem in recommendations involves the strategic utilization of additional data during the recommendation process. Directly harnessing supplementary information, such as social tagging or item descriptions, can discern a user's preferences and interests. This approach holds promise for mitigating the sparsity challenge and augmenting the quality of recommendations.

To address the problems mentioned earlier, this research proposes a new and efficient RS based on deep learning, aiming to tackle the issue of data sparsity in RSs. This study presents a RS that integrates a neural network and a convolutional neural network (CNN). CF, which exhibits excellent performance in other RSs, falls short in resolving nonlinear problems. However, deep learning enables CF to classify problems previously deemed unclassifiable. Furthermore, combining a CNN with a neural network captures contextual features in addition to user and item features, enhancing the accuracy of predicted ratings and extracting supplementary contextual information. We will employ the MovieLens 10M dataset to evaluate the accuracy of performance assessments. To facilitate comparison, we consider the other systems discussed earlier.

2. RELATED WORKS

2.1 Social Tagging Recommender Systems

We are currently living in an era characterized by the prevalence of personalization. Businesses are capitalizing on the fact that users desire customized content. One of the crucial tools for meeting this need is the recommendation engine, often developed using deep learning methods. While a recommendation engine can be highly useful, its creation is not without challenges. In recent years, social networks have provided access to various sources of information that were previously unavailable. Through social connections, ratings, comments, and tags, social networking generates a significant amount of data. The emergence of social networks enables us to indirectly infer a user's preferences by collecting the preferences of their friends. Additional information, such as product tags or categories, can be directly utilized to ascertain a user's preferences. For example, the categories of films or music albums can indicate the types of films and music they prefer (Guy, 2022). Leveraging this accessible social data in recommender systems can enhance prediction accuracy.

Numerous techniques, scenarios, and methods for social recommender systems (SRSs) have been proposed and employed for recommending items in the realm of publications (Shokeen & Rana, 2020b) (Alrashidi et al., 2023). According to research findings, tags, which are social media annotations applied to diverse types of content, including websites, images, and individuals, have shown significant efficacy as a foundation for recommendations. The ability of tags to succinctly summarize user perspectives across vast content areas renders them an immensely valuable tool for generating recommendations (Guy, 2015). According to (Shokeen & Rana, 2020a), tags are generally recommended from two perspectives: user-driven techniques and document-driven techniques. User-driven techniques strive to identify similar individuals or related groups and suggest titles based on their past tagging activities. Conversely, document-driven techniques categorize documents into various groups through document-level analysis. Two primary reasons underlie the relative inefficacy of user-centric methods in recommending categories compared to other approaches. Firstly, only a small number of individuals engage in comprehensive tagging, and secondly, tags exhibit limited reusability due to the ongoing expansion of tag terminology. Consequently, user-generated tags often tend to be vague and disorganized (J. Zhang et al., 2019a). Furthermore, the most significant hurdle for these endeavors is the data sparsity inherent in social media platforms, wherein each user maintains only a limited number of connections (Pan et al., 2020).

The authors of (Gao et al., 2021) employ a deep neural network to generate distinctive characteristics by leveraging search tags and user search history, thereby constructing a hybrid RS for microblogging content recommendations. The study authors collected datasets from Sina Weibo and Twitter, encompassing microblogs and social networks. The study authors utilized a deep neural network to suggest diverse microblogs and proposed two microblog recommendation techniques. The first technique utilizes association rules to expand user preference tags from microblog content. The second approach employs microblog subject connections to infer user preferences. Another study (J. Zhang et al., 2019b) introduces a user preference tree with deep characteristics and tag trees for suggesting social photos. Researchers rate tags and construct a social image tag tree to utilize effective tag information, organizing it by evaluating tag connections. The re-ranked tag lists then reflect image content. Deep feature extraction is introduced through AlexNet training. Social photos and tag-related user preferences are combined within a user preference tree and tag tree structure. This construction provides insight into user preferences, ultimately leading to the development of a personalized social image recommender system. For visualizing the relationship between users' tags and items, (Chen & Li, 2019) proposes a tensor factorization approach. Contextualization between customers and items is established through tensor factorization. Adversarial tensor factorization is employed to offer context-based suggestions. The SocialCDL model (Chen & Li, 2019) is tested using real-life scenario scores and social data, including ratings, comments, search behaviors, website visits, and user interactions with liked and disliked items, all contributing to the categorization of social recommendation information sources. Research conducted by (Lei et al., 2016) introduces a pair-wise learning approach (CDL-Image) tailored for picture recommendations. This approach employs three sub-networks. Two identical sub-networks leverage CNNs to generate representations of favorable and unfavorable pictures for each user. Simultaneously, the third sub-network focuses on learning user taste representations through four fully connected layers. Input user vectors of the network comprise related tag vectors generated by Word2Vec. However, considering the efficiency of item embedding techniques for recommendation projects, our study employs a deep learning approach to offer recommendations using social tagging as contextual features through feature extraction. This approach addresses the data sparsity issue in RSs and enhances recommendation performance.

2.2 Text-Feature Deep Learning Methods for Recommendation

In addition to the matrix structures that include user-item ratings, valuable reviews often provide insights into the reasons behind users' ratings. They outline the item attributes that held the most importance for them and convey their sentiments toward the item. As stated in (Sun et al., 2019), prior research has utilized textual attributes such as reviews and comments through traditional methodologies. This has been done either by measuring similarities among individuals based on the textual review's similarity rather than the ratings, or by employing sentiment analysis derived from user comments. However, it is important to recognize that not all user feedback or product comments carry equal significance in portraying user preferences or item features. Moreover, not all phrases within a review carry the same weight in conveying the review's essence. To effectively address these issues, deep learning algorithms integrate attention mechanisms. The field of deep learning-based recommendation, which incorporates side information, is an ongoing area of research. Due to their inherent flexibility, deep learning algorithms can flexibly incorporate diverse social information. They have shown impressive effectiveness in handling complex structured data like knowledge graphs, as well as unstructured data such as text and image attributes (Sun et al., 2019). Therefore, the primary focus lies in extracting valuable insights embedded within textual features, encompassing reviews, advice, feedback, content, and descriptions. This focus aims to significantly enhance the accuracy of recommendations by capturing nuanced information.

Prior studies have demonstrated the adoption of deep belief networks (DBNs) primarily for feature extraction and classification purposes, particularly in the context of text and audio content (Batmaz et al., 2019). For instance, (Yifei Zhao et al., 2015) employed DBNs in text-based RSs. Recurrent neural networks (RNNs) are specialized neural networks designed to handle sequential data. (Ko et al., 2016) used RNNs to represent temporal and contextual features of user behaviors, aiming to enhance recommendation accuracy by integrating these representations with latent variables of user preferences. In another study, (C.-Y. Wu et al., 2017) introduced an RNN-based RS that considers both reviews and ratings. This method combines static user and item representations with dynamic user and film representations through two long short-term memory (LSTM) units for the rating process. LSTM is utilized to represent review text, combining input character embeddings with user and item representations. Autoencoders (AEs) perform data compression through an encoder–decoder architecture. (H. Wang et al., 2015) employed a stacked denoising AE to extract textual representations from film and book descriptions. A CNN includes convolution in one or more layers and is often used for image recognition and object categorization. In (X. Wang et al., 2017), a CNN-based RS recommended articles for experienced editors, considering textual content and additional information. The authors proposed a dynamic attention-based deep model to address challenges posed by non-explicit selection criteria and non-stationary data. The DeepCoNN model (L. Zheng et al., 2017) simultaneously acquires user and product representations through textual feedback. Parallel networks for users and items process word embeddings of reviews to capture semantic information. CNNs extract textual features in subsequent layers, contributing to user and product representation acquisition. Another approach (Tuan & Phuong, 2017) integrates session clicks and content features via a three-dimensional CNN, utilizing character-level encoding. Traditional topic modeling techniques are detailed but struggle with complex relationships within text. Deep learning algorithms incorporate text characteristics by employing word-level embeddings as input for advanced techniques like CNNs, AEs, or RNNs. This approach transforms text features nonlinearly, making it suitable for variable-length sequences. CNNs can process such sequences, making them effective for analyzing item titles and tags. This enables RSs to grasp intricate patterns and semantic information that traditional collaborative filtering or content-based RSs might overlook.

In addition, the researchers (Y. Zheng et al., 2016) first describe the fundamental CF-NADE model, then proposing enhancements by means of parameter sharing across several ratings. In addition, they propose a factored version of CF-NADE to enhance scalability. The research further examines the ordinal nature of preferences and suggests an ordinal cost to optimize CF-NADE, which exhibits greater performance. A study by (Sedhain et al., 2015) suggest using CF, which is mainly used in recommender systems with autoencoders, a kind of artificial neural network. In addition, they provide data from the Movielens and Netflix datasets, demonstrating that AutoRec is superior to popular CF methods like biased matrix factorization, RBM-CF, and LLORMA. A research by (C C & Mohan, 2019) proposed a hybrid approach that involves a joint optimization function, extending the AE technique to integrate social information. The authors (Strub et al., 2016) give a great work concentrating on the incorporation of autoencoders in recommendation systems. The purpose of this study is to combine best practices from the literature in order to develop a recommender system utilizing autoencoders, a technology widely used in deep learning. A work by (Berg et al., 2017) takes a new approach to matrix completion for recommender systems, shifting away from traditional methods and approaching it as a link prediction issue on graphs. It shows a graph auto-encoder system that works with differentiable message forwarding on a two-way user-item engagement network. The authors named the proposed technique Graph Convolutional Matrix Completion (GC-MC). A study by (X. Wu et al., 2019) illustrates a new method called the NTF model. It uses the long-short-term memory (LSTM) architecture to describe how different latent factors interact with relational data over time in multiple dimensions. This method works especially well when the data is changing over time and has more than one variable.

However, although these robust approaches have proven efficacy in diverse scenarios, our suggested methodology sets itself apart through the successful integration of contextual information and deep learning, the overcoming of the data sparsity obstacle that frequently hinders the performance of other systems, and the production of accurate and relevant recommendations across multiple contexts. Our suggested methodology, as compared to other methods that mainly rely on user-item interaction data, utilizes supplementary information from item titles and tags via a convolutional neural network (CNN), which improves the process of making recommendations.

3. THE PROPOSED METHOD

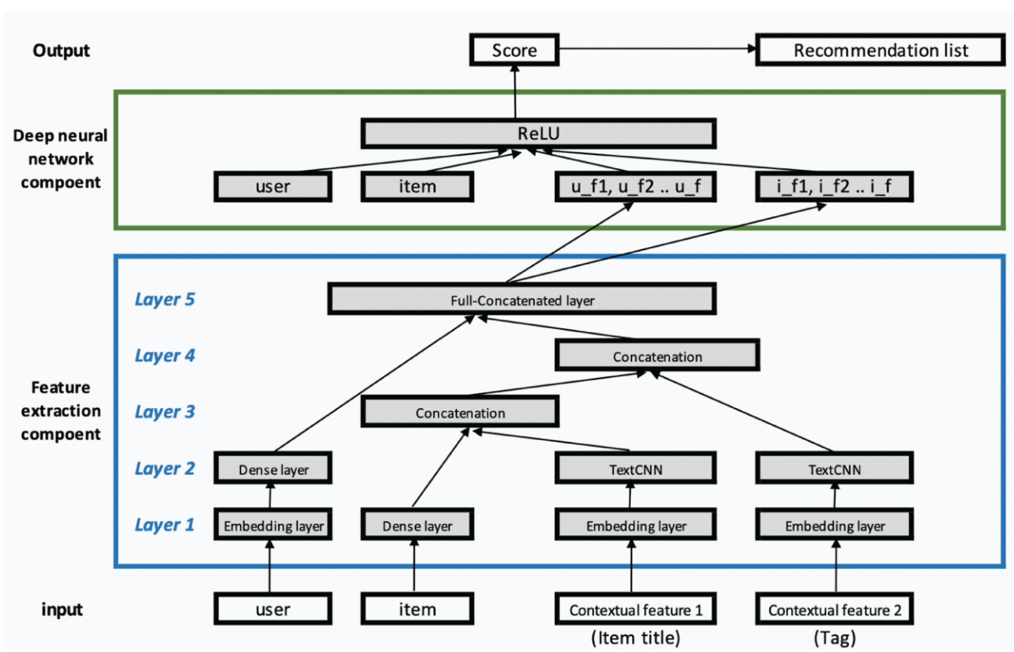
In this study, we propose a novel RS that combines deep learning techniques, specifically CNNs for text data, with a neural network to address data sparsity issues in RSs. Figure 1 illustrates the proposed method, SRSCNN, which is a social recommender system based on convolutional neural networks.

3.1 Data Representation

Incorporate supplementary data, including item descriptions, user profiles, and social network information, to improve the process of generating recommendations. The inclusion of this supplementary information can enhance the context and make better recommendations in situations where the user-item interaction data is sparse. In this proposed method, we leverage additional information from the dataset, encompassing movie titles, tags, ratings, and category details. This integrated dataset provides a richer input for the deep learning model to generate recommendations. This section delineates the data preprocessing techniques undertaken to refine the dataset for our RS, which employs deep learning methods. These techniques are crucial for ensuring that the data is appropriately formatted for both training and evaluating the model. The integration of data involves consolidating rating, item, and tag information using the item ID feature. Initial processing entails tokenizing the item titles. Fitting the tokenizer to movie titles involves merging ratings, items, and tags. To achieve uniformity in the length of tokenized item titles, we apply sequence padding with zeros. This step is crucial to ensure the resulting sequences can be effectively used as input for the TextCNN model.

During the data preprocessing phase, item titles undergo tokenization and sequence padding to establish a consistent length. Then, the preprocessed data is prepared as input for the TextCNN model. Subsequent stages encompass the development of deep learning models, their training on the preprocessed data, and the evaluation of their efficacy within the context of the recommendation task.

Figure 1. The proposed method (social recommender system based on convolutional neural networks)



3.2 Deep Learning for Feature Extraction

In recommendation systems, using deep learning for feature extraction may help with the problem of data sparsity by making it possible to find meaningful ways to represent sparse data. Deep learning models have the capability to identify complex patterns and derive features at a higher level of abstraction that may avoid traditional approaches. The CNN is a type of feedforward neural network that utilizes convolutional layers and other supplementary layers to extract features. Three fundamental elements facilitate the transformation of the input volume into an output volume in a standard CNN. These components consist of convolutional layers, pooling layers, and fully connected layers. CNNs are adept at capturing intricate and meaningful features from input data. This capability is particularly valuable in text data, such as movie titles and tags, where semantic and contextual information plays a crucial role. Extracted features from such data can be highly informative for generating recommendations. Hence, our recommended approach for the RS involves the utilization of a combination of TextCNN models and dense layers to analyze and consider item titles, tags, genres, and user data. These architectural designs extract distinct features from diverse data sources, merging them to create feature vectors for both users and items. This merging process enhances the quality of recommendations. Therefore, Convolutional neural networks (CNNs) are examples of deep learning models that have the capability to autonomously acquire significant features and representations from unprocessed data. This enables the model to identify the latent patterns and structure within the sparse data. In Addition, by using deep learning-based feature extraction, recommendation systems can overcome the challenges posed by data sparsity and generate more accurate and personalized recommendations.

3.2.1 TextCNN for Item Title Processing

CNNs can understand the meanings of words contained in item titles and subtitles by considering the adjacency of those words. This approach offers the advantage of capturing greater details, which improves the quality of recommendations. We use a TextCNN model tailored to capture textual patterns and semantic insights for processing item titles. The model encompasses an embedding layer with an input dimension equal to the vocabulary size and an output dimension of 25. This layer transforms tokenized item titles into dense vectors. After the embedding layer, we add a one-dimensional convolutional layer with 32 filters, a kernel size of 5, and an activation function called rectified linear unit (ReLU). This layer's role is to detect localized features or patterns within the embedded item titles. Subsequently, we implement a global max-pooling layer to reduce spatial dimensions while preserving crucial information. A dense layer with 16 units and a ReLU activation function is connected to the output of the pooling layer, serving as a high-level feature extractor for item titles.

3.2.2 TextCNN for Tag Processing

We employ a similar TextCNN architecture to handle the tags associated with each item. The architectural design bears resemblance to that of the item-named TextCNN, with a modification in the embedding dimension, set at 30. This model processes tokenized and padded tag data, extracting pertinent attributes that contribute to the eventual item representation.

3.2.3 Item Input Layer for Genre Information

CNNs are capable of learning independently a suitable representation of data in a lower-dimensional space, which can be advantageous when processing high-dimensional data. CNNs possess the ability to learn accurate representations of highly-dimensional data in a reduced-dimensional space. This capability proves advantageous for RSs, which frequently encounter high-dimensional data. Therefore, to capture genre-based information, we introduce a dense layer with 16 units and a ReLU activation function. This layer processes the item ID along with the corresponding 19 genres, acquiring meaningful representations that enhance the recommendation process.

The concatenated feature vector results from combining the TextCNN model's output for item titles and the input layer for items. This vector then passes through a dense layer comprising 16 units and a ReLU activation function. The purpose of this layer is to obtain a comprehensive item representation that encompasses both textual and genre-based data.

3.2.4 Inclusion of Tag Features

The TextCNN tag outputs are concatenated with the existing item feature vector. This merged vector then progresses through a dense layer containing 16 units and a ReLU activation function. This process aids the model in achieving an integrated item representation that amalgamates title, genre, and tag information.

3.2.5 User Network for Processing User Information

A user network is established to process user data. The model's architecture comprises an embedding layer with an output dimension of 22, followed by a flatten layer, a dense layer featuring 128 units and a ReLU activation function, and another dense layer with 16 units and a ReLU activation function.

3.2.6 Fusion of User and Item Feature Vectors

Concatenating the feature vectors from the user network and the item network forms a combined user-item feature vector. This vector then traverses a dense layer with eight units and a ReLU activation, further refining the connections between user and item features.

3.2.7 Output Layer

A final dense layer, housing a single output unit, connects to the preceding layer. This layer predicts the rating for a given user–item pair, forming the basis for generating recommendations.

3.2.8 Feature Extraction and Recommendation Generation

The Adam optimizer is used to compile the model. The trained model extracts user and item features, subsequently adding them to a new data frame. This structure, in conjunction with the learned feature vectors, enables recommendations based on user preferences and item features, ultimately enhancing the overall effectiveness of the RS.

In conclusion, CNNs excel at feature extraction from textual data. Our approach involves using a CNN-based model (TextCNN) to extract features from item titles and tags. This capability empowers the model to comprehend intricate patterns and semantic nuances within the text, aspects that traditional collaborative filtering or content-based RSs might overlook.

3.3 Deep Learning for Recommendation

The trained model leverages the extraction of both user and item features, as previously mentioned. The integration of these enhanced features holds the potential to heighten prediction accuracy, thereby leading to more precise recommendations. Incorporating these aforementioned features into a new data frame facilitates the generation of recommendations based on the acquired feature vectors.

Before proceeding with model training, the dataset undergoes a preprocessing phase, during which it is divided into distinct subsets designated for training and testing purposes.

Our proposed model adopts a feedforward neural network architecture. The input layer's dimensionality is determined by the number of item features present in the dataset. Subsequent dense layers consist of 32 and 8 neurons, respectively, featuring ReLU activation functions to introduce non-linearity. The output layer consists of a single neuron tasked with predicting the rating of a user–item pair, serving as a measure of user preference.

The model is compiled using the widely used Adam optimizer, an optimization algorithm grounded in first-order gradients. The choice of the Adam optimizer stems from its capacity to dynamically adjust the learning rate during training. Particularly suited for deep learning tasks, it often yields faster convergence and superior performance. After compilation, train the model using the training dataset, with a batch size of 32 and a total of 100 epochs. During this training regimen, the model reserves 20% of the data for validation purposes, which helps monitor model effectiveness and prevent overfitting.

Post-training, we visualize the model's performance by plotting training loss and validation loss against the number of epochs. This visualization aids in detecting potential overfitting or underfitting issues. Subsequently, the test data is employed for performance evaluation, producing a root mean square error (RMSE) value that signifies the model's prediction accuracy on unseen data. In addition to the loss function, a custom evaluation metric, RMSE, is defined to monitor model performance during training and on the test set.

We implement a customized recommendation function that accepts a user ID and the desired number of recommended items. This function initially identifies items the user has not yet encountered and extracts corresponding item features. The user features are then merged with the item features, and the neural network model predicts ratings for each user–item pair. Subsequently, the recommendation function returns the top-N items with the highest predicted ratings as recommendations. To illustrate the effectiveness of our recommendation function, a specific user undergoes the recommendation process, generating a list of top-N recommended items. Next, we present the item details, offering insights into personalized item recommendations facilitated by our proposed method.

In conclusion, to address the data sparsity challenge in RSs, we introduce a novel deep learning recommendation model that combines neural networks and CNNs to leverage additional data such as item titles and tags. Our proposed deep learning-based recommendation approach offers an avenue to produce highly accurate and personalized movie recommendations. Our goal is to enhance the overall quality of recommendations and contribute to the ongoing advancement in the field of RSs through the utilization of user and item features.

4. EXPERIMENTS

4.1 Dataset

We employ the MovieLens 10M dataset, a well-known dataset extensively utilized in research on CF and RSs. The dataset comprises 10,000,054 ratings (integer values ranging from 1 to 5) and 95,580 tags, which have been assigned to 10,681 distinct movies by 71,567 unique users. Additional information, such as movie titles, genres, and release years, is provided for each movie. The genres encompass a variety of 19 categories, including Action, Adventure, Animation, Children, Comedy, Crime, Documentary, Drama, Fantasy, Film-Noir, Horror, Musical, Mystery, Romance, Sci-Fi, Thriller, IMAX, War, and Western. Notably, the dataset lacks comprehensive demographic data about users, such as gender, age, or occupation. It only includes anonymized user IDs to ensure privacy. The MovieLens 10M dataset holds significant prominence in research circles, serving as a cornerstone for the development and evaluation of recommendation algorithms, including matrix factorization, CF, and deep learning-based methodologies. Table 1 provides a summary of the MovieLens 10M dataset.

In essence, this study converted the data from the MovieLens 10M dataset into numerical values suitable for training the model. They gathered essential information, including ratings, movie genres, titles, and tags, and merged them to create the foundation for the proposed deep learning

Table 1. Summary of MovieLens 10M dataset

Attribute	Ratings	Users	Movies	Genres	Tag	Sparsity
Value	10,000,054	71,567	10,681	19	95,580	98.36%

recommendation system. This process of data transformation lays the groundwork for the subsequent phase of model development.

4.2 Experimental Setup

In this study, we introduce a neural network and CNN-based RS to tackle the issue of data sparsity in RSs. The proposed model leverages both explicit feedback (ratings) and implicit feedback (movie titles and tags) to acquire robust user and item representations. In essence, the system takes user ID, movie ID, movie title, and tags as inputs. The system generates item features by subjecting the movie title and tags to CNN layers and processing the movie ID and genre details through dense layers. Then, we combine these two sets of item features. The embedding layer and dense layers process the user ID on the user side to generate user features. The model merges the user and item features to create a comprehensive user-item interaction representation. Supplementary dense layers further process this interaction representation to predict ratings.

During data preprocessing, we initially tokenize and pad the movie titles to establish a fixed input length. This is followed by an embedding layer, a one-dimensional convolutional layer with ReLU activation, global max-pooling, and a subsequent dense layer with ReLU activation. This sub-network generates feature representations for movie titles. Likewise, the tags linked with the movies are tokenized and padded. The same architecture as the movie title CNN is employed to construct feature representations for the tags. The movie input layer incorporates the movie ID and 19 genre features, which are then directed through a dense layer with ReLU activation to generate movie features. The outcomes of the movie title CNN, movie tag CNN, and movie input layer are concatenated to yield an aggregated feature representation for each movie. On the user side, the user network integrates an embedding layer, followed by a flatten layer, and two dense layers with ReLU activations. This sub-network creates feature representations for users. By merging the user and movie features, a unified feature representation is produced for each user–movie pair. Ultimately, a dense layer with a single output neuron is employed to predict the rating for each user–movie pair. The flowchart in Figure 2 visually represents the above-described model architecture, highlighting the CNN feature extraction process.

We train the proposed model using the Adam optimizer. Once the model training is complete, we extract the user and movie features generated by the model and compile a new dataset for the recommendation process.

To effectively tackle the data sparsity challenge, the acquired user and item features from the model are extracted and integrated as supplementary information into the original dataset. Subsequently, a recommendation model is trained on this enriched dataset to facilitate the provision of recommendations. We employ a straightforward feedforward neural network for training on this dataset to ultimately make precise rating predictions. Upon the completion of model training and evaluation, the developed system becomes capable of generating personalized recommendations for users by anticipating their ratings for different movies. To curate these recommendations, the projected ratings can be arranged in descending order, thereby identifying the top-N suggested items for each individual user.

4.3 Evaluation Metrics

The model's performance is assessed using RMSE as the evaluation metric. RMSE is a frequently employed metric in the field of recommender systems to indicate accuracy (Batmaz et al., 2019). RMSE provides a useful baseline for developing and evaluating models. Utilizing RMSE facilitates more convenient comparison and benchmarking with different approaches and papers. This metric provides a comprehensible gauge of the average error in the same units as the original values. This characteristic aids in understanding the scale of errors, enabling a more meaningful evaluation of the model's performance (Da'u & Salim, 2020). Notably, RMSE accentuates substantial errors. This attribute is crucial for gauging the model's proficiency in handling outliers or instances where

Figure 2. Flowchart of convolutional neural network model for feature extraction

predictions significantly deviate from actual values. Consequently, the selection of RMSE as the evaluation metric for the neural network recommendation model is driven by its capacity to furnish a comprehensive and interpretable assessment of performance. Moreover, RMSE establishes a quantifiable measure for average error, thereby enhancing the evaluation and discussion of the model's effectiveness. The differentiability of RMSE enables its compatibility with gradient-based optimization methods often used in deep learning systems. This is crucial because it enables the use of very effective optimization methods for training the model. RMSE can serve as a reference throughout the process of model development. To illustrate, in the case of employing a deep learning model comprising numerous layers and neurons, it is possible to optimize the RMSE by modifying these parameters.

Normalized discounted cumulative gain (NDCG) is an evaluation metric employed to gauge the quality of rankings or the relevance of items listed at the top. The underlying principle of NDCG prioritizes more pertinent items over irrelevant ones during the ranking process. According to the NDCG, extremely significant items provide greater satisfaction than low rated items (Da'u & Salim, 2020). Unlike RMSE, NDCG places a strong focus on item rating. This is crucial for systems that provide recommendations since the sequence of items suggested may have a major impact on the user experience. Items that are highly relevant are more valuable if they show at the top of the suggestion list. Therefore, the NDCG considers the importance of each suggestion, not merely its position. This implies that even if the absolute locations of the suggestions are the same, a system that suggests more relevant items will obtain a higher NDCG.

The selection of these approaches allows for a thorough comparison since they each approach the suggestion issue from a different perspective. Comparing our SRSCNN to these proven methods allows for a robust evaluation of its performance. It specifically serves to illustrate if the combination of deep learning and contextual information in SRSCNN can overcome the limits of these approaches, particularly the problem of data sparsity, and deliver more accurate and relevant recommendations.

4.4 Comparative Methods

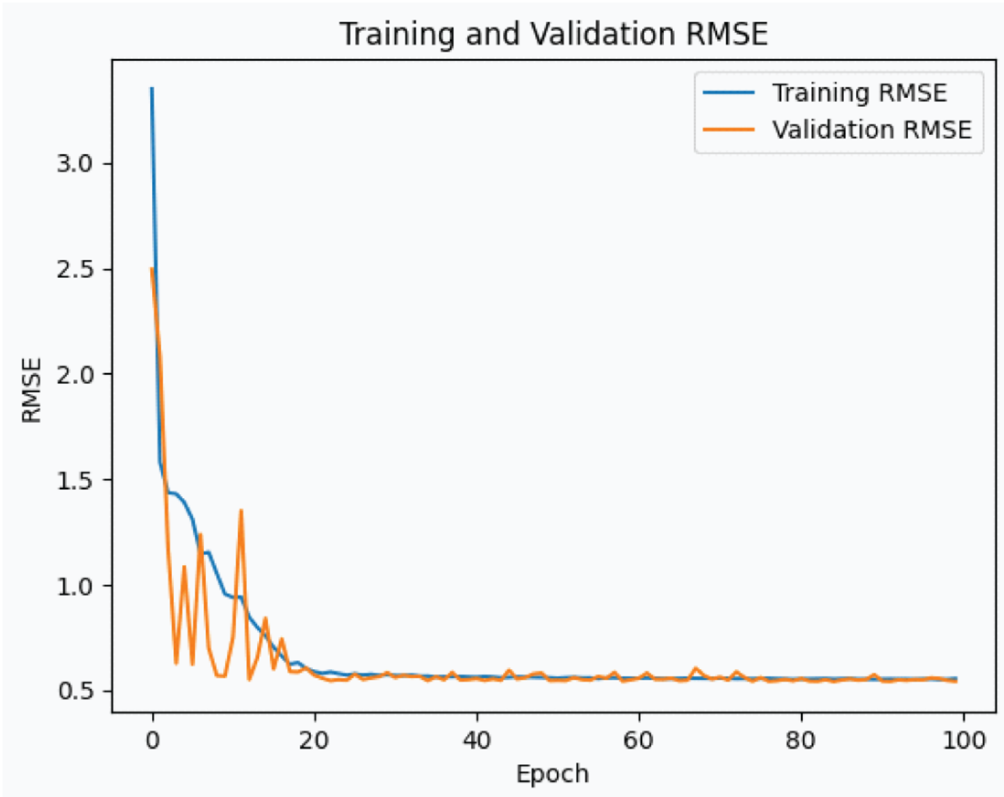
In order to evaluate the efficacy of our proposed approach (SRSCNN), we undertake a comprehensive comparative assessment against the following methods, aiming to gauge the ultimate performance outcomes. The ensuing descriptions provide a concise overview of the recommendation techniques utilized for comparison with our proposed method:

- **User-based collaborative filtering neural autoregressive density estimator (U-CF-NADE) (Y. Zheng et al., 2016):** An autoregressive model featuring a feedforward architecture, tailored explicitly for CF recommendations.
- **Item-based autoencoder for collaborative filtering (I-AutoRec) (Sedhain et al., 2015):** An AE-based model designed for collaborative RSs.
- **Adversarial embedding-based sparse reconstruction (AESR) (C C & Mohan, 2019):** A hybrid approach that involves a joint optimization function, extending the AE technique to integrate social information.
- **Item-based collaborative filtering neural (I-CFN) (Strub et al., 2016):** An approach exploring the relationships between AEs and matrix factorization in the context of matrix completion. This methodology eliminates the need for separate systems by expanding neural network applications to CF embedding construction.
- **Graph convolutional matrix completion (GC-MC)(Berg et al., 2017):** A specialized graph AE framework developed for the matrix completion tasks in RSs. It utilizes a graph convolution layer to generate user and item embeddings through message propagation on a bipartite user–item interaction graph.
- **Non-negative tensor factorization (NTF) (X. Wu et al., 2019):** A neural network-based NTF method tailored for modeling temporal interaction data. It addresses the critical issue of evolving user-item relationship data.
- **MRMA (Li et al., 2017):** This approach characterizes user-item ratings by amalgamating various ranks of low-rank matrix estimation techniques.

5. RESULTS AND DISCUSSION

We previously introduced an innovative recommendation model founded on deep learning principles to tackle the challenges posed by data sparsity in RSs. This model harnesses the power of neural networks and CNNs to effectively incorporate supplementary information, such as movie titles and tags. This study proposes a recommendation approach rooted in deep learning techniques that can generate highly precise and tailored movie recommendations based on individual user preferences. Our study aims to elevate the overall caliber of recommendations and contribute to the ongoing advancements in the realm of RSs by leveraging user and item features. Figure 3 shows how the training and validation RMSE changes over time. This gives us a clear picture of the average error in the same units as the original data.

Figure 3. Training and validation root mean square error (RMSE)



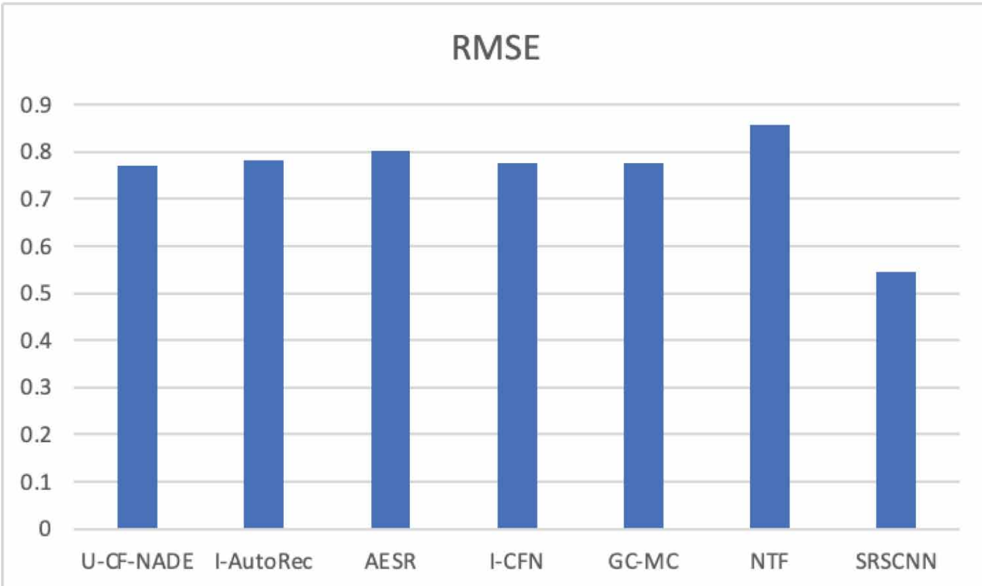
5.1 Rating Prediction Comparison

In this study, we conducted a comparative analysis of our proposed method (SRSCNN) against several established methods using the RMSE metric. The methods evaluated in this comparison include U-CF-NADE, I-AutoRec, AESR, I-CFN, GC-MC, and NTF. As previously mentioned, we employed the MovieLens 10M dataset for our evaluations. Table 2 presents the outcomes and performance of these methods, measured by RMSE, providing a comparison against baselines within the MovieLens 10M dataset.

Our proposed method exhibits superior performance in contrast to all other methods, achieving the lowest RMSE value of 0.5439. These results substantiate the effectiveness of our approach in generating accurate predictions when compared to alternative methods. Our deep neural network model's innovative architecture and learning strategies contribute to the observed reduction in RMSE compared to other approaches. It is vital to acknowledge a notable performance gap between our proposed method (SRSCNN) and the second-best contender, U-CF-NADE. This gap accentuates the superiority of our approach. In addition, we can quantify the performance gains by expressing them as the percentage reduction in RMSE compared to the subsequent most effective approach, which is U-CF-NADE in this case. Our findings illustrate that our proposed method effectively reduces RMSE by approximately 29.45% compared to the second most successful approach, U-CF-NADE. Figure 4 visually illustrates the performance of each of these methods.

Our proposed method (SRSCNN) exhibits remarkable performance, which carries significant implications for the field of RSs. Firstly, it suggests that deep learning-based models possess the capacity to surpass traditional CF and matrix factorization techniques in specific recommendation

Figure 4. Performance comparison of each method based on RMSE



tasks. This finding aligns with current scholarly research that highlights the benefits of integrating deep learning techniques across various domains. Furthermore, our findings have the potential to stimulate further exploration in the realm of designing sophisticated deep learning architectures and customized learning algorithms tailored explicitly for RSs. Lastly, our proposed approach holds promise for diverse recommendation tasks, encompassing content-based or hybrid RSs, allowing for an assessment of its applicability and performance in varying scenarios.

In summary, our suggested RS works effectively in terms of RMSE, beating out other well-known methods by a large amount. It does this by using a deep neural network approach. The finding contributes to the increasing body of academic research that emphasizes the potential of deep learning methods in the field of RSs and establishes a solid foundation for further investigation in this domain.

5.2 Item Ranking Comparison

Our proposed method (SRSCNN) was evaluated and compared with other established methods using NDCG at different list lengths (top_N = 1, 5, 10, 20). In this study, we initially examine the impact

Table 2. RMSE of different models using MovieLens 10M

Method	RMSE
U-CF-NADE	0.771
I-AutoRec	0.782
AESR	0.802
I-CFN	0.775
GC-MC	0.777
NTF	0.856
SRSCNN	**0.5439**

of recommendation ranking on the evaluation of NDCG for our proposed method. Subsequently, we present the findings obtained from this investigation. The approach used to calculate the NDCG scores for a particular user yields a result when applied to the NDCG score of the indicated recommendations. Figure 5 displays the NDCG performance for recommendations with a specific number of top_N items. Additionally, we calculated NDCG scores for different values for a specific user. Figure 6 illustrates the relationship between the NDCG evaluation score and the number of recommended items.

We further extend the evaluation of our proposed SRSCNN to include multiple users. We calculate the mean NDCG score and its standard error. These measures offer a more robust understanding of the system's performance across various user profiles. The performance of the SRSCNN method outperformed the other seven methods significantly across all NDCG list lengths. Table 3 presents the results for different lengths of recommendation lists based on the MovieLens 10M dataset.

The NDCG results achieved by the SRSCNN method were significantly higher compared to other approaches. Specifically, the NDCG@1 score reached 0.9228, NDCG@5 achieved 0.9048, NDCG@10 stood at 0.9092, and NDCG@20 reached 0.9153. Among the competitors, MRMA demonstrated the closest performance. These results underscore a substantial and noteworthy improvement achieved by SRSCNN in comparison to alternative methods. Notably, at NDCG@5, the SRSCNN method achieved a score of 0.9048, while at NDCG@10, it achieved 0.9092, and at NDCG@20, a score of 0.9048 was attained. These scores consistently highlight the SRSCNN method's capability to recommend relevant items, even with extended recommendation lists.

The superior performance of the SRSCNN method can be attributed to its effective utilization of Convolutional Neural Networks (CNNs) for feature extraction. CNNs excel in extracting rich and meaningful features from input data, a particularly advantageous trait when dealing with text data like movie titles and tags. This advantage becomes evident in scenarios with sparse user-item interactions, where the extracted additional features contribute valuable input for generating accurate recommendations.

When semantic and contextual information from movie titles and tags are added to the SRSCNN method, it makes it possible to represent high-dimensional data more accurately in a space with

Figure 5. NDCG scores for a particular user

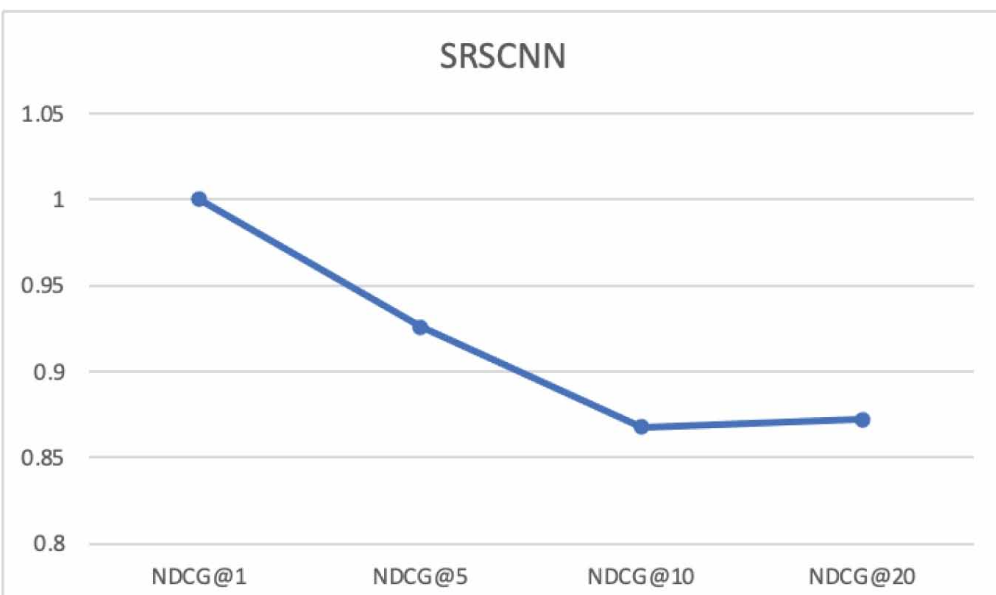


Figure 6. Normalized discounted cumulative gain (NDCG) evaluation score vs. number of recommended items

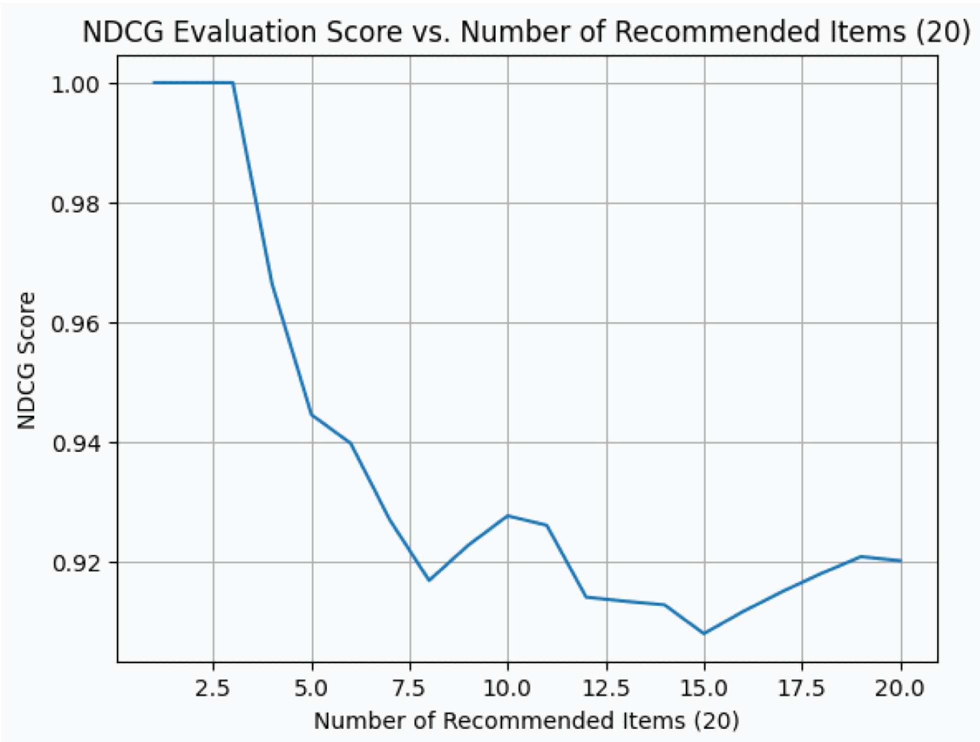


Table 3. NDCG results on MovieLens 10M dataset

Method	NDCG@1	NDCG@5	NDCG@10	NDCG@20
BPMF from (Li et al., 2017)	0.6563 ± 0.0005	0.6845 ± 0.0003	0.7467 ± 0.0007	0.8691 ± 0.0002
GSMF from (Li et al., 2017)	0.6708 ± 0.0012	0.6995 ± 0.0008	0.7566 ± 0.0017	0.8748 ± 0.0004
LLORMA from (Li et al., 2017)	0.6829 ± 0.0014	0.7066 ± 0.0005	0.7632 ± 0.0004	0.8782 ± 0.0012
WEMAREC from (Li et al., 2017)	0.7013 ± 0.0003	0.7176 ± 0.0006	0.7703 ± 0.0002	0.8824 ± 0.0006
MPMA from (Li et al., 2017)	0.6908 ± 0.0006	0.7133 ± 0.0002	0.7680 ± 0.0001	0.8808 ± 0.0004
SMA from (Li et al., 2017)	0.7002 ± 0.0006	0.7134 ± 0.0004	0.7679 ± 0.0003	0.8809 ± 0.0002
MRMA	0.7048 ± 0.0006	0.7219 ± 0.0001	0.7743 ± 0.0001	0.8846 ± 0.0001
SRSCNN	**0.9228 ± 0.0000**	**0.9048 ± 0.0045**	**0.9092 ± 0.0048**	**0.9153 ± 0.0028**

fewer dimensions. This proficiency holds substantial value in RSs, which frequently grapple with high-dimensional datasets. SRSCNN's performance not only demonstrates the promise of embedding deep learning techniques, like CNNs, into RS frameworks but also highlights the potential benefits of leveraging supplementary data, such as movie titles and tags, to significantly enhance recommendation performance. However, it is crucial to acknowledge that despite demonstrating superior performance based on NDCG, further research is needed to examine the effectiveness of the SRSCNN method across alternative metrics and its real-world applicability. Moreover, there is a need to assess the robustness and scalability of the SRSCNN approach when dealing with larger and more diverse datasets.

In summary, the results demonstrate that the proposed SRSCNN method, which combines neural networks and CNN within an RS, can achieve superior performance compared to other approaches. The ability of CNN to extract meaningful features from textual data helps mitigate data sparsity and provides valuable input for generating more effective recommendations.

6. CONCLUSION

The present study introduces a novel SRS that employs CNNs to integrate tagging and contextual features, with the goal of alleviating the issue of data sparsity in recommender systems. The study's findings indicate that the proposed SRSCNN method demonstrates notably superior performance compared to various current techniques. Specifically, in terms of rating prediction, the SRSCNN method achieved the lowest RMSE of 0.5439 on the MovieLens 10M dataset, surpassing other methods. This exceptional rating prediction underscores the model's capacity to effectively learn latent factors from sparse user-item data, capitalizing on user and item features, as well as semantic information from contextual data such as item titles and tags. Furthermore, the SRSCNN method secured the highest NDCG scores across different top-N recommendation list lengths, outperforming alternative methods. These impressive NDCG scores highlight the model's consistent capability to recommend the most relevant items to users. In essence, the proposed approach addresses crucial challenges in RSs, encompassing data sparsity issue, through the utilization of deep learning algorithms. The outcomes furnish robust evidence that the integration of neural networks and CNNs with textual data can significantly enhance recommendation accuracy. The favorable outcomes obtained from the evaluation suggest that the SRSCNN approach holds potential for application in other recommendation domains grappling with sparse data challenges. Furthermore, it underscores the capacity of deep learning techniques to advance the evolution of RSs.

The SRSCNN approach presents a potential strategy to tackle significant obstacles in RSs, such as data sparsity problems, by using deep learning methods. Nevertheless, it is not without limitations. The model's dependence on the accuracy of tagging and contextual data offers a potential limitation. The SRSCNN model demonstrates satisfactory performance on the MovieLens 10M dataset, but its potential to handle much larger datasets, particularly in terms of scalability, has not been thoroughly examined. The computational complexity of deep learning models may be an obstacle, and the model's performance could decrease as the dataset size increases. The temporal dynamics of user preferences and item popularity may undergo changes over time; however, the existing SRSCNN model fails to account for this in its recommendation process. This may potentially constrain the efficacy of the model in situations where time-dependent variables exert a substantial influence.

Further research is suggested to evaluate the SRSCNN model's performance on a variety of datasets, including those deriving from irrelevant domains or possessing characteristics like sparser user-item interactions or larger dimensions. Subsequent studies could delve into exploring various model architectures, optimizing hyperparameters, and incorporating additional features to enhance the model's performance. Furthermore, potential enhancements could involve experimenting with deeper networks, fine-tuning hyperparameters, and utilizing more intricate loss functions tailored to recommendation tasks.

ACKNOWLEDGMENT

This work was supported and funded by the Ministry of Higher Education under the Fundamental Research Grant Scheme (FRGS/1/2018/ICT04/UTM/01/1) and UTM Reference of FRGS (R.K130000.7843.5F558).

REFERENCES

Ahmadian, S., Ahmadian, M., & Jalili, M. (2022). A deep learning based trust- and tag-aware recommender system. *Neurocomputing*, *488*, 557–571. doi:10.1016/j.neucom.2021.11.064

Alrashidi, M., Selamat, A., Ibrahim, R., & Krejcar, O. (2023). Social Recommendation for Social Networks Using Deep Learning Approach: A Systematic Review, Taxonomy, Issues, and Future Directions. *IEEE Access : Practical Innovations, Open Solutions*, *11*, 1–1. doi:10.1109/ACCESS.2023.3276988

Batmaz, Z., Yurekli, A., Bilge, A., & Kaleli, C. (2019). A review on deep learning for recommender systems: Challenges and remedies. *Artificial Intelligence Review*, *52*(1), 1–37. doi:10.1007/s10462-018-9654-y

C C, N., & Mohan, A.C C. (2019). A social recommender system using deep architecture and network embedding. *Applied Intelligence*, *49*(5), 1937–1953. doi:10.1007/s10489-018-1359-z

Catherine, R., & Cohen, W. (2017). TransNets. *Proceedings of the Eleventh ACM Conference on Recommender Systems*, (pp. 288–296). ACM. doi:10.1145/3109859.3109878

Chen, H., & Li, J. (2019). Adversarial tensor factorization for context-aware recommendation. *Proceedings of the 13th ACM Conference on Recommender Systems*, (pp. 363–367). ACM. doi:10.1145/3298689.3346987

Da'u, A., & Salim, N. (2020). Recommendation system based on deep learning methods: a systematic review and new directions. In Artificial Intelligence Review, 53(4). Springer Netherlands. doi:10.1007/s10462-019-09744-1

Gao, J., Zhang, C., Xu, Y., Luo, M., & Niu, Z. (2021). Hybrid microblog recommendation with heterogeneous features using deep neural network. *Expert Systems with Applications*, *167*, 114191. doi:10.1016/j.eswa.2020.114191

Guy, I. (2015). Social recommender systems. In Recommender Systems Handbook. Springer. doi:10.1007/978-1-4899-7637-6_15

Guy, I. (2022). Social Recommender Systems. In *Recommender Systems Handbook* (pp. 835–870). Springer US., doi:10.1007/978-1-0716-2197-4_22

Ko, Y.-J., Maystre, L., & Grossglauser, M. (2016). Collaborative Recurrent Neural Networks for Dynamic Recommender Systems. In R. J. Durrant & K.-E. Kim (Eds.), *Proceedings of The 8th Asian Conference on Machine Learning, 63*, 366–381. PMLR. https://proceedings.mlr.press/v63/ko101.html

Lei, C., Liu, D., Li, W., Zha, Z.-J., & Li, H. (2016). Comparative Deep Learning of Hybrid Representations for Image Recommendations. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (pp. 2545–2553). IEEE. doi:10.1109/CVPR.2016.279

Li, D., Chen, C., Liu, W., Lu, T., Gu, N., & Chu, S. M. (2017). Mixture-Rank Matrix Approximation for Collaborative Filtering. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, (pp. 477–485). IEEE.

Pan, Y., He, F., & Yu, H. (2020). Learning social representations with deep autoencoder for recommender system. *World Wide Web (Bussum)*, *23*(4), 2259–2279. doi:10.1007/s11280-020-00793-z

Ricci, F., Rokach, L., & Shapira, B. (2022). Recommender Systems: Techniques, Applications, and Challenges. In Recommender Systems Handbook (pp. 1–35). Springer US. doi:10.1007/978-1-0716-2197-4_1

Sedhain, S., Menon, A. K., Sanner, S., & Xie, L. (2015). AutoRec. *Proceedings of the 24th International Conference on World Wide Web*, (pp. 111–112). ACM. doi:10.1145/2740908.2742726

Shokeen, J., & Rana, C. (2020a). A study on features of social recommender systems. *Artificial Intelligence Review*, *53*(2), 965–988. doi:10.1007/s10462-019-09684-w

Shokeen, J., & Rana, C. (2020b). Social recommender systems: Techniques, domains, metrics, datasets and future scope. *Journal of Intelligent Information Systems*, *54*(3), 633–667. doi:10.1007/s10844-019-00578-5

Strub, F., Gaudel, R., & Mary, J. (2016). *Hybrid Recommender System based on Autoencoders*. ACM. 10.1145/2988450.2988456

Sun, Z., Guo, Q., Yang, J., Fang, H., Guo, G., Zhang, J., & Burke, R. (2019). Research commentary on recommendations with side information: A survey and research directions. *Electronic Commerce Research and Applications*, *37*(August), 100879. doi:10.1016/j.elerap.2019.100879

Tuan, T. X., & Phuong, T. M. (2017). 3D Convolutional Networks for Session-based Recommendation with Content Features. *Proceedings of the Eleventh ACM Conference on Recommender Systems*, (pp. 138–146). ACM. doi:10.1145/3109859.3109900

Wang, H., Wang, N., & Yeung, D.-Y. (2015). Collaborative Deep Learning for Recommender Systems. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (pp. 1235–1244). ACM. doi:10.1145/2783258.2783273

Wang, X., Yu, L., Ren, K., Tao, G., Zhang, W., Yu, Y., & Wang, J. (2017). Dynamic Attention Deep Model for Article Recommendation by Learning Human Editors' Demonstration. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (pp. 2051–2059). ACM. doi:10.1145/3097983.3098096

Wu, C.-Y., Ahmed, A., Beutel, A., & Smola, A. J. (2017). *Joint training of ratings and reviews with recurrent recommender networks*.

Wu, X., Shi, B., Dong, Y., Huang, C., & Chawla, N. V. (2019). Neural Tensor Factorization for Temporal Interaction Learning. *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, (pp. 537–545). ACM. doi:10.1145/3289600.3290998

Yao, D., Zhang, C., Huang, J., & Bi, J. (2017). SERM. *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, (pp. 2411–2414). ACM. doi:10.1145/3132847.3133056

Yin, H., Wang, W., Wang, H., Chen, L., & Zhou, X. (2017). Spatial-Aware Hierarchical Collaborative Deep Learning for POI Recommendation. *IEEE Transactions on Knowledge and Data Engineering*, *29*(11), 2537–2551. doi:10.1109/TKDE.2017.2741484

Zhang, J., Yang, Y., Zhuo, L., Tian, Q., & Liang, X. (2019a). Personalized Recommendation of Social Images by Constructing a User Interest Tree with Deep Features and Tag Trees. *IEEE Transactions on Multimedia*, *21*(11), 2762–2775. doi:10.1109/TMM.2019.2912124

Zhang, J., Yang, Y., Zhuo, L., Tian, Q., & Liang, X. (2019b). Personalized Recommendation of Social Images by Constructing a User Interest Tree With Deep Features and Tag Trees. *IEEE Transactions on Multimedia*, *21*(11), 2762–2775. doi:10.1109/TMM.2019.2912124

Zhang, Z., Xu, G., Zhang, P., & Wang, Y. (2017). Personalized recommendation algorithm for social networks based on comprehensive trust. *Applied Intelligence*, *47*(3), 659–669. doi:10.1007/s10489-017-0928-x

Zhao, Y., Wang, J., & Wang, F. (2015). Word embedding based retrieval model for similar cases recommendation. *2015 Chinese Automation Congress (CAC)*, (pp. 2268–2272). CAC. doi:10.1109/CAC.2015.7382881

Zheng, L., Noroozi, V., & Yu, P. S. (2017). Joint deep modeling of users and items using reviews for recommendation. *WSDM 2017 - Proceedings of the 10th ACM International Conference on Web Search and Data Mining*, (pp. 425–433). ACM. doi:10.1145/3018661.3018665

Zheng, Y., Tang, B., Ding, W., & Zhou, H. (2016). A Neural Autoregressive Approach to Collaborative Filtering. In M. F. Balcan & K. Q. Weinberger (Eds.), *Proceedings of The 33rd International Conference on Machine Learning* (Vol. 48, pp. 764–773). PMLR., https://proceedings.mlr.press/v48/zheng16.html

Muhammad Alrashidi has received a B.Sc. in Computer Science from Taibah University, Saudi Arabia. and M.Sc. in Information Technology from Bournemouth University, U.K. in 2015. He is currently a PhD candidate at the School of Computing at UTM University, Malaysia. His research interests include recommendation systems, deep learning, and social systems. He is interested in Artificial intelligence, Software Engineering, Databases, computer networks, programming, applications, and information technology management.

Ali Selamat received the B.Sc. degree (Hons.) in IT from Teesside University, U.K., in 1997, the M.Sc. degree in distributed multimedia interactive systems from Lancaster University, U.K., in 1998, and the Dr.Eng. degree from Osaka Prefecture University, Japan, in 2003. He is currently the Dean of the Malaysia–Japan International Institute of Technology (MJIIT), which is an educational institute that is established by the Ministry of Higher Education, Malaysia, to enhance Japanese oriented engineering education in Malaysia and Asia with the support from the Government of Japan through the Japanese International Cooperation Agency (JICA) and Universiti Teknologi Malaysia (UTM) together with 29 Japanese University Consortium (JUC). Prior to that, he was a Chief Information Officer (CIO) and the Director of Communication and Information Technology at UTM. He was elected as the Chair of IEEE Computer Society, Malaysia Section, under the Institute of Electrical and Electronics Engineers (IEEE), USA. He was previously assuming the position of Research Dean on the knowledge economy research alliance at UTM. He was a Principal Consultant of big data analytics at the Ministry of Higher Education, in 2010, a member of the Malaysia Artificial Intelligence Roadmaps, from 2020 to 2021, and a keynote speaker in many international conferences. He was a Visiting Professor at Kuwait University and few other universities in Japan, Saudi Arabia, and Indonesia. Currently, he is a Visiting Professor with the University of Hradec Králové, Czech Republic, and the Kagoshima Institute of Technology, Japan. His research interests include data analytics, digital transformations, knowledge management in higher education, key performance indicators, cloud-based software engineering, software agents, information retrievals, pattern recognition, genetic algorithms, neural networks, and soft computing.

Roliana Ibrahim is the Director of Applied Computing at the School of Computing, formerly known as Faculty of Computing. Previously she was the Head of Information Systems Department for three consecutive years at the faculty. She received her PhD in the field of Systems Science from Loughborough University, MSc Computer Science from Universiti Teknologi Malaysia, BSc (Honours) Computer Studies from Liverpool John Moores University. Being as a faculty member since 1999, she was previously a Coordinator of BSc Computer Science (Bioinformatics) Program and Master of Information Technology (IT Management). She is one the Editorial Board members of Journal of Information Systems Research and Innovation (JISRI) and the Guest Chief Editor of Special Issue on Science and Technology, Jurnal Teknologi Vol 70, No. 5. She used to serve as the Organizing Committee member of 5th Asian Conference in Intelligent Information and Database Systems (ACIIDS 2013), the Program Committee member of 8th, 7th and 5th Malaysia Software Engineering Conference (MySEC'14, MySEC'13, MySEC'11), the 1st, 2nd and 3rd International Conference on Research and Innovation in Information Systems (ICRIIS'09, ICRIIS'11, ICRIIS13). Most of the publications from the conferences are published by IEEE and are indexed in SCOPUS.

Hamido Fujita (Life Senior Member, IEEE) received the master's and Ph.D. degrees in information engineering from Tohoku University, Sendai, Japan, in 1985 and 1988, respectively, and the Doctor Honoris Causa degrees from Obuda University, Budapest, Hungary, in 2013, and Timisoara Technical University, Timisoara, Romania, in 2018.,He was an Adjunct Professor of computer science and artificial intelligence with Stockholm University, Stockholm, Sweden; the University of Technology Sydney, Ultimo, NSW, Australia; and others. He is currently a Distinguished Professor of artificial intelligence with Iwate Prefectural University, Takizawa, Japan. He is a Research Professor at the University of Granada, Granada, Spain; HTECH University, Ho Chi Minh City, Vietnam; Harbin Engineering University, Harbin, China; and others. He has supervised Ph.D. students jointly with the University of Laval, Quebec City, QC, Canada; Oregon State University, Corvallis, OR, USA; and many others. He has given many keynotes in many prestigious international conferences on intelligent system and subjective intelligence. He headed a number of projects, including intelligent Human-Computer Interaction (HCI), a project related to mental cloning for healthcare system as an intelligent user interface between human users and computers, and the Service Center of Port Engineering (SCOPE) project on virtual doctor systems for medical applications.,Dr. Fujita was a recipient of title of Honorary Professor from Obuda University in 2011 and the Honorary Scholar Award from the University of Technology Sydney in 2012. He is also a Highly Cited Researcher in Crossfield for the year 2019 and in computer science for the year 2020, 2021, and 2022, respectively, by Clarivate Analytics. He is the Chairperson of the i-SOMET Incorporated Association. He was an Editor-in-Chief of Knowledge-Based Systems from 2005 to 2020, and he is now an Emeritus Editor of Knowledge-Based Systems. He is an Editor-in-Chief of Applied Intelligence (Springer) and Healthcare Management (Tayler & Francis).