A Review of Applications of Deep Learning Techniques in Recommender Systems

Zhengyang Liu

Chinese University of Hong Kong(Shenzhen), Shenzhen, 518172, China

18396765791@163.com

Abstract. Nowadays, with tons of information brought by the development of the Internet, recommender systems are becoming more and more popular, and they are affecting many aspects of people’s daily lives. When people use the search engine, shop on e-commerce platforms or watch videos on video software, they are enjoying the content provided by recommender systems. Deep learning techniques also gained a rapid development recently due to their strengths in dealing with complex tasks. Deep learning techniques have triumphed in many areas. Also, there are more efforts contributed to the deep learning based recommender system. However, very few studies focused on giving a detailed review of deep -learning-based recommender system , and these reviews were usually conducted several years ago, which cannot reflect the latest research trends. Hence, the aim of this paper is to summary the deep learning techniques used in recommender systems. A classification scheme is given, and some possible future directions are pointed out.

Keywords: Deep learning, recommender system, review.

1. Introduction

Deep learning designates a class of machine learning models in which a large number of layers are present for the purpose of data transformation [1]. Numerous research fields, such as image identification, computer vision, and natural language processing, have made extensive use of it [2-4]. It typically outperforms conventional methods like random forest (RF) or K-nearest neighbor (KNN) [5]. The number of research on the application of deep learning in recommender system also experienced a surge during these years. When such techniques are incorporated into recommender system [6]. A lot of enterprises have developed their recommender systems with deep learning techniques. For example, Steck has developed an autoencoder-based deep learning algorithm called “the EASE model,” which notably increased ranking accuracy compared with the Slim model [6].

The achievement of deep learning in recommender systems requires a systematic overview to display the mainstream techniques and show advantages and disadvantages of these techniques. Although there are many reviews of recommender systems, few of them concentrate on deep learning. And those focusing on deep learning were usually published several years ago and cannot keep track of the new developments in deep learning algorithms. Thus, this paper intends to give a detailed review of the efforts made in the last 5 years and show some future directions in this area.

2. Recommender systems

Recommendation systems serve as information filters to recommend items to different users, with an environment that can collect and process data [7]. It offers personal filtering for all users, which helps them save time and effort searching for the information they want on the Internet. All recommender systems have to do with users and items. The systems will collect users' ratings on different items using different methods. By combining all the ratings together, a utility matrix will be generated, in which a row represents a user and a column represents an item. The ratings are elements of the utility matrix. In most cases, there will be a lot of missing values, and the problem is how to fill the utility matrix to further perform the recommendation. Also, according to their different methods to solve this problem, there are three kinds of systems: collaborative filtering system, content-based system, and hybrid system.

2.1. Collaborative filtering system

Content-based recommendation is achieved by learning from known user preferences for different items and making recommendations [8]. There are also two branches of collaborative-filtering system [9]. First, memory-based system can still be classified into two types: user-based system and item-based system. Since the principles of these two systems are similar, only the principle of user-based system is introduced here. For a user-based system, suppose there is a user X, and its rating on item Y is missing, then the system aims to find another set of users that have known ratings on item Y and have similar known ratings with X on other items. This set is called the "neighbor" of X, and the similarity can be calculated via many functions, like Euclidean distance, cosine similarity, and Pearson correlation coefficient. After the neighbor is given, the model can predict User X's rating of item Y based on the ratings of users in the neighbor on Y and these users' similarities with X (similarities usually served as weights). By repeatedly doing so, all missing values can be filled. Second, a model-based system developed a machine-learning-based model to predict ratings for different items [9]. Depending on the techniques used to build the model, there are two kinds of model-based systems: factorization-based models and neural network-based models. Under the assumption of factorization-bzsed models, utility matrix is gained by performing dot product on user latent matrix and item latent matrix. During the training process, it minimizes the reconstruction error between the ratings calculated by the latent factors and the true ratings. The neutral network-based model, similarly, uses neutral network techniques to fit the utility matrix.

The collaborative filtering method has many advantages. Firstly, no domain knowledge is needed. During the calculation process, only user behavior matters, so item details are irrelevant. Secondly, such method can capture the fact that users may have diverse preferences. However, without auxiliary information, the collaborative filtering system cannot handle the cold-start problem. Think about the occasion where there is a new item to handle: since there is no known feedback from any user in the utility matrix, the ratings cannot be computed.

2.2. Content-based system

Content-based recommender systems create item profiles based on their features or characteristics [9]. For instance, a person can be represented as a feature vector with attributes such as age, gender, and education level. By characterizing users and items as feature vectors, a classification or regression model can be trained for predicting utilities and making recommendations. The advantages of a content-based system are evident. Firstly, it is user-independent as it solely relies on the user's and item's profile extracted from the content for recommendations. This ensures the safety of users' private information and enables effective handling of new users or items. Secondly, the model generates more explainable recommendations by utilizing meaningful interacting features. Thirdly, this approach effectively captures dynamic user preferences since most preferences are variable with time [9]. However, there are drawbacks associated with content-analysis requirements. Rich domain knowledge is necessary due to reliance on discrete features, which may not always be easily obtainable. Furthermore, the system tends to recommend similar items to a user because these items have high scores.

2.3. Hybrid system

The hybrid model combines two or more classical collaborative filtering systems or model-based systems to deal with deficiency of single recommender techniques [9]. The Hybrid model can reduce the disadvantages of the two categories mentioned above, and most of the systems used today are hybrid models. It can be implemented by combining separate recommenders using ensemble techniques like linear weight and stacking or by simply adding content-based or collaborative filtering aspects to collaborative filtering or content-based models. For example, some side information can be added to the matrix factorization method to address its inability to deal with new users or items.

3. The classification of deep learning based recommendation model

The classic categorization of recommenders has been shown above. However, concerning recommender systems with deep learning architecture, it will be better to do the classification based on the categories of deep learning methods. Deep learning architectures can be classified into three categories: generative system, discriminative system, and hybrid system. Correspondingly, there will be three kinds of deep leaning based recommender systems.

3.1. Generative model based recommender system

In a generative model, the data flow begins in the output layer, passes hidden layers, and flows into input layers [10]. Its function, as its name implies, concentrates on learning the implicit features and distribution of the input data as well as trying to generate data that simulates the distribution of the original input. This model is employed for unsupervised pre-training and addressing issues related to probabilistic distributions [10]. Popular generative models applied in recommenders includes Deep Autoencoder (Deep AE), Restricted Boltzmann Machines (RBM), Generative Adversarial Network (GAN), etc.

3.1.1. Deep autoencoder based recommender system. An autoencoder (AE) is a type of neural network that operates with an unsupervised learning approach, designed to capture the underlying structure of input data for purposes of dimensionality reduction. It aims to produce an output that is a faithful reconstruction of the input data [10]. In deep learning, autoencoders have the capability to autonomously identify and extract key features from data, which makes it an effective tool to deal with the cases where conventional manual methods cannot extract features that are sufficient enough and mitigate the risk of overfitting at the same time [11]. There have been a lot of branches of deep autoencoder, like denoising autoencoder (DAE), variational autoencoder (VAE), and sparse autoencoder.

An elementary autoencoder (AE) is composed of three principal layers: an input layer, a hidden layer, and an output layer, each populated with a substantial number of neurons.And there are two important processes: encoding and decoding. Encoding is the phase in which the input data is projected onto the hidden layer, resulting in a transformation of the original data.The hidden layer of AE usually has a relatively narrow bottleneck to compress high-dimensional input into low-dimensional representation. Correspondingly, the decoding process means mapping the transformed data to the output layer to obtain the reconstructed data.

The basic structure of an autoencoder implies its application in the recommender system: It can be used to learn a low-dimensional representation of user or item profile at the hidden layer or be directly trained to fill in the missing entries in the utility matrix using results from output layer. For instance, Mao utilized a deep encoding network to convert a pattern image into a 128-dimensional vector, and produces the taste vector of a user based on his or her purchase history for the computation of cosine similarity in his textile pattern recommendation model [12]. Yu applied a stacked autoencoder to learn a high-level feature and turn a recommendation problem into a SoftMax classification problem [13]. A stacked autoencoder consists of several AE structures and can be viewed as a kind of multilayer AE. This enhancement bolsters its capacity to extract features from the input data. Fang proposed a differentially private variational autoencoder, in which he put the utility matrix in a variational autoencoder (VAE) to generate the predicted ratings using the decoder [14]. VAE is a branch of the autoencoder that can learn the prior distribution of input by the encoder and reconstruct it with decoder. It finds extensive application in recommender systems for the purpose of filling in the missing values within the utility matrix.

3.1.2. Generative adversarial network based recommender system. Generative Adversarial Network (GAN) consists of two models: a generator and a discriminator [15]. The generator, referred to as G, is designed to learn the underlying pattern of the actual data, ensuring that the synthesized samples closely mimic the authenticity of the original data.The discriminator, labeled as D, assesses the likelihood that a given sample originates from the authentic dataset instead of being artificially produced by G. One iteration of the training process also includes two stages: The first stage is keeping G fixed and training D to optimize its accuracy in distinguishing between genuine data samples and those fabricated by G. Correspondingly, the second stage is to train G to maximize the probability of D falsely labeling samples generated by G as coming from a real dataset. After training, both a strong generator and a strong discriminator are obtained.

GAN can produce more realistic samples than many other techniques, like VAE. Thus, it is becoming increasingly popular in recommender systems. Ali utilized a non-saturating Generative Adversarial Network (NS GAN) for the training process of the GAN-driven distributed representation component in his global citation recommender system [16].Zhou crafted an innovative framework known as PURE, designed to educate an impartial positive-unlabeled discriminator for the purpose of differentiating between genuine user-item associations and those deemed non-relevant, while also training a generator to comprehend the intrinsic continuous distribution of user-item interactions [17]. Chen addressed the cold-start problem with a trained generator generating cold item embedding that had almost the same distribution as the warm item embedding [18].

3.1.3. Restricted boltzmann machine based recommender system. The Restricted Boltzmann Machine (RBM) is a unique type of neural network comprising two distinct layers: one visible and one hidden. In RBM, the neurons are fully connected across the two layers, with no connections existing between neurons within the same layer. The visible layer of an RBM takes in the input and projects it onto the hidden layer, thereby learning the input's distribution. The goal of RBM is to train the network to enhance the likelihood of the vectors in the visible units, enabling it to probabilistically reconstruct the input data[10]. After training, the model can generate samples that follow the distribution of input using Gibbs Sampling [19].

In a recommender system, with the known ratings as the input, RBM can learn the probabilistic distribution of the ratings and give the predicted values of the missing ratings. Kirubahari came up with an improved RBM recommender system that uses RBM to learn the distribution and Bayesian optimization to perform multi-hyperparameter tuning [20]. The system worked well on the dataset of 100 K Movielens.

3.2. Discriminative model based recommender system

The discriminative model, extensively applied in supervised learning tasks such as regression and classification, employs a bottom-up strategy where data progresses from the input layer, through the hidden layers, to the output layer [10]. The primary techniques utilized in recommender systems can be categorized into three main types: multilayer perceptrons, convolutional neural networks, and recurrent neural networks:

Multilayer Perceptron (MLP) is a feed-forward neutral network that consists of single or multiple layers as well as nonlinear activation. Compared with other deep learning structures, it is simpler but useful in recommender systems. MLP has many variations. Li proposed a model named Auto MLP, which is exclusively composed of MLP blocks, designed for sequential recommendations that cater to users' interests, whether long-term or short-term, derived from their past interactions [21]. Since it only used MLP blocks, linear time and space complexity were achieved. Gao proposed SMLP4Rec, a model composed of pure MLP blocks that transforms the defective cascading architecture into a parallel configuration and incorporates normalization layers to reduce their detrimental effects on the model's performance while enhancing their beneficial outcomes. [22].

A Convolutional Neural Network (CNN) is composed of multiple convolutional layers, several pooling layers, an activation layer, and a fully connected layer [23]. The earlier convolutional layers are utilized to extract features from the input data, and the subsequent layers are responsible for recombining the features that have been extracted. After that, the pooling layer reduced the dimension of extracted features with operations like maximization or average. The convolutional layers of CNN have a strong ability to extract features, which makes CNN a good choice for recommender systems. Khan proposed a novel model named CNN-DSCK in which he used two parallel CNN structures to extract user and item features, respectively, and then used several fusion layers to concatenate the latent feature vectors for the regression of the fully connected layer [24]. CNN can also be used for further operations on the results of a recommender system. Arsytania used a type of 1D-CNN specifically designed to handle one-dimensional data to do the classification of data from the hybrid filtering method [25].

A key feature of a Recurrent Neural Network (RNN) is its recurrent connections, which allow information to be fed back into earlier layers, contrasting with the strictly feed-forward connections found in other types of neural networks [1]. This makes RNN capable of dealing with sequential information. RNN also has many branches, such as Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU).

As mentioned above, in recommender systems, RNN is widely used to deal with sequential user behaviors. Huang designed a RNN-based system for long-term recommendation in which the interactions between the agent (recommender system) and environment (user) are simulated by the recurrent neural network (RNN) [26]. Lee's proposed RNN-based recommender system can recommend items by multiple periods in a time sequence [27].

3.3. Hybrid model

Hybrid models refer to those deep learning based recommender systems that make use of two or more kinds of deep learning algorithms. It can be implemented by applying different techniques at different stages or by directly combining the algorithms of different techniques. Hybrid models can combine the advantages of different techniques and avoid the disadvantages of some techniques. Thus, most of the new models proposed in the last five years are hybrid models.

The composition of hybrid models is quite diverse. Xu proposed RCNN, a model that capitalizes on the recurrent framework of RNNs to handle intricate long-term dependencies while also harnessing the convolutional capabilities of CNNs to identify short-term sequential motifs [28]. Duong introduces the Half Convolutional Autoencoder, employing convolutional layers to uncover high-order relationships among structured features, utilizing side information, thereby creating a resilient feature vector [29]. Hiriyannaiah proposed a DeepLSGR model, which consists of an ensemble of LSTM and GRU in its hidden layers, and delivers suggestions predicated on the foresight of user ratings derived from their textual feedback [30]. The system achieved an accuracy of 97% on the Amazon Fine Food Reviews dataset and the OpinRank Dataset [22].

4. Conclusion

This paper provides a comprehensive overview of the latest advancements in deep learning-based recommender systems and a classification system, with lots of academic papers, mainly from 2020 to 2024. In this paper, the basic structures and strengths of different deep learning techniques are introduced with a bunch of academic papers as examples, which can give readers a preliminary understanding of how these techniques are being employed in recommender systems. This paper also has some limitations. For instance, although this paper covers different deep learning techniques, it does not attach too much content to their evaluation metrics. The evaluation metrics can also affect the evolution of deep learning-based recommender systems, and this direction is worth considering in the future. Apart from this, the databases used for this paper are IEEE Explore, Elsevier, the ACM digital library, Archive, Springer, and Web of Science. This paper cannot cover all databases, so the content may not be exhaustive enough.

As for future directions, since large companies may have a wider range of business in the future, cross-domain recommendations may become popular. Some domains do have correlations. Suppose a person buys a computer on an app, then some paid software or video games that one might be interested in can be recommended based on the configurations of the computer one has bought. Apart from this, the use of auxiliary information can have more significance. It presents a viable solution to the cold-start problems of many recommender systems, and learning about auxiliary information is relatively straightforward with deep learning techniques.

References

[1] Dargan S, Kumar M, Ayyagari MR, Kumar G. A survey of deep learning and its applications: a new paradigm to machine learning. Archives of Computational Methods in Engineering. 2020; 27:1071-92.

[2] Mehrish A, Majumder N, Bharadwaj R, Mihalcea R, Poria S. A review of deep learning techniques for speech processing. Information Fusion. 2023; 99:101869.

[3] Chai J, Zeng H, Li A, Ngai EW. Deep learning in computer vision: A critical review of emerging techniques and application scenarios. Machine Learning with Applications. 2021; 6:100134.

[4] Li Y, editor Research and application of deep learning in image recognition. 2022 IEEE 2nd international conference on power, electronics and computer applications (ICPECA); 2022: IEEE.

[5] Zhang S, Yao L, Sun A, Tay Y. Deep learning based recommender system: A survey and new perspectives. ACM computing surveys (CSUR). 2019; 52(1):1-38.

[6] Steck H, Baltrunas L, Elahi E, Liang D, Raimond Y, Basilico J. Deep learning for recommender systems: A Netflix case study. AI Magazine. 2021; 42(3):7-18.

[7] Ko H, Lee S, Park Y, Choi A. A survey of recommendation systems: recommendation models, techniques, and application fields. Electronics. 2022; 11(1):141.

[8] Khanal SS, Prasad P, Alsadoon A, Maag A. A systematic review: machine learning based recommendation systems for e-learning. Education and Information Technologies. 2020; 25(4):2635-64.

[9] Roy D, Dutta M. A systematic review and research perspective on recommender systems. Journal of Big Data. 2022; 9(1):59.

[10] Shrestha A, Mahmood A. Review of deep learning algorithms and architectures. IEEE access. 2019; 7:53040-65.

[11] Li P, Pei Y, Li J. A comprehensive survey on design and application of autoencoder in deep learning. Applied Soft Computing. 2023; 138:110176.

[12] Mao K, Wu S, He JJ, Huang HC, Yin YL, Ren ZJ. Textile pattern recommendations with convolutional neural networks and autoencoder. Concurrency and Computation-Practice & Experience. 2023; 35(18).

[13] Yu M, Quan T, Peng Q, Yu X, Liu L. A model-based collaborate filtering algorithm based on stacked AutoEncoder. Neural Computing and Applications. 2022:1-9.

[14] Fang L, Du B, Wu C. Differentially private recommender system with variational autoencoders. Knowledge-Based Systems. 2022; 250:109044.

[15] Gui J, Sun Z, Wen Y, Tao D, Ye J. A review on generative adversarial networks: Algorithms, theory, and applications. IEEE transactions on knowledge and data engineering. 2021; 35(4):3313-32.

[16] Ali Z, Qi G, Muhammad K, Kefalas P, Khusro S. Global citation recommendation employing generative adversarial network. Expert Systems with Applications. 2021; 180:114888.

[17] Zhou Y, Xu J, Wu J, Taghavi Z, Korpeoglu E, Achan K, et al., editors. PURE: Positive-unlabeled recommendation with generative adversarial network. Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining; 2021.

[18] Chen H, Wang Z, Huang F, Huang X, Xu Y, Lin Y, et al., editors. Generative adversarial framework for cold-start item recommendation. Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval; 2022.

[19] Ghojogh B, Ghodsi A, Karray F, Crowley M. Restricted boltzmann machine and deep belief network: Tutorial and survey. arXiv preprint arXiv:210712521. 2021.

[20] Kirubahari R, Amali SMJ. An improved restricted Boltzmann Machine using Bayesian Optimization for Recommender Systems. Evolving Systems. 2024;15(3):1099-111.

[21] Li M, Zhang Z, Zhao X, Wang W, Zhao M, Wu R, et al., editors. Automlp: Automated mlp for sequential recommendations. Proceedings of the ACM Web Conference 2023; 2023.

[22] Gao J, Zhao X, Li M, Zhao M, Wu R, Guo R, et al. SMLP4Rec: An Efficient all-MLP Architecture for Sequential Recommendations. ACM Transactions on Information Systems. 2024;42(3):1-23.

[23] Alzubaidi L, Zhang J, Humaidi AJ, Al-Dujaili A, Duan Y, Al-Shamma O, et al. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. Journal of big Data. 2021;8:1-74.

[24] Khan ZY, Niu Z. CNN with depthwise separable convolutions and combined kernels for rating prediction. Expert Systems with Applications. 2021;170:114528.

[25] Arsytania IH, Setiawan EB, Kurniawan I. Movie Recommender System with Cascade Hybrid Filtering Using Convolutional Neural Network. Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI). 2024;9(4):1262-74.

[26] Huang L, Fu M, Li F, Qu H, Liu Y, Chen W. A deep reinforcement learning based long-term recommender system. Knowledge-based systems. 2021;213:106706.

[27] Lee HI, Choi IY, Moon HS, Kim JK. A Multi-Period Product Recommender System in Online Food Market based on Recurrent Neural Networks. Sustainability. 2020;12(3):969.

[28] Xu C, Zhao P, Liu Y, Xu J, S. Sheng VSS, Cui Z, et al., editors. Recurrent convolutional neural network for sequential recommendation. The world wide web conference; 2019.

[29] Duong TN, Doan NN, Do TG, Tran MH, Nguyen DM, Dang QH. Utilizing Half Convolutional Autoencoder to Generate User and Item Vectors for Initialization in Matrix Factorization. Future Internet. 2022;14(1).

[30] Hiriyannaiah S, GM S, Srinivasa K. DeepLSGR: Neural collaborative filtering for recommendation systems in smart community. Multimedia Tools and Applications. 2023;82(6):8709-28.