

Recommender System for Telecom Packages Based on the Deep & Cross Network

Congming Shi

Anyang Normal University <https://orcid.org/0000-0002-4666-553X>

Wen Wang

Kunming University of Science and Technology

Shoulin Wei (✉ weishoulin@kust.edu.cn)

Kunming University of Science and Technology <https://orcid.org/0000-0002-3547-4025>

Feiya Lv

Anyang Normal University

Research

Keywords: Machine Learning, Recommender System, Deep Learning, Telecom Packages, Feature Engineering

Posted Date: September 16th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-889572/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Recommender System for Telecom Packages Based on the Deep & Cross Network

Congming Shi¹, Wen Wang^{2,3}, Shoulin Wei^{2,3,*}, Feiya Lv¹

¹ School of Software Engineering, Anyang Normal University, China

² Faculty of Information Engineering and Automation, Kunming University of Science and Technology, China

³ Computer Technology Application Key Lab of Yunnan Province, Kunming University of Science and Technology, China

{shicongming@astrolab.cn, wangwen@cnlab.net, weishoulin@kust.edu.cn, lvfeiya0215@126.com}

Abstract: With the evolution of 5G technology, the telecom industry has influenced the livelihood of people and impacted the development of the national economy significantly. To increase revenue per customer and secure long-term contracts of users, telecommunications firms and enterprises have launched different types of telecom packages to satisfy the varying requirements of users. Although several recommender systems have been proposed in recent years for telecommunication package recommendation, extracting effective feature information from large and complex consumption data remains challenging. Considering the telecom package recommendation problems, traditional recommendation methods either use complex expert feature engineering or fail to perform end-to-end deep learning training. In this study, a recommender system based on deep & cross network (DCN), deep belief network (DBN), Embedding, and Word2Vec is proposed using the powerful learning abilities of deep learning. The proposed system fits the telecom package recommender system in terms of click-through rate prediction to provide a potential solution for the recommendation challenges faced by telecom enterprises. The proposed model can effectively capture the finite order interactional features and deep hidden features. Additionally, the text information in the data is completely used to further improve the recommendation ability of the model. Moreover, the proposed method does not require feature engineering. We conducted comprehensive experiments using real-world datasets, and the results verify that our method can generate improved recommendation accuracy in comparison with those observed in DBN, DCN, deep factorization machine, and deep neural network models individually.

Keywords: Machine Learning, Recommender System, Deep Learning, Telecom Packages, Feature Engineering

1. Introduction

The telecom industry has become a critical support industry, serving almost all

* Corresponding author

domains in the world owing to the development of communication technology. Furthermore, with the evolution of 5G technology, the telecom industry is closely associated with the livelihood of people and contributes to the development of the national economy. This new era of 5G networks is expected to increase the competitive pressure among telecom enterprises. Particularly, the implementation of number portability that allows mobile users to switch from one operator to another without changing numbers further enhances the competitive pressure on various service providers, such as China Unicom, China Mobile, and China Telecom, in the market. To retain existing customers and attract new customers, operators have launched various telecom packages, which often combine Internet services, cable television and telephone services, rendering it more cost-effective than purchasing each service individually. However, telecom enterprises face challenges in recommending a suitable telecom package to users with similar needs.

In other words, if packages recommended by telecom enterprises are regarded as recommended items in recommender systems, the challenge faced by these enterprises is transformed into the task of improving the probability of a user clicking on a recommended item. Recommender system is an important basic technology of machine learning that can be used as a potential solution to address the recommendation challenges faced by telecom enterprises. As a recommender system aims to solve the problem of information overload, it can be used to assist users in obtaining their best choices from multiple options based on their interests^[1-3]. Most recommender systems, including content-based^[4,5], collaborative filtering (CF)^[6,7] and knowledge-based^[8,9] systems, rely on a single-criterion rating as a primary source to obtain the potential recommendation results^[10]. Although these models have remarkably succeeded in several applications, they are generally affected by issues such as data sparseness^[11]. Moreover, they fail to generate effective recommendations in the case of multi-dimensionality, such as contextual information^[12], despite being widely adopted. In other words, traditional recommender systems based on traditional item-based CF cannot satisfy the recommendation requirements of telecom enterprises with high-dimensional features for customers.

However, the recent success of deep learning in multiple application fields has led to the combination of recommender systems with deep learning to be considered as a novel and potential solution to address the recommendation challenges faced by telecom enterprises. This combination can effectively capture non-linear features and non-trivial user or item relationships and enable the codification of highly complex abstractions as data representations in the higher layers. In recommender systems, Product-based Neural Networks (PNN) can be used to learn high-order feature interactions^[13], Deep neural networks (DNN) with more than three layers can obtain improved generalization capability for unseen feature combinations^[14], and the Wide & Deep network can aid in modeling low-order and high-order feature interactions^[15]. A Deep Belief Network (DBN) is generated by stacking multiple Restricted Boltzmann machines (RBM) and greedy training of the RBM^[16,17]. As DBN is a competent generative model based on unlabeled data with a powerful predictive classification function, it can be used in recommender system^[18]. Additionally, a Factorization-

Machine based neural network (DeepFM) uses the power of factorization machines for recommendation and deep learning for feature learning^[19]. The eXtreme Deep Factorization Machine (xDeepFM) combines the Compressed Interaction Network (CIN) and a classical DNN, which can learn certain bounded-degree feature interactions explicitly and arbitrary low-order and high-order feature interactions implicitly^[20]. Deep & Cross Network (DCN) can maintain the DNN's benefit of the generalization capability to unseen feature combinations and efficiently obtain certain bounded-degree feature interactions^[21]. Although these deep learning-based models were developed to predict the Click Through Rate (CTR) on the Internet, their direct application to telecom package recommender systems remain a challenge.

Therefore, we propose a novel recommender system for telecom packages based on DCN, DBN, Embedding, and Word2Vec. We used Embedding^[22] for mapping discrete categorical features, and Word2Vec^[23] was used to obtain similarities between numerical features and data wrangling (imputation and drop). Additionally, the raw dataset conversion was used to fit the format of the CTR paradigm.

The remainder of the paper is organized as follows. A brief overview of the related work and proposed model is presented in Section 2. Section 3 explains the comprehensive set of experiments performed to validate the proposed method by comparing its performances with that of other methods. Finally, the conclusions are summarized in Section 4.

2. Methods

2.1 Related Work

With the rapid advancement of smartphones, communication service providers need to establish a comprehensive recommender system to ensure high-quality and custom-made services for different customer demands. Before the development of deep learning, traditional machine learning and recommender systems were used for telecom package recommendations.

The K-Nearest Neighbor (KNN)^[24] can separate the historical datasets of customers into several classes to predict the classification of a new customer. However, the computing cost is typically high for a large characteristic dimension. Moreover, KNN is sensitive to noise in the dataset and exhibits a low fault tolerance.

CF is a popular technique for developing recommendation systems and has been employed in several applications^[25-27]. Herein, the behaviors of users based on the data collected in the past are analyzed to determine the connection between users and their items of interest, which can aid in recommending items to the users with similar preferences and support considering the opinions of different users. A hybrid recommendation approach that combines user-based and item-based CF techniques was proposed for mobile product and service recommendation^[28-30]. Although CF can achieve efficiency based on similar measurements of users' interests and recommended items, it is difficult to exploit the cross features completely owing to

the lack of deep and effective feature extraction. Furthermore, problems of cold start and low recommendation accuracy exist in CF recommendation systems.

Although recommender systems are widely used in the modern telecom industry, the methods employed have been focused on traditional machine learning and CF techniques. Only a few practical or academic cases exist, wherein the state-of-the-art deep CTR model was used in telecom package recommendations. However, the significant success of deep learning introduced deep learning-based CTR into the field of recommender systems, which were applied in websites on the Internet.

Factorization Machine (FM)^[31] integrated with deep learning exhibited outstanding performance and achieved promising results in CTR prediction. However, the two types of deep learning models proposed, namely FM-supported Neural Network and Sampling-based Neural Network, can capture only high-order feature interactions^[32]. Wide & Deep Learning^[15], which trains wide linear models and deep neural networks simultaneously to combine the benefits of memorization and generalization for recommender systems, was initially introduced for app recommendation in Google Play. However, it requires the designing of cross-product transformations through expertise feature engineering.

PNN^[13] attempts to capture high-order feature interactions by involving a product layer between the embedding and hidden layers. Unlike the traditional embedding combined with multi-layer perceptron, PNN explicitly captures the second-order feature correlation in multiple fields.

Influenced by Wide & Deep, DeepFM^[19] replaced the “wide” segment of the Wide & Deep model with FM to learn the low-order features. The “deep” segment was developed with a feedforward neural network to learn high-order features. Both the “wide” and “deep” segments share identical raw feature inputs. In comparison with the Wide & Deep model, DeepFM is an end-to-end learning model that is independent of manual feature engineering. Moreover, the xDeepFM^[20] explicitly models low-order and high-order feature interactions by using a novel CIN segment.

To address the drawbacks of the aforementioned models, we propose a novel recommender system for telecom packages based on DCN, DBN, Embedding, and Word2Vec, which can provide a potential solution to the recommendation challenges faced by telecom enterprises.

2.2 Proposed Method

In this section, we describe the architecture of the proposed model and present its underlying theory. The process of developing the dataset is explained, and the methods used to evaluate the proposed model are outlined.

2.2.1 Model Overview

Fig. 1 illustrates the architecture of the proposed model based on the DCN. The model begins with an embedding and stacking layer, wherein the sparse, dense, and converted features are stacked with Word2Vec. Subsequently, the stacked value is fed into a cross network and a deep network in parallel inputs. The two results are combined in the combining layer, and the final output is computed through a single-

layer neural network and activated by a sigmoid function. Unlike the original DCN, the Word2Vec model is combined with the DCN to train text features, which solves the problem of DCN being unable to process text information. This enhances the recommendation accuracy of the proposed model. Moreover, using the DBN network to replace the “deep” segment of the original DCN can exploit the advantages of DBN completely.

2.2.2 Dataset

The dataset used in this study was provided by the China Unicom Research Labs, which is available from DataFoundatin² in csv format. The dataset comprised 743 990 samples. Each sample indicates a consumption record, implying that a telecom package is currently being used by a customer. The dataset comprises 27 fields, such as USERID, current_type, and service_type. Excluding the USERID field, 22 features, such as age, gender, and complaint level, are used to describe a user, whereas four features, such as the type of package and whether the mixed package is opted, are used to describe the current package. Each categorical field is represented as a vector of one-hot encoding. Conversely, each continuous field is represented as the value itself or as a vector of one-hot encoding discretization.

For the dataset to be suitable for the CRT prediction task, it must be transformed into binary classification data. In the case of CTR, each sample represents a historical click behavior from a specific user on a telecom package, which is considered as a displayed advertisement. Typically, all samples of the raw dataset serve as positive samples. The raw dataset used in this study comprised 12 unique telecom packages. We randomly chose five samples from 11 telecom packages to construct negative samples by maintaining consistency with the user-side features. According to the CTR model, a “clicked” column must be created in the dataset. Additionally, the click values of positive and negative samples were "1" and "0", respectively.

² <https://www.datafountain.cn/competitions/311/datasets>

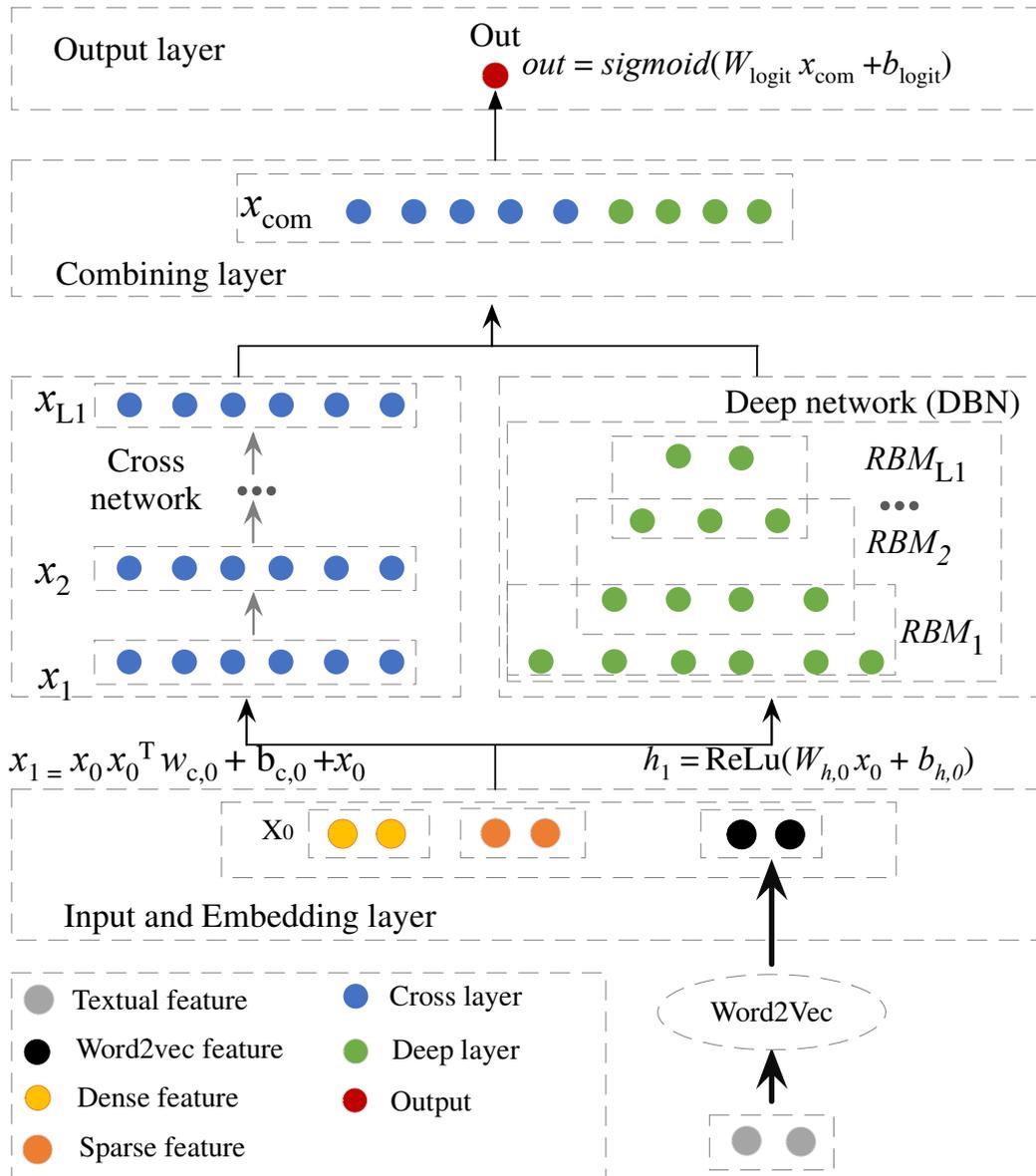


Fig.1 The architecture of the proposed model based on the DCN.

2.2.3 Data Wrangling

As the raw dataset comprised missing values, we used two methods to determine these values from the existing data.

1) In the case of numerical features, the datasets were grouped in terms of the current package by first calculating the mean of the non-missing values in a column and then replacing the missing values within each group separately, independent of others groups.

2) In the case of categorical features, the datasets were grouped considering the current package, replacing the missing data with the most frequently occurring values within each group separately.

2.2.4 Theoretical Analysis

1) Input and Embedding Layer

We classified all features of the raw dataset into three types, namely categorical features, continuous integer features (such as times of payment), and decimal numerical features. We used Embedding for dimensional reduction to transform the categorical features into a low-dimensional space. Additionally, we performed normalization on the continuous integer features. In the case of numerical decimal features, we converted the decimal type to textual type and employed Word2Vec to transform these features into embedding vectors. Finally, we stacked the embedding vectors of categorical features, normalized integer features, and textual features into one vector:

$$x_0 = [X_{emb}^T, X_{norm}^T, X_{text}^T] \quad (1)$$

x_0 serves as the input for both cross and deep networks simultaneously.

2) Cross Network

The cross network comprises several cross layers, wherein each cross layer can be calculated as follows:

$$x_{l+1} = x_0 x_l^T w_l + b_l + x_l = f(x_l, w_l, b_l) + x_l, \quad (2)$$

where x_l and x_{l+1} represent the outputs of the cross layer l and $l+1$, respectively; and w_l and b_l denote the weight and bias parameters of the l -th layer, respectively. All the aforementioned variables are column vectors. Furthermore, the features of each layer are cross-combined by the previous layer and original features (x_0) and added back to the previous layer. This is similar to the structure of a residual network, wherein the function f of each layer fits the residual of $x_{l+1} - x_l$. Thus, the gradient dispersion problem caused by the DNN can be solved using this residual network.

3) Deep Network

The deep network is a fully connected feedforward neural network that shares the same input with the cross network. It enables the model to learn the combined features of higher-order nonlinearity. The output of each layer can be calculated as:

$$h_{i+1} = F(W_i h_i + b_i), \quad (3)$$

where h_{i+1} and h_i represent the output of the $i+1$ -th and i -th layers, respectively; w_i and b_i denote the weight and bias parameters of the i -th layer, respectively; and F indicates the activation function to enhance the non-linear capability of the network. The Rectified Linear Units (ReLU)^[33] is used as the activation function in the proposed model owing to its calculation simplicity. Moreover, the convergence speed of ReLU significantly outperforms those of other activation functions, such as sigmoid^[34] and tanh^[35].

When x_0 flows through the cross network and deep network, x_{cross} and x_{deep} are generated, respectively. In the combining layer, x_{cross} and x_{deep} are stacked into x_{com} as follows:

$$x_{com} = (x_{cross}, x_{deep}), \quad (4)$$

Here, we can calculate the output through a perceptron using a sigmoid activation function δ , which is calculated as:

$$out = \delta(wx_{com} + b). \quad (5)$$

3. Results and discussion

In this section, we evaluate the performance of the proposed model using the dataset provided by the China Unicom Research Labs. The experiments were conducted using a high-performance computer configured with a dual Intel Xeon CPUs E5-2660 v4 processor at 2.00 GHz, including 56 cores, 630 GB memory, 1 TB hard disk, and 3 Tesla P40 graphic processing unit (GPU) cards. The system used Ubuntu 18.04.1 LTS with the Python 3.6.5, Torch 1.1.0, and Tensorflow 1.14.0.

3.1 Comparison of models and their hyper parameters

We compared the proposed model with four models, namely the DBN, DCN, DeepFM, and DNN. All models used Adam^[36] optimizer with the learning rate and mini-batch size set to 0.001 and 1000, respectively. Furthermore, as dropout can be used to prevent neural networks from overfitting^[37], the dropout rates of all models were set to 0.2. The Adam optimizer addressed the sparseness faced by the models, whereas the dropout prevented the neural networks from overfitting.

Proposed Model: Table 1 lists the hyperparameters of the proposed model. The dimension of word embedding was set to eight, with three and four cross and hidden layers, respectively, in the deep network. Additionally, the number of hidden neurons in each hidden layer was 32, and the number of RBMs was set to 4, with a network structure of [200, 100, 50, 20].

Table 1 Hyperparameters of the proposed model.

Parameter name	Parameter value
Embedding dimension	8
Cross layer size	3
Deep layer size	4
Number of neurons in each layer	32
Dropout rate	0.2
Number of RBM	4
Network structure of RBMs	[200, 100, 50, 20]
Mini-batch size	1000
Learning rate	0.001

DBN: Table 2 lists the hyperparameters of the DBN. The value of k in CK-k was set to two, and four RBMs were used, with the network structure set to [200, 100, 50, 20].

Table 2 Hyperparameters of the DBN.

Parameter name	Parameter value
Value of k used in CD-k	2
Dropout rate	0.2
Number of RBM	4
Network structure of RBMs	[200, 100, 50, 20]
Mini-batch size	1000
Learning rate	0.001

DCN: Table 3 lists the hyperparameters of the DCN. Similar to the proposed model, the dimension of word embedding was set to eight, with three and four cross and hidden layers, respectively, in the deep network. The number of hidden neurons in each hidden layer was 32.

Table 3 Hyperparameters of the DCN.

Parameter name	Parameter value
Embedding dimension	8
Cross layer size	3
Deep layer size	4
Number of neurons in each layer	32
Dropout rate	0.2
Mini-batch size	1000
Learning rate	0.001

DeepFM: Table 4 lists the hyperparameters of the DeepFM. Herein, the dimension of word embedding was set to eight, with four hidden layers in the deep network and 32 hidden neurons in each hidden layer.

Table 4 Hyperparameters of the DeepFM.

Parameter name	Parameter value
Embedding dimension	8
Deep layer size	4
Number of neurons in each layer	32
Dropout rate	0.2
Mini-batch size	1000
Learning rate	0.001

DNN: Table 5 lists the hyperparameters of the DNN. Herein, the deep network comprised four hidden layers, with the network structure set to [200, 100, 50, 20].

Table 5 The hyper-parameters of DNN.

Parameter Name	Parameter Value
Deep layer size	4
Network structure	[200, 100, 50, 20]
Dropout rate	0.2
Mini-batch size	1000
Learning rate	0.001

3.2 Evaluation Methods

We used the Area Under ROC Curve (AUC)^[38-40] and cross entropy (Logloss)^[41], which are widely used metrics in the CTR prediction, as the evaluation metrics in our experiments. The upper bound of AUC was set to 1, wherein a larger value indicates better performance. Logloss measures the distance between two distributions, and a smaller log loss value indicates better performance. All experiments were repeated 10 times to determine the averaged results.

Fig. 2 illustrates the AUC metrics. As indicated in the figure, the AUC metrics obtained by the proposed model were larger than those of the other models. Furthermore, the AUC metrics increased rapidly as the value of epoch increased, and subsequently attained the optimal or maximum/best values.

Fig. 3 illustrates the Logloss metrics, which shows that the Logloss metrics obtained by the proposed model were less than those of the other models. Moreover, the Logloss metrics decreased rapidly with increase in the epoch value, which then attained the optimal or minimum/best values. Table 6 summarizes the best metrics of both AUC and Logloss obtained by all models.

In other words, the proposed model outperformed all the other models. First, in comparison with other models that do not consider text feature information, the proposed model combined with the Word2Vec model, which can train text information, can extract richer text information. This significantly improved the performance of the proposed model than those of other models. Second, unlike DBN and DNN, the proposed model considers the extraction of high-level features and the impact of low-level features, which ensures that the proposed model achieves better performance. Third, unlike the DeepFM and DCN, the proposed model uses the DBN rather than the DNN. This implies that the proposed model is less likely to encounter the local optimal situation. Therefore, we trained the proposed model in the deep network layer considering the global optimal situation

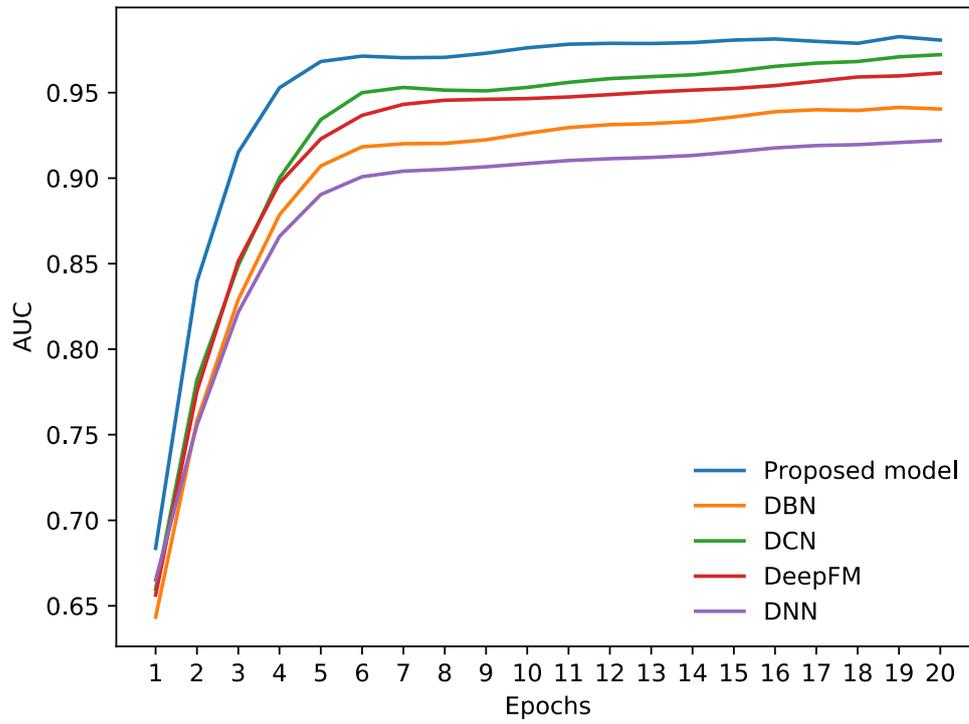


Fig.2 The change curve of AUC vs. Epoch.

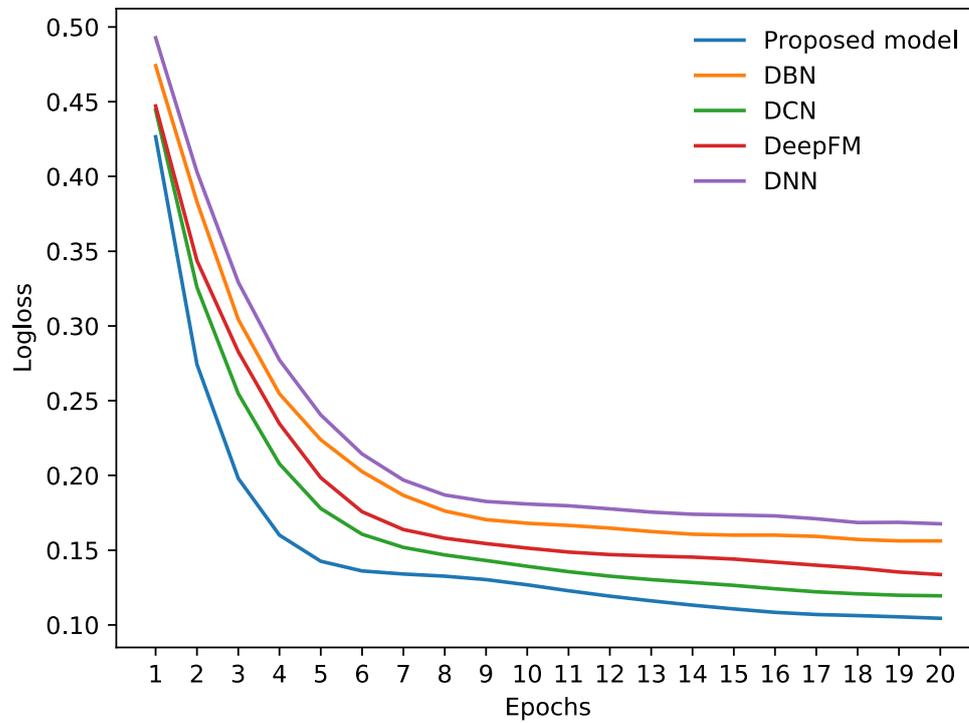


Fig.3 The change curve of Logloss vs. Epoch.

Table 6 Best LogLoss and AUC of different models after 20 epochs.

Model \ Metric	DBN	DNN	DeepFM	DCN	Proposed model
Logloss	0.1432	0.1602	0.1348	0.1202	0.1061
AUC	0.9413	0.9233	0.9602	0.9713	0.9814

4. Conclusions

To provide a potential solution to the recommendation challenges faced by telecom enterprises, we proposed a recommender system for telecom packages using the DCN, DBN, Embedding, and Word2Vec models. The experimental results indicate that the proposed model outperforms the existing models, namely DBN, DCN, DeepFM, and DNN, in terms of AUC and Logloss. The aspects contributing to the performance improvement of the proposed model can be summarized as follows: 1) the addition of the new embedding features generated by applying Word2Vec on textual features originated from the decimal features, which enables the model to learn more low-order and high-order feature interactions; 2) the replacement of the deep segment of the original DCN by DBN renders the proposed model an efficient generative model, which considers unlabeled data and a powerful predictive classification function. In the future, we intend to investigate the distributed training on multiple GPUs and the imputation of missing values using machine learning techniques.

List of Abbreviations

- (1) RBM: Restricted Boltzmann Machine
- (2) DBN: Deep Belief Networks
- (3) CD- k : k -step Contrastive Divergence learning
- (4) DCN: Deep & Cross Network
- (5) FM: Factorization Machine
- (6) DeepFM: Deep Factorization Machines
- (7) DNN: Deep Neural Networks
- (8) CTR: Click Through Rate
- (9) KNN: K-Nearest Neighbor
- (10) GPU: Graphics Processing Unit
- (11) AUC: Area Under the ROC Curve
- (12) ROC: Receiver Operating Characteristic
- (13) ReLU: Rectified Linear Units
- (14) CIN: Compressed Interaction Network
- (15) xDeepFM: eXtreme Deep Factorization Machine

- (16) PNN: Product-based Neural Network
- (17) CF: Collaborative filtering
- (18) Adam: Adaptive Moment Estimation

Availability of data and material

The Dataset used in this paper is provided by China Unicom Research Labs, and is available at <https://www.datafountain.cn/competitions/311/datasets>.

Competing interests

We declare that there is no conflict of interest regarding this submission.

Funding

This work is supported by the National Natural Science Foundation of China (No.62003004). The Key Science and Technology Program of Henan Province (No.212102210611), Anyang Science and Technology Plan Project (Grant No.2022C01NY020), and the Research and Cultivation Fund Project of Anyang Normal University (AYNUKPY-2020-25).

Authors' contributions

Congming Shi and Wen Wang introduced the idea of this work and finished partial English writing. Shoulin Wei, and Feiya Lv designed the algorithms and finished partial English writing. Shoulin Wei, and Wen Wang implemented the algorithms and finished partial English writing.

Acknowledgements

Not applicable.

References

Uncategorized References

- [1] Lu J, Wu D, Mao M, et al.Recommender system application developments: a survey[J].Decision Support Systems,2015, 74: 12-32.
- [2] Liu L, Mehandjiev N, Xu D-L. Multi-criteria service recommendation based on user criteria preferences[C].Proceedings of the fifth ACM conference on Recommender systems,2011: 77-84.
- [3] Al-Ghuribi S M, Noah S a M.Multi-criteria review-based recommender system—the state of the art[J].IEEE Access,2019, 7: 169446-169468.
- [4] Pazzani M J, Billsus D: Content-based recommendation systems,The adaptive web: Springer,2007: 325-341.
- [5] D'addio R, Conrado M, Resende S, et al. Generating recommendations based on robust term extraction from users' reviews[C].Proceedings of the 20th Brazilian Symposium on Multimedia and the Web,2014: 55-58.
- [6] García-Cumbreras M Á, Montejo-Ráez A, Díaz-Galiano M C.Pessimists and optimists: Improving collaborative filtering through sentiment analysis[J].Expert Systems with Applications,2013, 40 (17): 6758-6765.
- [7] Yang C, Yu X, Liu Y, et al.Collaborative filtering with weighted opinion aspects[J].Neurocomputing,2016, 210: 185-196.
- [8] Tarus J K, Niu Z, Mustafa G.Knowledge-based recommendation: a review of ontology-based recommender systems for e-learning[J].Artificial intelligence review,2018, 50 (1): 21-48.
- [9] Vijayakumar V, Vairavasundaram S, Logesh R, et al.Effective knowledge based recommender system for tailored multiple point of interest recommendation[J].International Journal of Web Portals (IJWP),2019, 11 (1): 1-18.
- [10] Wang Y, Wang M, Xu W.A sentiment-enhanced hybrid recommender system for movie recommendation: a big data analytics framework[J].Wireless Communications and Mobile Computing, 2018.

- [11] Da'u A, Salim N. Sentiment-aware deep recommender system with neural attention networks[J]. *IEEE Access*, 2019, 7: 45472-45484.
- [12] Wasid M, Ali R. An improved recommender system based on multi-criteria clustering approach[J]. *Procedia Computer Science*, 2018, 131: 93-101.
- [13] Qu Y, Fang B, Zhang W, et al. Product-based neural networks for user response prediction over multi-field categorical data[J]. *ACM Transactions on Information Systems (TOIS)*, 2018, 37 (1): 1-35.
- [14] Sun Y, Wang W, Chambers J, et al. Two-stage monaural source separation in reverberant room environments using deep neural networks[J]. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2018, 27 (1): 125-139.
- [15] Cheng H-T, Koc L, Harmsen J, et al. Wide & deep learning for recommender systems[C]. *Proceedings of the 1st workshop on deep learning for recommender systems*, 2016: 7-10.
- [16] Hinton G E, Osindero S, Teh Y-W. A fast learning algorithm for deep belief nets[J]. *Neural computation*, 2006, 18 (7): 1527-1554.
- [17] Li Y, Fang S, Bai X, et al. Parallel design of sparse deep belief network with multi-objective optimization[J]. *Information Sciences*, 2020, 533: 24-42.
- [18] Zhang Y, Liu F. An Improved Deep Belief Network Prediction Model Based on Knowledge Transfer[J]. *Future Internet*, 2020, 12 (11): 188.
- [19] Guo H, Tang R, Ye Y, et al. DeepFM: a factorization-machine based neural network for CTR prediction[C]. *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 2017: 1725-1731.
- [20] Lian J, Zhou X, Zhang F, et al. xdeepfm: Combining explicit and implicit feature interactions for recommender systems[C]. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018: 1754-1763.
- [21] Wang R, Fu B, Fu G, et al. Deep & Cross Network for Ad Click Predictions[C]. *Proceedings of the ADKDD'17*, 2017: Article 12.
- [22] Carrano E G, Takahashi R H, Fonseca C M, et al. Nonlinear network optimization—an embedding vector space approach[J]. *IEEE Transactions on Evolutionary Computation*, 2009, 14 (2): 206-226.
- [23] Lilleberg J, Zhu Y, Zhang Y. Support vector machines and word2vec for text classification with semantic features[C]. *2015 IEEE 14th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC)*, 2015: 136-140.
- [24] Wei D, Chun M, Qing W. Autonomous Learning Recommendation Algorithm Based on K-Means and K-Nearest Neighbor[C]. *International Conference on Applications and Techniques in Cyber Security and Intelligence*, 2019: 1235-1244.
- [25] Cui Z, Xu X, F X U E, et al. Personalized Recommendation System Based on Collaborative Filtering for IoT Scenarios[J]. *IEEE Transactions on Services Computing*, 2020, 13 (4): 685-695.
- [26] Lu Q, Guo F, Zhang R. User-based collaborative filtering recommendation method combining with privacy concerns intensity in mobile commerce[J]. *International Journal of Wireless and Mobile Computing*, 2019, 17 (1): 63-70.
- [27] Li M, Wen L, Chen F. A novel Collaborative Filtering recommendation approach based on Soft Co-Clustering[J]. *Physica A: Statistical Mechanics and its Applications*, 2021, 561: 125140.
- [28] Zhang Z, Liu K, Wang W, et al. A personalized recommender system for telecom products and services[C]. *International Conference on Agents and Artificial Intelligence*, 2011: 689-693.
- [29] Zhang Z, Lin H, Liu K, et al. A hybrid fuzzy-based personalized recommender system for telecom products/services[J]. *Information Sciences*, 2013, 235: 117-129.
- [30] Wu D, Zhang G, Lu J. A fuzzy tree similarity based recommendation approach for telecom products[C]. *2013 Joint IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS)*, 2013: 813-818.
- [31] Rendle S. Factorization machines[C]. *2010 IEEE International conference on data mining*, 2010: 995-1000.
- [32] Zhang W, Du T, Wang J. Deep learning over multi-field categorical data[C]. *European conference on information retrieval*, 2016: 45-57.

- [33] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. Advances in neural information processing systems, 2012, 25: 1097-1105.
- [34] Pratiwi H, Windarto A P, Susliansyah S, et al. Sigmoid Activation Function in Selecting the Best Model of Artificial Neural Networks[C]. Journal of Physics: Conference Series, 2020: 012010.
- [35] Sharma S, Sharma S. Activation functions in neural networks[J]. Towards Data Science, 2017, 6 (12): 310-316.
- [36] Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.
- [37] Srivastava N, Hinton G, Krizhevsky A, et al. Dropout: a simple way to prevent neural networks from overfitting[J]. The journal of machine learning research, 2014, 15 (1): 1929-1958.
- [38] Fawcett T. An introduction to ROC analysis[J]. Pattern recognition letters, 2006, 27 (8): 861-874.
- [39] Bradley A P. The use of the area under the ROC curve in the evaluation of machine learning algorithms[J]. Pattern Recognition, 1997, 30 (7): 1145-1159.
- [40] Yang Z, Xu Q, Bao S, et al. When All We Need is a Piece of the Pie: A Generic Framework for Optimizing Two-way Partial AUC[C]. International Conference on Machine Learning, 2021: 11820-11829.
- [41] Zhang S, Yao L, Sun A, et al. Deep learning based recommender system: A survey and new perspectives[J]. ACM Computing Surveys (CSUR), 2019, 52 (1): 1-38.