Turkish Online Journal of Qualitative Inquiry (TOJQI)
Volume 12, Issue 7, July 2021: 8294 - 8302

Research Article

# Hyperparameter tuning in Deep Learning TechniquesforCollaborative Recommendation

SathiyaDeviS [a], ParthasarathyG [b]

[a]AssistantProfessor,AnnaUniversity,BITCampus,Tiruchirappalli,India
sathyadevi.2008@gmail.com
[b]AssistantProfessor, SRM TRP Engineering College, SRM Nagar,Tiruchirappalli,India.
parthasarathee.g@gmail.com

**Abstract**

Recommender systems are virtual devices that help people cope with information overload in a variety of fields. Customers are recommended items after a vast volume of data is analyzed to determine their preferences. Deep learning makes it possible to train models more precisely, which is challenging in a traditional environment. Later on, it necessitates a significant computational cost and has an impact on the success of big data appeals. In this article, we present a unique recommender scheme for Movielens data. It has a Multi-Layer Perceptron built in to handle large data sets and increase prediction accuracy by addressing data sparsity and scalability concerns. This dissertation focuses on the prediction model when dealing with this dataset. We discovered that our recommended model outperformed existing recommendation models when comparing Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE).

**Keywords**: activationfunction,deeplearning,parametertuning,prediction

## 1. Introduction

Machine learning is a branch of artificial intelligence that uses a large amount of data to make predictions. The Recommender System (RS) recommends music, movies, and other online products using machine learning techniques. Collaborative Filtering (CF) and Content-Based Filtering (CBF) are the two main types of this RS. CBF analyses the user and item profile from the details and recommends the products ,while CF builds a recommendation framework based on previous user ratings. The CF algorithm, on the other hand, generates accurate recommendations based on interactive data between users and objects, such as browsing, ranking, and clicks. Despite its simplicity and efficiency, the algorithm's output is constrained by the data's high sparsity. As a result, improved recommendation success necessitates the use of alternate approaches.

Deep learning is increasingly used in a variety of fields, including natural language processing, image and video processing, computer vision, and data mining. It's a promising research direction in the area of automated composite feature extraction at a high abstraction level. With activations such as ReLU, Sigmoid, and TanH, it can model nonlinear interactions in data. This increases the likelihood of capturing the complex and challenging user-item contact patterns. Deep learning can research the underlying explanatory factors and generate good representations from input data quickly and efficiently. In general, real- world applications have a wealth of descriptive data about objects and users. For sequential modeling tasks, deep learning is very useful. RNNs and CNNs are essential in tasks like machine translation, natural language comprehension, and speech recognition. Deep learning is a highly adaptable method of learning. TensorFlow, Keras, Caffe, MXnet, DeepLearning4j, PyTorch, and Theano are among the most common deep learning frameworks available today..

Thispaperestablishedadeeplearningmethodforasharedrecommendersystem (DLRS) that is independent of any additional knowledge other than user-iteminteraction.Incomparisonototraditionalmethods,DLRSproducessignificantly better results. The RMSE of the DLRS model is consistently lowerthan that of existing methods. The following is the outline for the rest of the paper:Section 2 covered relevant studies and the current state of a recommender scheme,while Section 3 covered deep learning preliminary work. Section 4 details theproposed scheme. Section 5 contains the experimental data and findings. Section 6concludeswithaconclusionandpossiblework.

## 2. Relatedworkandcurrentstateofrecommendersystem

Y. Hu, F. Xiong, and D. Lu et al. proposed a recommendation approach that incorporates implicit feedback, such as user comparisons for movie preferences, ratings, and positive manners, and uses it for collective filtering. They proposed a recommendation approach that factorizes both the explicit rating matrix and the implicit attitude matrix using multiplex implicit feedbacks. Another study suggested a novel hybrid recommendation algorithm that bridged the movie function and the user's interest to solve the two problems. The user rating matrix is combined with the user interest vector in this algorithm, and the movie attribute vector is generated based on the movie's attributes.

Deeplearninghasbecomeincreasinglypopularinrecentyearsasasolutiontoa variety of machine learning problems. Using the benchmark datasets, a latentfactor model for recommendation is suggested. It outperforms many collaborativefiltering techniques and proves to produce better results. The performance ofRecommenderSystemsdropsdrasticallywhenthedataissparse,soaCollaborativeDeepLearningusedneuralnetwork s'stringfeaturelearningcapability and model fitting robustness to solve the problem. When recommendersystems are confronted with a large amount of data, however, it makes modeltraining difficult to manage, and a variety of unpredictable issues arise. Deepcollaborative learning and parallelization methods were trained and optimized inparallel to improve the size and scalability of data in order to solve the problemsdescribed above. A new recommendation model is built using a CF approachand a deep learning neural network model to solve the Full Cold Start(CCS) andIncompleteColdStart(ICS)problems.Toextractthecontentfeaturesoftheobjects, a deep neural network known as SADE is used . Using, a compositeTag recommendation system (TagDC) is proposed that incorporates Deep learningandCollaborativefiltering.ThecollaborativefilteringmoduleandtheCNNcapsule module (TagDC-DL) (TagDC-CF). This model will generate a list of tagtrust probabilities and then rank the probabilities in the list to suggest TOP-Ktags.

Deeplearning-basedcollaborativefilteringaffectstheaccuracyoftherecommendation, according to the above literature study. Our proposed system isdescribed inthe followingsection.

## 3. Preliminaries

### 3.1 Introduction

For the collaborative recommender model, this section presents an effective(DLCRM)deeplearningapproach.TheDLCRMmodeltakesuserandmoviefeatures as input and predicts the rating score using a multi-layerperceptron basedArtificial Neural Networks (ANN) process. The fundamentals of neural networksare as follows. A neuron is the fundamental building block of a neural network.The weight is multiplied by the input as it reaches the neuron, and Bias is anotherlinear component that is added to the input. A nonlinear function is applied to theinputasthe activationfunction.InEq.1,the outputUisnowdefined.

$$U=f(x*W+b) \quad (1)$$

where x is the input, the weight is the weight, the bias is the bias,and the activation function is the activation function. The performance of a neuralnetwork is calculated by activation functions, which are nonlinear components.The role is attached to each neuron in the network and decides whether or not itshould be triggered based on whether or not the neuron's input is important to themodel'sprediction.(i)Sigmoid,(ii)Hyperbolictangent,(iii)ReLu,(iv)LeakyReLu,(v)ParameterisedRelu,(vi)Swis h,and(vii)Softmaxaresomeoftheactivationfunctions.TheSigmoidandTanhfunctionsaretwoofthefewactivation functions used in deep earning models to express features learned atintermediate layers. In both speech recognition and image classification tasks,linear rectifier functions (ReLU) proposed in were found to be efficient. Thisactivation feature literally masks out the negative half-plane, but interestingly,proper initialization mitigates these issues to a large extent. It is essentially thecommunity's dominant activation mechanism due to its simplicity and superiorefficiency. The leaky ReLU (LReLU) and parametric ReLU (PReLU), which wereshown to outperform original ReLU due to richer exploitation of negative planefeatures, discovered that simply dropping negative values could lose some details.In addition, a formed ReLU and a multi piece-wise linear rectifier activation wereinvestigated, and both showed superior output toReLU and its simplevariants.The activations mentioned above, on the other hand, all have a non-differentialdevice at some stage. Exponential Linear Unit (ELU), which smooths out theoutput slowly until a certain threshold is reached, tends to converge faster witheven higherperformance to overcome thisproblem.

Aside from the activation function, the loss function calculates the error for asingle training sample, and the cost function averages the loss function over theentiredataset.

**OptimizationofHyperparameters**

Presetting parameter values (hyperparameters) that govern how an algorithmlearns from data is often needed when fitting a machine learning model. Tuning oroptimisingthesehyperparametersbecomesaproblem ofselectinganoptimalmodelthatminimiseserrorandgeneralizeswelltounknowndata.Sinceit quickly obtains optimal values, the Bayesian Optimization algorithm is widelyused to calculate the optimal hyperparameter value of a model (Garrido-MerchánandHernández-Lobato, 2020). The Bayesian algorithm can fullymasterpriorknowledge with high robustness thanks to the use of the GP. This algorithm willmatch the posterior distribution of the objective function by increasing the numberof samples, resulting in the optimal value for model hyperparameter optimization.Hyperparametersarepronetomachinelearningmodels,andtheirevaluations are usually costly. Users urgently need intelligent methods to rapidlyoptimise hyperparameter settings to effectively promote machine learning models'efficiency within a restricted and small budget, according to known evaluationinformation .

**DeepLearning-BasedRecommendation**

Many deep learning frameworks are available, with support for languages suchas Python and C++, as well as the ability to build models. Keras is a well-knownneuralnetworks library written in Python. It supports both convolutional andrecurrentnetworksandcanrunonTensorFloworTheano.Kerashaseasy-to-understand and stable APIs, which are its biggest selling points. The TensorFlowworkflow is seamlessly integrated. Multiple deep learning backends, distributedtraining built-in, and multi-GPU parallelism are all supported. We chose MLP-based recommendation based on the above basics for our movie recommendationframework.Table1showsthedifferentdeeplearningrecommendationmodels.

**Table1**.DeepLearning-basedRecommendation

| Sl.No. | BasedRecommendation | Description |
|---|---|---|
| 1 | Multi-LayerPerception | Feed-forwardneuralnetwork,stackedlayerofnonlinear transformations. |
| 2 | AutoencoderBasedRecommendation | Unsupervisedmodel |
| 3 | ConvolutionalNeuralNetworks | feed-forward neural network withconvolutionlayersandpoolingoperations |
| 4 | RecurrentNeuralNetworks | modelingsequentialdata |
| 5 | RestrictedBoltzmannMachines | two-layerneuralnetwork |
| 6 | NeuralAttentionModels | differentiableneuralarchitecturesthatoperate based on soft content addressingoveraninputsequenceoraninputimage |
| 7 | NeuralAuto-Regressive | an unsupervised neural network built ontop of an autoregressive model and feed-forwardneuralnetworks |
| 8 | DeepReinforcementLearning | trial-and-errorparadigmandconsistsoffivecomponents(agents,environments,states,actions,andrewards |
| 9 | AdversarialNetworks | agenerativeneuralnetworkwhichconsistsofadiscriminatorandagenerator |

| | | |
|---|---|---|
| 10 | DeepHybrid | goodflexibilityofdeepneuralnetworks |

**ProposedAlgorithm**

The first phase in our proposed model is explained in Algorithm 1. Ratings andmovie item details are combined and used as data. The prediction model is basedonthemulti-layerperceptronmodel,andhyperparameter tuningiscarried out.

**Algorithm1:**MLPBDL

**Input:**IntegratedDataSet

**Output:**PredictionMeasures

*Step1* : *FeatureEngineeringonInput*

*Step2* : *PerformHyperParameterTuning*

*Step3* : *RandomSplitforTrainingandTestingdata set*

*Step4* **:** *ModelcreationusingMLP*

*Step5* : *Performprediction*

*Step6* : *End*

## 4. ProposedSystem

Inthispaper,theproposeddeeplearning-basedrecommendationmodelhavingdifferentphases.(i)Problemdefinition,(ii)Preprocessing,(iii)Parametertuning,(iv)Modelcreation,and(v)Prediction.Figure1showstheblockdiagramofourproposedmodel,and thefunctionalitiesofeachphaseareexplained below.

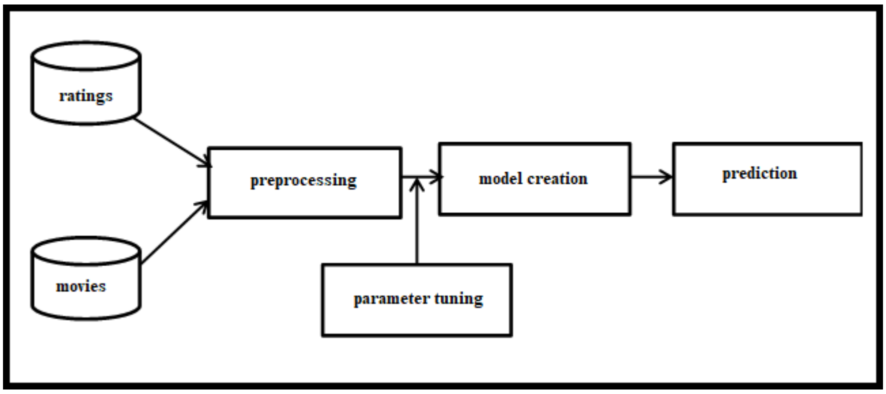


**Figure 1**.ProposedDeepLearning-basedRecommendationModel

**ProblemDefinition**

Let U1,u2,...un represent a set of users, and i1,i2,... represent a set of objects.The user u gave ratings to products I based on his personal preferences. Theunrated things can be predicted using user interests that are similar. Various filesareusedtocombinethevariousfeatures.Itismadeupofthreesteps:(ii)preprocessing,(ii)model formation,and(iii)prediction.

**Preprocessing**

The raw dataset is filthy and needs to be cleaned up before it can be used. Itcontainsincomplete,noisy,andinconsistentdata;incompletedatareferstoattribute values that are missing. Because of errors or outliers, the data becomesnoisy. Due to differences in name or code, the data is variable. Data cleaning,integration, reduction, and transformation are four preprocessing techniques. Datacleaning entails filling in missing values, data integration entailsmerging thedifferent attributes in the dataset, reduction entails dimensionality reduction anddatacompression,anddatatransformationentailstransformingorconsolidatingthedataintoacceptableformsformining.Weusedfeaturecombination andfeaturecleaningto preprocessthedataset inour experiment.

Therawdatasetisdividedintoseveralfiles,eachwithitsownsetofcharacteristics.Otherfeaturescanbecombinedinasinglefilecreatedwithrelational algebra. This integrated file allows us to obtain all of the item and user-relatedinformation.Therearemissingvalues,outliers,andanomaliesinthecombinedfeatures.Missingvalues,outliers,andcontradictorydataareallrevealed through exploratory data analysis (EDA). The mean or median of non-missing values in a column is used to measure the missing value. Another optionforfilling missingvaluesistousethemostcommon valueinthecolumnorzero.

**ParameterTuning**

The problem of selecting a set of suitablehyperparameters for alearningalgorithm is known as hyperparameter optimization or tuning in machine learning.Hyperparameter optimization identifies a tuple of hyperparameters that results inan optimized model that minimizes a predefined loss function on independent data.Table 2 lists the parameters for our proposed algorithm using the grid searchalgorithm,whichhasundergonetwotypesofhyperparametertuning.Thehyperparameterscanbefoundusingavarietyofmethods,includingmanualsearch, grid search, random search, and Bayesian optimization. We used a Gridsearchtofindthehyperparameterforourmodelamongthem.Amodelhyperparameterisaconfigurationthatisn'tpart ofthemodelandcan'tbeestimated from data. They're often used in processes to aid in the estimation ofmodelparameters,whichareusuallyteninnumberanddeterminedbythepractitioner. Heuristics are often used to set them. They're often fine-tuned for aspecificpredictivemodelingissue**.**

**Table2.**Hyperparametersettings

| ParameterName | Setting |
|---|---|
| Learningrate | 0.2 |
| Momentum | 0.4 |
| Numberofepochs | 10 |
| No.ofHiddenLayers | 3with30Neuronsineachlayer |
| Activationfunction | tanh |
| Solver | adam |
| BatchSize | 10 |

Table 2 lists hyperparameters tuned for our experiment. For each parameter, thevalues are given as input, and the value is chosen based on the best test score fortheparticularparameter.

**ModelCreation**

We have chosen Multi-Layer Perceptron-based prediction model among thedifferentdeeplearning-basedrecommendersystemsexplainedinsection3.Figure1explainsourproposedMulti-LayerPerceptron basedDeepLearningalgorithm.

**Prediction**

Prediction is the process of calculating the unknown target value using thedependentvariablesinthe input.heproposed prediction model algorithm2.

For prediction, Mean Absolute Error(MAE) is used and represented as the differ-ence between the predicted rating of user *u* on item *i*($p_{u,i}$) and the actual rating ofuser*u*onitem*i*($r_{u,i}$). andisrepresentedinEq. 2.

$$MAE = \frac{1}{N}\sum_{u,i} | p_{u,i} - r_{u,i} | \quad (2)$$

Eq. 3 shows root mean squared error, whichis the standard deviation of theresiduals. Residuals are measures of how far from the regression line data pointsare plotted.

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{\left(y_i - \overline{y_i}\right)^2}{n}} \quad (3)$$

## 5. ExperimentandResults

WeusedtheMovielens100KdatasetfromGroupLensLabintheexperiments. We also expand our experiments to the wider Movielens-1 M datasetfor retesting and verification to ensure the algorithm's feasibility and robustness.

Each dataset is randomly divided into a training set and a test set prior to theexperiments. Movielens is a popular video-sharing website that keeps track ofusers' movie ratings. Table 3 shows the specifications of the Movielens 100K and1M. The Movielens 100K dataset has 100,000 ratings from 943 users on 1682items, while the Movielens 1M dataset has 6040 users and 3952 items. There are1,000,209 documentsinthe dataset.

**Table3.**MovielensDataset Description

| Dataset | Users | Items | Records |
|---|---|---|---|
| 100K | 943 | 1682 | 100,000 |
| 1M | 6040 | 3,900 | 1,000,209 |

For datasets, we use 80 percent of the data as the training set and the remainingdataasthetest set.Werepeat theexperiments ten times in arow,with theaveragevalues serving as the final results. We use two common assessment metrics toverify the accuracy of predicted outcomes. The first is the Mean Absolute Error(MAE), which is found in Eq (1). The other is the RMSE (Root Mean SquareError), which is found in Eq (2). A lower MAE or RMSE value indicates that themodel produces more reliable performance. For our convenience, features fromdifferent files have been combined. Preprocessing operations are completed, anddifferentfeaturesare analyzedinFig. 2.

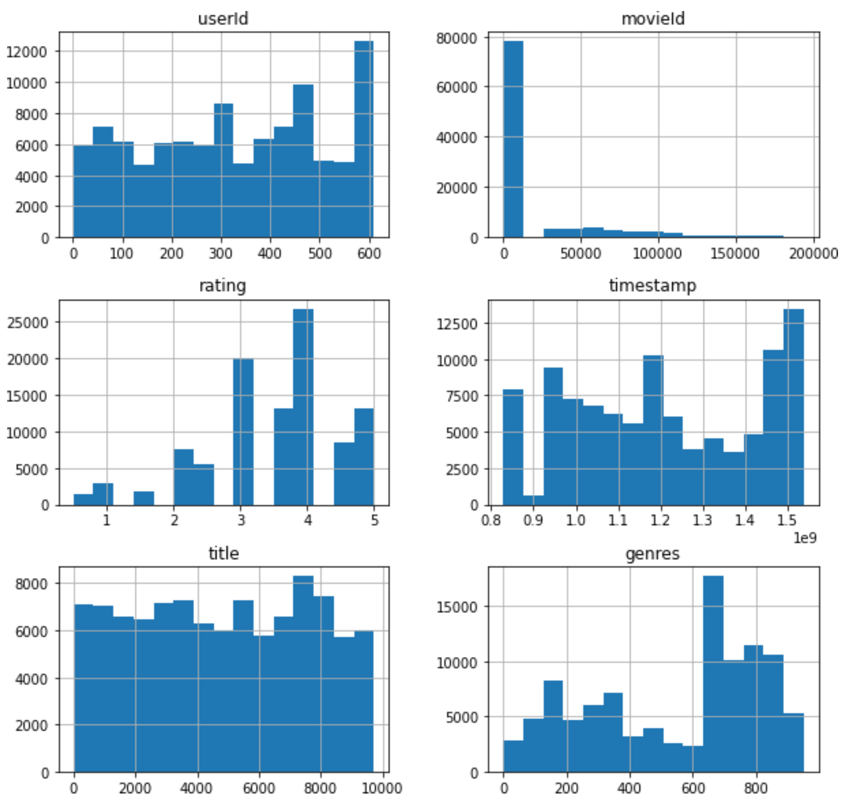


**Figure2.**FeatureAnalysis

Itshowsthedistributionoffeaturesamongthese1,00,0000records.Theobservation made here is among the ratings given by the 943 users and 1682 movies 4 scale rating occupies 34.8%,3 scale rating occupies 26.1% and 22.6%, 10.7% and 5%.

Figure 3 shows the tuning of momentum and learn rated using the test scoreaverage.The grid search takes the series of values and found the best value foreachparameter.


**Figure3.**Hyperparametertuning

The tuned parameters are taken for our model creations. The available training setis created using a multi-layer perceptron algorithm with 3 hidden layers with 30neurons. The model performances are measures using the test dataset. The twoperformance measures are calculated using Eq.1 and Eq.2and listed in Table 4.OurProposedmethodiscomparedwiththe existingstate-of-the-art.

**Table4.**PredictionPerformance

| Dataset | Metric | Proposed | Existing |
|---|---|---|---|
| 100K | MAE | 0.7742 | 0.8453 |
| 1M | RMSE | 0.9883 | 1.0549 |

Figure4showstheevaluationmetricoftheexistingandproposedmodel.Ourdeep learning model performs better while applying 100 K dataset. The 1M resultsshowthat bothMAE and RMSEare lagging withtheexisting.

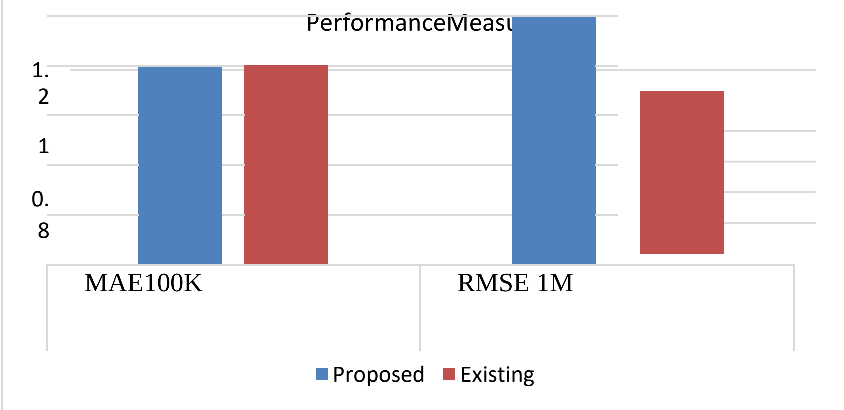


**Figure 4.**Comparisonofperformance measures

The prediction measures of our proposed model are shown in Figure 4. Themean absolute error of the proposed model is 7.1% lower than the existing model.The RMSE value of the proposed model is 6% is reduced when comparing witht he existing work. This comparison shows our deep learning-based prediction model performance is better than the existing one.

## 6. ConclusionandFutureWork

Collaboration filtering (CF) is a crucial component of recommendation systemdesign and development, according to the literature. Its drawbacks include datasparsity, scalability, and the integral existence of data. A deep learning model for acollaborativerecommendermodelisproposedinthispaper(DLCRM).Theproposed deep learning approach wascompared to current methods in a report.Our findings show that the proposed method out performs current methods, demonstrating that incorporating a deep learning approach into a recommenderframework is a worthwhile endeavor. On 100k and 1M MovieLens datasets, wetested the model.We plan to expand to another deep learning approach for recommender systems, such as auto encoder,infutureworkandimprovetheoutputevenfurther.

## References


[1] Arash Khomeini, SamanHaratizadeha,EhsanHoseinzade(2020).Representation Extraction and Deep Neural Recommendation for Collaborative Filtering,21- 23.

[2] AanchalMongia ,NehaJhamba , Emilie Chouzenoux , AnshulMajumdar(2020). Deep latent factor model for collaborative filtering,Signal Processing 169.

[3] Andrew L. Awni Y. Hannun, Andrew Y. Ng(2013). Rectifier Nonlinearities Improve Neural Network Acoustic Models, Proceedings of the 30 the International Conference on Machine Learning, Atlanta, Georgia, USA.

[4] Can Li,LingXu,MengYan,YanLei,TagDC(2020).A tag recommendation method for software information sites with a combination of deep learning and collaborative filtering,The journal of systems and software 170.

[5] Charu C. Aggarwal(2016).Recommender Systems, Springer,71-224.

[6] FrancescoRicci,LiorRokach,BrachaShapira(2011).RecommenderSystemsHand-book,Springer,1-35.

[7] F. Yang, H. Wang and J. Fu (2020): Improvement of recommendation algorithm based on Collaborative Deep Learning and its Parallelization on Spark, Journal of Parallel and Distributed Computing.

[8] Hang Li(2018), Deep learning for natural language processing: advantages and challenges, National Science Review, Volume 5, Issue 1,24–26.

[9] Hu, Y., Xiong, F., Lu, D., Wang, X., Xiong, X., and Chen, H (2019).Movie Collaborative Filtering with Multiplex Implicit Feedbacks. Neuro computing.

[10] Jian Wei, Jianhua He, Kai Chen, Yi Zhou, ZuoyinTang(2016):Collaborative Filtering and Deep Learning Based Recommendation System For Cold Start Items, Expert Systems With Applications.

[11] Jing Li, WentaoXu, Wenbo Wan, Jiande Sun(2018), Movie Recommendation Based on Bridging Movie Feature and User Interest, Journal of Computational Science.

[12] Mohammed Fadhel Aljunid1, Manjaiah DH (2020).An Efficient Deep Learning Approach for Collaborative Filtering Recommender System, Third International Conference on Computing and Network Communications (CoCoNet'19), ProcediaComputer Science 171, 829–836.

[13] Najafabadi, M. M., Villanustre, F., Khoshgoftaar, T. M., Seliya, N., Wald, R, and Muharemagic, E.(2015).Deep learning applications and challenges in big data analytics. Journal of Big Data.

[14] Nguyen, G., Dlugolinsky, S., Bobák, M. (2019).Machine Learning and Deep Learning frameworks and libraries for large-scale data mining: a survey, ArtifIntell Rev 52, 77–124.

[15] Terada, Y., & Hirose, R (2020).Fast generalization error bound of deep learning without scale invariance of activation functions. Neural Networks, 129, 344– 358.

[16] Y. Hu, F. Xiong and D. Lu (2019).Movie collaborative filtering with multiplex implicit feedbacks, Neurocomputing.

[17] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay(2018): Deep Learning based Recommender System: A Survey and New Perspectives. ACM Comput. Surv. 1, 1, Article 1, 35 pages.

[18] F. Maxwell Harper and Joseph A. Konstan(2015): The MovieLens Datasets: History and Context. ACM Transactions on Interactive Intelligent Systems (TiiS) 5, 4, Article 19 , 19 pages.

[19] Zhi Chen, Pin-Han Ho(2021) : A generic shift-norm-activation approach for deep learning , Journal of pattern recognition, 10 pages.

[20] Xavier Glorot Antoine BordesYoshuaBengio(2011) :Deep Sparse Rectifier Neural Networks ,Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS) ,315-323.

[21] Kaiming He Xiangyu Zhang ShaoqingRenJian Sun (2015):Delving Deep into RectifiersSurpassing Human-Level Performance on ImageNet Classification, arXiv:1502.01852v1 [cs.CV] ,11 pages.

[22] Xiaojie Jin, ChunyanXuJiashiFeng, Yunchao Wei Junjun Xiong, Shuicheng Yan (2016): Deep Learning with S-shaped Rectified Linear Activation Units, Associa- tion for the Advancement of Artificial Intelligence ,7 pages,(2016).

[23] Zhi Chen, Pin-Han Ho : Deep Global-Connected Net With The Generalized Mul- ti-Piecewise ReLU Activation in Deep Learning, arXiv:1807.03116v1 [cs.CV], (2018).

[24] Bjork-Arne Clevert, Thomas Unterthiner&SeppHochreiter(2016), “Fast And Accurate Deep Network Learning By Exponential Linear Units (Plus)”, Proceedings of ICLR ,14 pages.

[25] William W. Tsao,b , BarisBurnaka,b , Efstratios N. Pistikopoulos(2020): HY-POP: Hyperparameter optimization of machine learning models through parametric programming,Journal of Computers and Chemical Engineering,(2020).

[26] D. Sun, J. Xu, H. Wen (2020): Assessment of landslide susceptibility mapping based on Bayesian hyperparameter optimization: A comparison between logistic regression and random forest, Engineering Geology (2020), https://doi.org/10.1016/j.enggeo.2020.105972

[27] C. Wang, H. Wang, C. Zhou (2020): Experience Thinking: Constrained Hyperparameter Optimization based on knowledge and pruning, Knowledge- Based Systems, doi: https://doi.org/10.1016/j.knosys.2020.106602.

[28] Li, H., Cui, J., Shen, B., & Ma, J. (2016).An intelligent movie recommendationsystem through group-level sentiment analysis in microblogs. Neurocomputing,210,164–173,.doi:10.1016/j.neucom.2015.09.134.