International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. #, No.# (Mon-20..)

A Deep Context-Based Factorization Machines for Context-aware Recommender Systems

Rabie MADANI [1], Abderrahmane EZ-ZAHOUT[1] and Fouzia OMARY[1]

[1] Intelligent Processing and Security of Systems Team,
Computer Sciences Department
Faculty of Sciences, Mohammed V University
Rabat, Morocco

E-mail address: rabie.madani@um5.ac.ma, abderrahmane.ezzahout@um5s.net.ma, omary@fsr.ac.ma.

Received ## Mon. 20##, Revised ## Mon. 20##, **Accepted** *## Mon. 20##, Published ## Mon. 20##*

Abstract: Context-aware recommender systems (CARS) aim to offer personalized recommendations by incorporating user contextual information through analysis. By analyzing these contextual cues, CARS can better understand the preferences and needs of users in different situations, thereby improving the relevance and effectiveness of the recommendations they provide. However, integrating contextual information such as time, Location and into a recommendation system presents challenges due to the potential increase in the sparsity and dimensionality. Recent studies have demonstrated that representing user context as a latent vector can effectively address these kinds of issues. In fact, models such as Factorization Machines (FMs) have been widely used due to their effectiveness and their ability to tackle sparsity and to reduce feature space into a condensed latent space.

In this article we introduce a Context-aware recommender model called Deep Context-Based Factorization Machines (DeepCBFM). The DeepCBFM combines the power of deep learning with an extended version of Factorization Machines (FMs) to model non-linear feature interactions among user, item, and contextual dimensions. Moreover, it addresses certain limitations of FMs, in order to improve the accuracy of recommendations. We implemented our method using two datasets that incorporate contextual information, each having distinct context dimensions. The experimental results indicate that the DeepCBFM model outperforms baseline models and validates its effectiveness.

Keywords: ***Recommender systems, Context Aware Recommender Systems, Factorization Machines, Context-Based Factorization Machines, Deep Learning, Deep Neural networks***

1. INTRODUCTION

In light of the vast and ever-expanding array of products and services available today, Recommender Systems (RS) have become indispensable across a range of sectors, including ecommerce, video content, cinema, travel, and, notably, the gaming industry. RS belong to the category of data filtering systems, and their primary function is to provide user-specific recommendations based on individual preferences [1]. The adoption of RSs offers two significant advantages: firstly, they streamline the user experience by minimizing the time and effort required to find desired items, and secondly, they contribute to increased company sales and revenue generation. Classical Recommender Systems primarily rely on just two key dimensions: the user dimension and the item dimension, to offer personalized recommendations. In the content-based approach [2], a more detailed strategy is employed. This method incorporates both item features and user profiles in the recommendation process. By analyzing the characteristics of items and the preferences and behavior of users, content-based systems provide recommendations that are tailored to individual users based on the content. Conversely, Collaborative filtering approaches [3] take a different route. They don't require detailed information about users or items. Instead, they rely on user ratings to gauge a user's preference for a particular item. However, these approaches have limitations because they overlook various factors that can impact a user's preferences, such as the contextual factor. For example, a father's movie preference may shift based on whether he's watching alone or with his kids, with context (like the presence of companions) influencing his choices.

E-mail:author's email

Recommendation systems often face two primary challenges: sparsity and high dimensionality in the data. Sparsity arises when there are limited interactions or ratings between users and items, while high dimensionality results from the addition of contextual information, which effectively increases the number of dimensions in the dataset. Paradoxically, incorporating contextual data can exacerbate the sparsity issue by introducing more dimensions to the data, further complicating the recommendation process. Sparsity refers to the situation where there are significant missing data points in a dataset, while high dimensionality results from the addition of new features, which effectively increases the number of dimensions. Paradoxically, incorporating contextual data can exacerbate the sparsity and dimensionality issues by introducing more dimensions to the data, further complicating the recommendation process. However, these approaches have limitations because they overlook various factors that can impact a user's preferences, such as the contextual factor. For example, a father's movie preference may shift based on whether he's watching alone or with his kids, with context (like the presence of companions) influencing his choices.

Factorization Machines FM [4] which is a supervised algorithm effectively resolves aforementioned problems and delivers impressive performance. This method involves the transformation of users and items into a latent space, creating a low-dimensional representation of the data. The majority of research in the field of Context-aware Recommender Systems (CARS) [5] has been dedicated to improving and advancing Factorization Machines. One of the primary drawbacks of FM is its reliance on a fixed interaction function, often an inner product, to estimate the intricate interactions between User and Item. This fixed function may not adequately capture the complexity and nuances of real-world user-item relationships. This can result in less accurate recommendations compared to more advanced techniques that can better model the complexities of user-item interactions. Moreover, FM employs a single latent vector even when dealing with features originating from distinct contextual dimensions or contexts. This can be a limitation of FM, since it tends to overlook the fact that features may exhibit distinct behavior when interacting with features from separate contexts. This lack of context-awareness can limit FM's ability to capture the nuanced interactions that occur between features across different contexts. The main contributions proposed by the present work are:

- New Context-Aware Recommender Model: The main contribution is the introduction of a new context-aware recommender model. This model combines Deep Neural Networks (DNNs) and Factorization Machines (FMs). This hybrid approach likely aims to leverage the strengths of both DNNs and FMs to improve the accuracy and relevance of recommendations. Context-aware recommender systems consider additional information, such as user context or real-time data, to make personalized recommendations.

- Exploiting Deep Learning Techniques: The second contribution is the utilization of deep learning techniques to capture high-order nonlinear feature interactions. This is important because deep learning models are known for their ability to capture complex patterns and relationships in data. By doing so, the proposed model can address some of the limitations associated with traditional Factorization Machines, which may struggle with modeling higher-order interactions.

- New Variant of FM for Context-Aware Recommender Systems: The final contribution is the introduction of a new variant of Factorization Machines specifically adapted to Context Aware Recommender Systems (CARS). This variant is designed to capture the differences between different contexts and also to capture low order feature interactions.

The rest of this paper is structured as follows. Section two reviews related works. Section three presents the proposed model (Deep Context-Based Factorization Machines model). Section four analyses results. Finally, the last section presents a conclusion of the realized work.

2. RELATED WORKS

Recently, many studies have been presented to enhance the exactness and the efficiency of recommenders, either by improving FM, which is a reference algorithm, by exploiting the strengths of deep learning or by using new methods. Cheng et al. [6] presented a Wide and deep learning model, that uses the wide linear model to memorize the interaction of features and deep learning DNNs for feature generalization. The model was evaluated using PlayStore, the outcomes demonstrate that the model increased the acquisitions of apps. Guo et al. [7] presented the DeepFM model, which combines the power of FMs and DNNs to improve recommendation performances with less manual feature engineering work. Both components of the model were trained jointly in order to gain in terms of performance and also to capture high order interactions.

Xiao et al. [8] presented an Attentional FM model, which captures the importance of feature interaction using neural attention networks. The model tries to enhance FM by improving the interpretability and the representation ability of a FM algorithm.

Chen et al. [9] introduced the Compressed Interaction Network(CIN) than combined it with deep learning to create a consolidated model. The model aims to automatically learn the interaction between features in an explicit way and avoid feature engineering.

Song et al. [10] proposed a CTR model using a self-attentive neural network called AutoInt. It learns automatically high order by allowing the interaction between features for relevance determination.

Yu et al. [11] presented a Input aware Factorization Machine to enhance FMs by considering the inputs influence on feature representations. The model uses neural networks to learn the input factors of each features in several instances. The model aims to enhance the power of predication while keeping the linear complexity of the traditional FM.

Pan et al. [12] presented Field-weighted FM for recommendation in display advertising field. The model is an extension of FM that aims to capture the importance of interaction of different pairs of fields in reasonable complexity time.

Trigeorgis et al. [13] presented a deep MF to learn attribute representations. The model uses a semi Non-Negative Matrix Factorization algorithm to model representations of low dimensions.

Lara-Cabrera et al. [14] proposed a collaborative filtering Recommender Systems that exploits DNNs and FMs to enhance the accuracy of recommendations. The method uses a deep learning paradigm not to capture high order features but to improve the MF model.

Ez-Zahout et al. [15] presented a hybrid movie Recommender System based on Matrix Factorization and KNN. The model provides recommendation to a user by computing the similarity between movies and generating top k movies. While prior research has demonstrated strong results in terms of accuracy, performance, and interpretability in recommendation systems, they have generally overlooked the significant impact of contextual factors on user preferences. In response to this limitation, other researchers have made efforts to incorporate contextual data into the recommendation process with the aim of achieving improved outcomes, based on Matrix Factorization and Factorization Machines.

For instance, Baltrunas et al. [16] proposed a context-aware recommendation algorithm that builds upon Matrix Factorization. This algorithm takes into account the interplay between contextual factors and item ratings by introducing extra model parameters. One key benefit of this solution is its lower computational overhead, and it also offers the flexibility to depict the interaction between context and items at various levels of detail or granularity.

In their work, Madani and Ez-zahout [17] proposed a Context-Aware Recommender System model that utilizes a Bidirectional Encoder Representations from Transformers (BERT) pretrained model for personalized Named Entity Recognition (NER). The model enables the automatic extraction of contextual information. Furthermore, the authors adapted traditional Factorization Machines to accommodate contextual information, enhancing their capability to provide accurate rating predictions.

Casillo et al. [18], proposed a Context-Aware Recommender System that leverages embedded context likely involves integrating contextual information directly into the recommendation process to enhance the system's performance. In this work, the concept of embedded context suggests that instead of treating contextual information as an external or additional input to the recommendation model, the system incorporates this context within the model itself. Moreover, the method leverages matrix factorization's computational efficiency to address scalability issues. While Matrix Factorization and Factorization Machines offer advantages, they encounter challenges when it comes to capturing the intricacies of non-linear feature interactions in the learning process. Numerous researchers have endeavored to harness the capabilities of deep learning in crafting more intricate context-aware recommender models, aiming to address the complexities posed by non-linear problems in recommendation systems.

Jeong et al. [19] introduced a context-aware recommender system leveraging deep learning techniques. This approach considers contextual features to enhance recommendation accuracy. The model integrates a neural network and autoencoder, leveraging established deep learning architectures. Through this combination, the model effectively extracts distinctive features and forecasts scores during the input data restoration process. Notably, the proposed model exhibits versatility in accommodating diverse contextual information types.

Sattar and Bacciu [20] introduced a Context-Aware Graph Convolutional Matrix Completion method. This approach captures structural details within a graph while incorporating user opinions on items, surrounding contextual information represented on edges, and static features associated with user and item nodes. Their graph encoder generates user and item representations by considering context, features, and opinions. These representations are aggregated and fed into the decoder, which predicts ratings.

VU et al. [21] introduced an innovative strategy that leverages deep learning for the development of context-aware multi-criteria recommender systems. In their approach, deep neural network (DNN) models play a pivotal role in predicting context-aware multi-criteria ratings and learning the aggregation function. The study effectively demonstrates the incorporation of contextual information into Multi-Criteria Recommender Systems (MCRSs) through the application of DNN models. Specifically, they showcased the utility of DNN models in forecasting context-aware multi-criteria ratings and acquiring insights into the aggregation function.

Previous research heavily relied on deep learning as the primary method for generating recommendations but failed to adequately address the challenges of sparsity and high dimensionality in the data, which is the case of real-world datasets. Unlike prior studies, our objective is to introduce a pioneering context-aware recommender system that specifically tackles the shortcomings of Factorization Machines (FMs) by harnessing the capabilities of Deep Neural Networks. Additionally, our aim is to unveil an evolved FM variant optimized for CARS. This refined FM version excels in capturing intricate feature interactions, spanning across both low and high orders. More detailed insights into the DeepCBFM model will be elaborated upon in the subsequent section.

3. PROPOSED MODEL

Our primary aim is to create an advanced Recommender system capable of integrating contextual information into the recommendation process. In pursuit of this goal, we introduce the Deep Context-Based Factorization Machines Model (DeepCBFM), a fusion of Factorization Machines (FM) and deep learning techniques.

Illustrated in figure 1 is the model architecture comprising two simultaneous components: the primary CBFM component and the secondary deep component. The CBFM segment extends the functionality of Factorization Machines (FM) to capture second-order interactions, while the deep component models higher-order feature interactions. Both segments utilize common input and embedding layers, and the model's ultimate prediction results from the combined outputs of these two components. This section provides a comprehensive breakdown of each component within the proposed model.

This section discusses each component of the proposed model in detail.

A. *CBFM component*

In previous works, Linear regression (LR) model has been widely employed in ratings prediction [22] [23]. To predict ratings LR uses a linear combination of features as shown in the equation below:

$$y_{LR} = w_0 + \sum_{i=1}^{d} w_i\, x_i \qquad (1)$$

However, the LR model does not perform well since it does not take into account the interaction between features, which is crucial. Poly2 models [24] tackle this problem by adding order-2 feature interactions to the above equation, which gives as a result:

$$y_{Poly2} = w_0 + \sum_{i=1}^{d} w_i\, x_i + \sum_{i=1}^{d} \sum_{j=i+1}^{d} x_i\, x_j w_{h(i,j)} \qquad (2)$$

However, it is clear that this method suffers from some drawbacks. For instance, the interaction parameters of features can be trained only when these features appear in the same record, which means that unseen features will have insignificant predictions. The FM model outperforms Poly2 especially when the model deals with sparse data. FM calculates interactions between two features via the

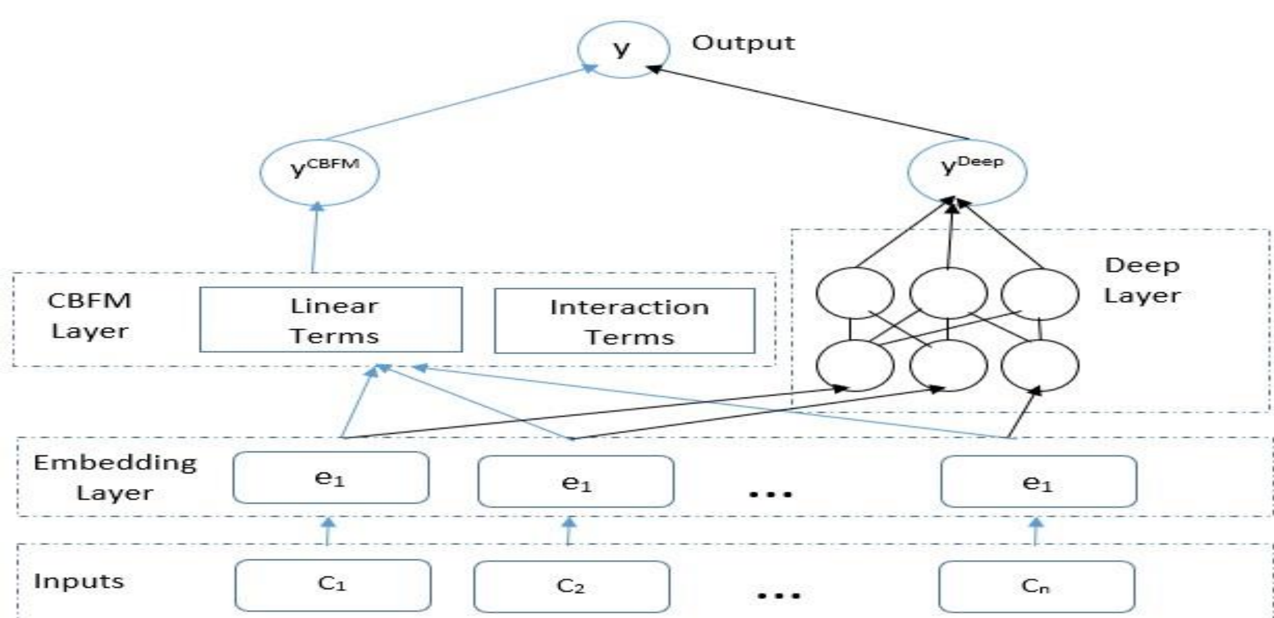

Fig. 1. The architecture of the DeepCBFM.

dot product of their corresponding latent vectors. FM equation is defined as follows:

$$y_{FM}(x) = w_0 + \sum_{i=1}^{d} w_i x_i + \sum_{i=1}^{d}\sum_{j=i+1}^{d} < v_i, v_j > x_i x_j \quad (3)$$

FM can train embedding vector $v_i(v_j)$ even if it never or rarely appeared in the data. To learn the effect of latent between features, FM uses only one latent vector even for features from different contexts. For instance, when computing interactions among three contexts (Day, Companion, and Mood), FM utilizes the same embedding vector for Monday to capture its latent effects when paired with Companion $< V_{Monday}, V_{Son} >$ and also with Mood $< V_{Monday}, V_{Happy} >$, despite these contexts being distinct. This approach overlooks the nuanced behavior of features when they interact across different contexts.

To verify this observation, we utilize ANOVA, a statistical method developed by Ronald Fisher in the early 20th century. ANOVA is employed to detect and showcase potential similarities or differences in specific aspects within a studied population through variance analysis. The formula of ANOVA is defined as follow:

$$F = \frac{MST}{MSE} \quad (4)$$

Where, F is the ANOVA coefficient, MST is the mean sum of squares due to treatment and MSE the mean sum of squares due to error.

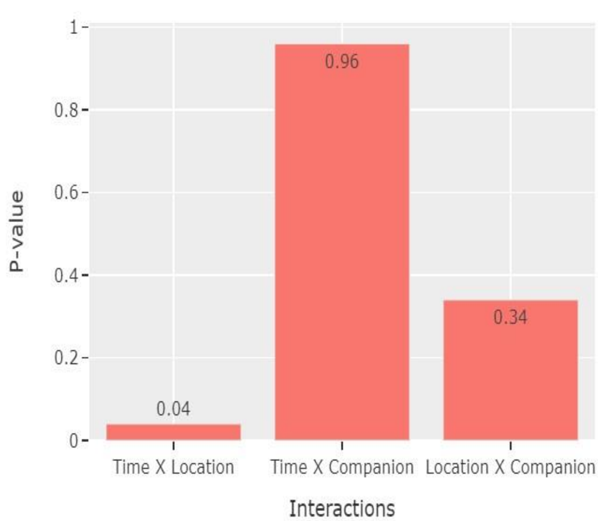

Fig. 2. Interaction between contexts.

We do not use ANOVA method to verify the existence of relationship between two features, but simply to capture the difference of features interaction strength. We conduct this experiment on DePaulMovie dataset [26]. Figure 2 shows the obtained results from the interaction of three contexts namely Time, Location and Companion using ANOVA two-way. This statistical tool helps us to compute the interaction strength between two contexts (Ci, Cj) and an outcome Y.

As we can see, the interaction between features is expressed in P-value, when the P-value is less than 0,05 it means that the interaction is significant, otherwise, there is no significant interaction between features. For the first interaction between Time and Location the P-value is 0.04 and since the obtained value is less than 0.05, the interaction effect between these two contexts is significant. For the second interaction (Time, Companion) and the third interaction (Location, Companion), the obtained values are higher than 0.05 which means that, there is no significant interaction between these two pairs of contexts.

The primary conclusion drawn from this experiment is that variables exhibit varying interaction patterns when paired with variables from distinct contexts. Addressing the limitations of FM highlighted earlier, we introduce an enhanced version termed CBFM. This extension of FM incorporates extra weights to discern variations between contexts and to distinguish the latent vectors of a feature when it interacts with other features from different contexts. Within our model, the primary goal of the CBFM component is to capture second order feature interactions while accommodating contextual differences. The equation of CBFM is defined as follow:

$$y_{CBFM}(x) = w_0 + \sum_{i=1}^{d} w_i x_i + \sum_{i=1}^{d}\sum_{j=i+1}^{d} < v_i w_{c(i)}, v_j w_{c(j)} > x_i x_j \quad (5)$$

Where w_0 and w are respectively the bias term and weights of feature vectors, v_i and v_j the latent vectors of $feature_i$, $feature_j$, $w_{c(i)}$ and $w_{c(j)} \in R$ are weights to capture the importance between $context_i$ and $context_j$.

B. Deep component

To model nonlinear feature interactions, we present a feed forward neural network [7]. The input of the deep layer is the concatenation of the feature embedding vectors which is denotes as:

$$s^{(0)} = [e_1, e_2, e_3, \dots, e_m] \quad (6)$$

Where m the number of contexts, e_i the embedding feature of i^{th} feature. Embedding outputs are fed in the neural network which is described as follow:

$$s^{(k)} = \alpha\left(W^{(k)}s^{(k-1)} + b^{(k)}\right) \quad (7)$$

α is the activation function, k the layer depth, W(k) the weight of the k(th) layer, b(k) the bias at the kth layer and s(k) the k(th) layer output. let y(DNN) be the deep layer output, the final prediction of DeepCBFM model is:

$$y = \sigma\left(y^{(CBFM)} + y^{(DNN)}\right) \quad (8)$$

Where σ is the sigmoid function.

4. EXPERIMENT AND RESULTS

A. Datasets

To evaluate the DeepCBFM, two datasets provided by CARSKit [25] are used:
The first is DePaulMovie dataset includes 5043 ratings collected from 123 users and contains three different contextual dimensions (Companion, Time and Location). The Companion dimension has three features (Alone, Partner and Family), the Time dimension has two features (Weekday, Weekend), the Location has two features (Cinema, Home). The second is InCarMusic dataset includes 4013 ratings from 42 users. The dataset contains eight contextual dimensions (Driving Style, Mood, Landscape, Weather, Natural Phenomena, Roadtype, Sleepiness and TrafficConditions) and a total of 26 features. Table 1 shows more detail about the two datasets.

TABLE 1: STATISTICS ABOUT THE TWO DATASETS.

	DePaulMovie	InCarMusic
Users	123	42
Items	79	139
Ratings	5043	4013
Dimensions	3	8
Sparsity	94%	99%

B. Evaluation Measures

To check the DeepCBFM model performance the RMSE and R-Squared are used:

Root Mean Squared Error (RMSE) is one of the largest used metrics for regression problems. It is the MSE square root which is calculated as the squared differences between actual the target values.

$$RMSE = \sqrt{\frac{1}{N}\sum_{j=1}^{m}(y_j - \hat{y})} \quad (9)$$

y_j is the real value and $\widehat{y_j}$ is the target value.

R-squared (R^2) is a statistical measure that shows how close the data are to the regression line.

$$R^2 = 1 - \frac{SumSquaredRegression(SSR)}{TotalSumofSquares(SST)} = 1 - \frac{\sum(y_i - \hat{y})^2}{\sum(y_i - \bar{y})^2} \quad (10)$$

Where SSR is the sum square of the difference between real and predicted variables, SST is the total of sum squares and $\bar{y}$ is the mean of all values.

B. Performance Comparisons

To confirm the effectiveness of CBFM and DeepCBFM, we select four baseline models:

To confirm the effectiveness of CBFM and DeepCBFM, we select four baseline models:
FM [4]: Factorization Machines are a well-known method in Collaborative Filtering recommendation systems. They work by processing the rating matrix and converting it into a pair of low-rank matrices through a transformational procedure.
CBMF [18]: It is an extension of Matrix Factorization adapted with contextual information.
CANCF [26]: It is a hybrid approach that adapts and repurposes a prefiltering method for integrating context.
CAMCRS [21]: It employs deep learning models to predict ratings while considering contextual factors and simultaneously learns how to aggregate this information effectively.

D. Results of the Experiments

For the development environment, we use TensorFlow [27] to implement the model, installed on a computer using Windows 10 with 16GB of RAM. The implementation is inspired from [28]. Each dataset is splitted to train data (80%) and testing data (20%). The optimization method used is Adam, we fix the learning rate to 0.00001. we use a mini-batch of 4096. To avoid the overfitting problem, we use L2 regularization. We fine-tune two parameters to extract the best performance of DeepCBFM. The first parameter is the embedding size, as shown in figure 3, we observe that the RMSE reaches its lowest value when the size is 32, because a large size brings a better representation capacity to the model.

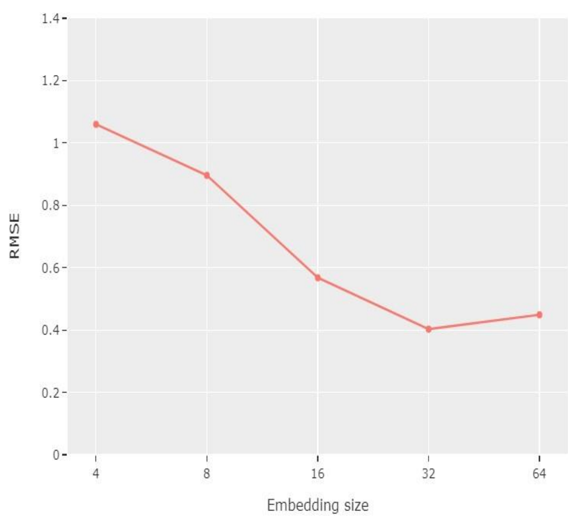

Fig. 3. RMSE results of CBFM under the embedding size parameter.

Secondly, we analyze the dropout parameter, which is used to prevent the overfitting.

As illustrated in figure 4, the dropout is set to a value between 0 and 0.9 with a jump of 0.1 each time. The results show that the DeepCBFM reaches the lowest RMSE value as the dropout equals 0.3, which justify the usefulness of the dropout.

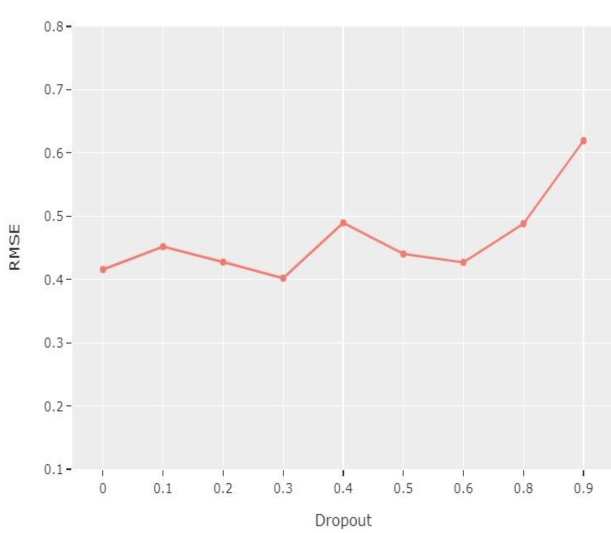

Fig. 4. RMSE results of CBFM under the dropout parameter.

As mentioned earlier, we conducted a comparative analysis of The DeepCBFM against four state-of-the-art models, utilizing two distinct datasets. The results obtained from both datasets were evaluated based on Root Mean Square Error (RMSE) and R-squared metrics. Additionally, a specific focus was placed on assessing the effectiveness of the CBFM component in comparison to other methodologies, with a particular emphasis on the FM algorithm.

Figure 5 illustrates the Root Mean Square Error (RMSE) results for the DePaulMovie dataset. Notably, the model attains the lowest RMSE of 0.4027, signifying its superior performance compared to other models. It is noteworthy that the CBFM component demonstrates favorable outcomes in contrast to the conventional FM approach.

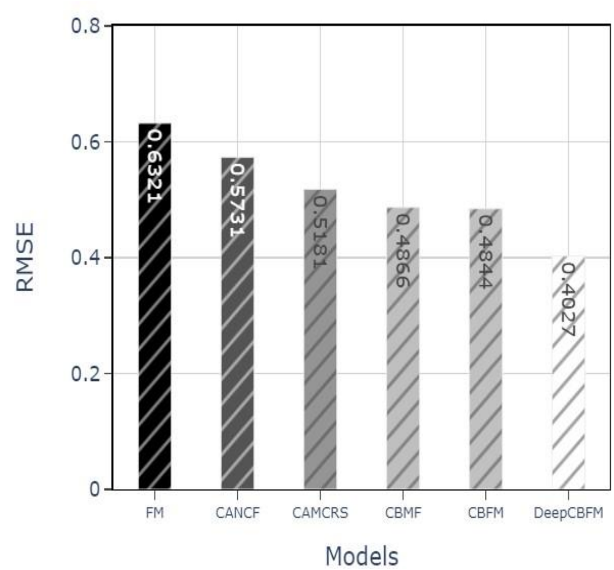

Fig. 5. RMSE results obtained for DePaulMovie dataset.

In figure 6 the results for the DePaulMovie dataset are presented using the R Squared (R^2) metric. R^2, ranging between 0 and 1, indicates the goodness of fit, with a higher value suggesting a more accurate alignment between predicted and actual values. The R^2 measure provides results in a readily interpretable percentage format. In this context, the DeepCBFM model surpasses the CBMF model, the second-best performer, by more than 3%, and outperforms the FM model, the least efficient model, by approximately 8%.

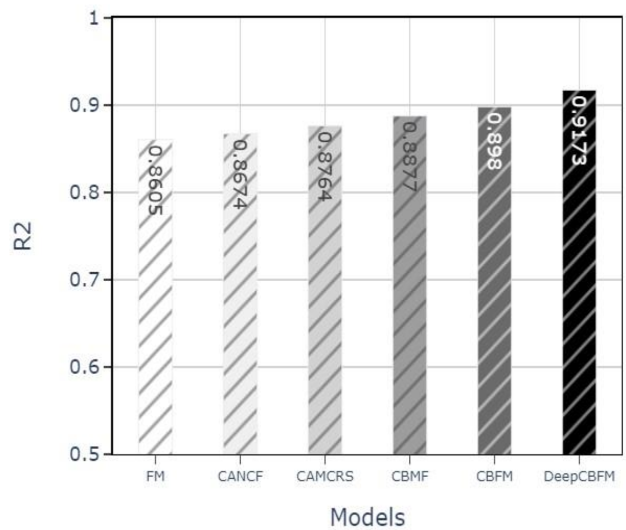

Fig. 6. R Square results obtained for DePaulMovie dataset.

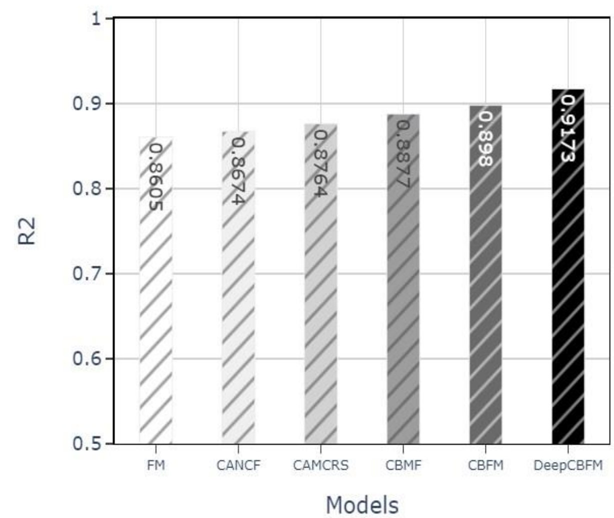

Fig. 8. R Square results obtained for InCarMusic dataset.

Figure 7 and 8 show the results for the InCarMusic dataset expressed in RMSE and R2.

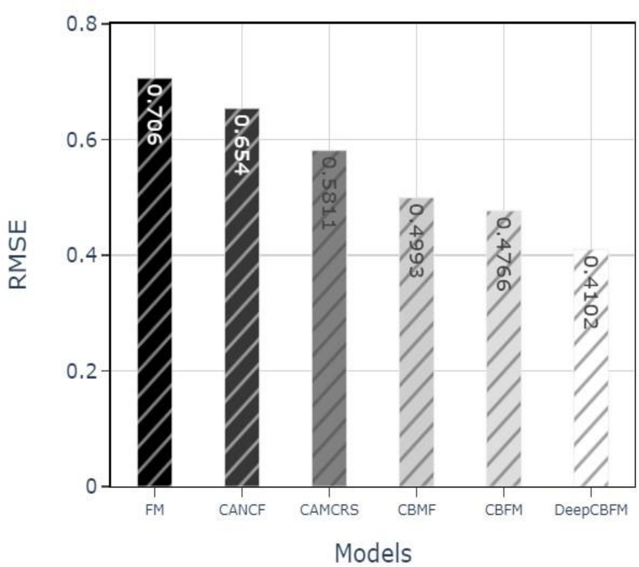

Fig. 7. RMSE results obtained for InCarMusic dataset.

Consistent with the findings from the initial dataset, the outcomes from the Music dataset reaffirm the ranking of performance among all models. Notably, the FM model exhibits a decline in results, attributed to the dataset's inherent high contextual dimensions. Additionally, noteworthy observations include the close similarity in results between CBMF and CBFM models. This similarity can be attributed to their shared design focus on capturing low-order feature interactions.

Table 2 shows all the obtained results in more details.

TABLE 2: OBTAINED RESULTS FOR BOTH DATASETS.

DePaulMovie dataset		
	RMSE	R2
FM	0.6421	0.8605
CANCF	0.5831	0.8674
CAMCRS	0.5181	0.8764
CBMF	0.4866	0.8877
CBFM	0.4844	0.898
DeepCBFM	0.4027	0.9173
InCarMusic dataset		
	RMSE	R2
FM	0.706	0.8735
CANCF	0.654	0.8867
CAMCRS	0.5811	0.8910
CBMF	0.4993	0.8973
CBFM	0.4766	0.9054
DeepCBFM	0.4102	0.9283

5. Conclusion

In this work, we presented DeepCBFM, a Factorization machines based deep neural network for CARSs, to overcome shortcomings of baseline models and to propose a more adaptive model to model contextual data efficiently. The proposed model has two virtues: first, it handles the different feature behaviors when they interact with other features from different contexts and second, it captures the high order features interactions and models nonlinear problems using deep learning. To illustrate the DeepCBFM effectiveness, we conduct experiments on two real world datasets (DePaulMovie and InCarMusic datasets) and results show that our model improves the accuracy of predictions and performs better than the state of art models.

There are two potential directions for future work. First, the time complexity of the model is higher than the baseline models. Therefore, it is interesting to improve the complexity of the model. Second, it is difficult to obtain contextual information from real world applications, so, we plan to explore new sources of contextual data such as unstructured data and develop efficient methods to extract them.

References

[1] R. MADANI, A. EZ-ZAHOUT, et A. IDRISSI, « An Overview of Recommender Systems in the Context of Smart Cities », in 2020 5th International Conference on Cloud Computing and Artificial Intelligence: Technologies and Applications (CloudTech), nov. 2020, p. 1-9. doi: 10.1109/CloudTech49835.2020.9365877.

[2] C. Basu, H. Hirsh, et W. Cohen, « Recommendation as classification: using social and content-based information in recommendation », in Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence, USA, juill. 1998, p. 714-720.

[3] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, et J. Riedl, « GroupLens: an open architecture for collaborative filtering of netnews », in Proceedings of the 1994 ACM conference on Computer supported cooperative work, New York, NY, USA, oct. 1994, p. 175-186. doi: 10.1145/192844.192905.

[4] S. Rendle, “Factorization Machines,” in 2010 IEEE International Conference on Data Mining, Dec.2010,pp.9951000.doi:10.1109/ICDM.2010.127.

[5] G. Adomavicius, B. Mobasher, F. Ricci, et A. Tuzhilin, « Context-Aware Recommender Systems », AI Magazine, vol. 32, no 3, Art. no 3, 2011, doi: 10.1609/aimag.v32i3.2364.

[6] H.-T. Cheng et al., « Wide & Deep Learning for Recommender Systems ». arXiv, 24 juin 2016. doi: 10.48550/arXiv.1606.07792.

[7] H. Guo, R. Tang, Y. Ye, Z. Li, et X. He, « DeepFM: A FactorizationMachine based Neural Network for CTR Prediction ». arXiv, 13 mars 2017. doi: 10.48550/arXiv.1703.04247.

[8] J. Xiao, H. Ye, X. He, H. Zhang, F. Wu, et T.-S. Chua, « Attentional Factorization Machines: Learning the Weight of Feature Interactions via Attention Networks ». arXiv, 15 août 2017. doi: 10.48550/arXiv.1708.04617.

[9] J. Lian, X. Zhou, F. Zhang, Z. Chen, X. Xie, and G. Sun, "xDeepFM: Combining Explicit and Implicit Feature Interactions for Recommender Systems," in Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Jul. 2018, pp. 1754–1763. doi: 10.1145/3219819.3220023.

[10] W. Song et al., "AutoInt: Automatic Feature Interaction Learning via Self-Attentive Neural Networks," in Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Nov. 2019, pp. 1161–1170. doi: 10.1145/3357384.3357925.

[11] Y. Yu, Z. Wang, and B. Yuan, “An Input-aware Factorization Machine for Sparse Prediction,” pp. 1466–1472, 2019.

[12] J. Pan et al., “Field-weighted Factorization Machines for Click-Through Rate Prediction in Display Advertising,” in Proceedings of the 2018 World Wide Web Conference on World Wide Web - WWW ’18, 2018, pp. 1349–1357. doi: 10.1145/3178876.3186040.

[13] G. Trigeorgis, K. Bousmalis, S. Zafeiriou, and B. W. Schuller, “A Deep Matrix Factorization Method for Learning Attribute Representations,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 3, pp. 417–429, Mar. 2017, doi: 10.1109/TPAMI.2016.2554555.

[14] R. Lara-Cabrera, Á. González-Prieto, and F. Ortega, “Deep Matrix Factorization Approach for Collaborative Filtering Recommender Systems,” Applied Sciences, vol. 10, no. 14, Art. no. 14, Jan. 2020, doi: 10.3390/app10144926.

[15] A. Ez-zahout, H. Gueddah, A. Nasry, R. Madani, and F. Omary, “A hybrid big data movies recommendation model based k-nearest neighbors and matrix factorization,” Indonesian Journal of Electrical Engineering and Computer Science, vol. 26, no. 1, Art. no. 1, Apr. 2022, doi: 10.11591/ijeecs.v26.i1.pp434-441.

[16] L. Baltrunas, B. Ludwig, and F. Ricci, “Matrix factorization techniques for context aware recommendation,” in Proceedings of the fifth ACM conference on Recommender systems, New York,

NY, USA, Oct. 2011, pp. 301–304. doi: 10.1145/2043932.2043988.
[17] R. Madani and A. Ez-zahout, “A Review-based Context-Aware Recommender Systems: Using Custom NER and Factorization Machines,” International Journal of Advanced Computer Science and Applications (IJACSA), vol. 13, no. 3, Art. no. 3, 42/30 2022, doi: 10.14569/IJACSA.2022.0130365.
[18] M. Casillo, B. B. Gupta, M. Lombardi, A. Lorusso, D. Santaniello, and C. Valentino, “Context Aware Recommender Systems: A Novel Approach Based on Matrix Factorization and Contextual Bias,” Electronics, vol. 11, no. 7, Art. no. 7, Jan. 2022, doi: 10.3390/electronics11071003.
[19] S.-Y. Jeong and Y.-K. Kim, “Deep Learning-Based Context-Aware Recommender System Considering Contextual Features,” Applied Sciences, vol. 12, no. 1, Art. no. 1, Jan. 2022, doi: 10.3390/app12010045.
[20] A. Sattar and D. Bacciu, “Graph Neural Network for Context-Aware Recommendation,” Neural Process Lett, vol. 55, no. 5, pp. 5357–5376, Oct. 2023, doi: 10.1007/s11063-022-10917-3.
[21] S.-L. Vu and Q.-H. Le, “A Deep Learning Based Approach for ContextAware Multi-Criteria Recommender Systems,” csse, vol. 44, no. 1, Art. no. 1, 2022, doi: 10.32604/csse.2023.025897.
[22] O. Chapelle, E. Manavoglu, and R. Rosales, “Simple and Scalable Response Prediction for Display Advertising,” ACM Trans. Intell. Syst. Technol., vol. 5, no. 4, p. 61:1-61:34, décembre 2015, doi: 10.1145/2532128.
[23] M. Richardson, E. Dominowska, and R. Ragno, “Predicting clicks: estimating the click-through rate for new ads,” in Proceedings of the 16th international conference on World Wide Web, New York, NY, USA, mai 2007, pp. 521–530. doi: 10.1145/1242572.1242643.
[24] Y.-W. Chang, C.-J. Hsieh, K.-W. Chang, M. Ringgaard, and C.-J. Lin, “Training and Testing Low-degree Polynomial Data Mappings via Linear SVM,” p. 20.
[25] Yong Zheng, Bamshad Mobasher, Robin Burke. "CARSKit: A JavaBased Context-aware Recommendation Engine", Proceedings of the 15th IEEE International Conference on Data Mining (ICDM) Workshops, pp. 1668-1671, Atlantic City, NJ, USA, Nov 2015
[26] I. M. A. Jawarneh et al., "A Pre-Filtering Approach for Incorporating Contextual Information Into Deep Learning Based Recommender Systems ", IEEE Access, vol. 8, p. 40485-40498, 2020, doi: 10.1109/ACCESS.2020.2975167.
[27] M. Abadi et al., “TensorFlow: a system for large-scale machine learning,” in Proceedings of the 12th USENIX conference on Operating Systems Design and Implementation, USA, Nov. 2016, pp. 265–283.
[28] Weichen Shen. (2017). DeepCTR: Easy-to-use,Modular and Extendible package of deep-learning based CTR models. https://github.com/shenweichen/deepctr.

Rabie MADANI received the master's degree in Big Data in 2018 from Mohammed V University, Rabat, Morocco and he is currently a Ph.D. student at the faculty of sciences, Mohammed V University in computer science.

He is Head of IT at the Institute of Africans Studies in Rabat. His research domains include artificial intelligence, big data, recommender systems and text mining. has authored and coauthored many publications in international journals and conferences. He is currently a member of the Intelligent Processing Systems and Security team.

Abderrahmane EZ-ZAHOUT is currently an assistant professor of computer science at department of computer science/faculty of sciences at Mohammed V University. His researches are in fields of computer sciences, digital systems, big data and computer vision. Recently, He works on intelligent systems. He has served as invited reviewer for many journals. Besides, he is also involved in NGOs, student associations, and managing nonprofit foundation

Fouzia Omary is the leader IPSS team research and currently a full professor of computer science at department of computer science/ENS, Mohammed V University. She is a full time researcher at Mohammed V University and she is the chair of the international conference on cybersecurity and blockchain and also the leader of the national network of blockchain and cryptocurrency.