

A Novel Context-Aware Deep Learning Algorithm for Enhanced Movie Recommendation Systems



Qi Zhang*^{}, Mideth Abisado^{}

College of Computing and Information Technologies, National University, Manila 1008, Philippines

Corresponding Author Email: zhangq@students.national-u.edu.ph

Special issue: Machine Learning and Artificial Intelligence for Context-Aware Network Optimization

Copyright: ©2023 IIETA. This article is published by IIETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/mmep.100613>

ABSTRACT

Received: 15 July 2023

Revised: 3 September 2023

Accepted: 10 September 2023

Available online: 21 December 2023

Keywords:

deep learning, recommendation system optimization, behavioral modeling, tensor factorization, weighted high-order factorization, machine learning in media, pattern recognition, context-sensitive analysis, context-aware, collaborative filtering

Recommendation systems serve as a pivotal solution to address the increasing issue of information overload. While traditional recommendation algorithms have been grounded primarily on user-item interactions, the significance of a user's contextual information influencing decision-making has often been overlooked. Such neglect becomes more evident in the realm of systems integrating contextual mechanisms, which encounter pronounced data sparsity challenges. Existing studies in contextual recommendation systems tend to treat all contextual features uniformly as influencers of user decisions. Yet, a prevalent dilemma is the frequent absence of contextual data, leading to potential misallocations of contextual features. To mitigate these challenges, a novel deep learning-based recommendation system, termed the CAW-NeuMF Model, has been designed. Accompanying this model, a Context-aware Weighted high-order Tensor Factorization algorithm (CAWTF) has been introduced. This algorithm facilitates the calculation of correlations between user ratings in varied contexts, relying on the said context. Additionally, it ascertains the weight of context features grounded on the user ratings correlation. Such a process aids in isolating the most influential contextual features, thereby amplifying the efficiency of personalized recommendations. Empirical evaluations using the LDOS CoMoDa dataset revealed that the proposed model substantially enhances prediction score accuracy. Comparative analyses against alternative recommendation models further affirmed the superior efficacy of the introduced approach.

1. INTRODUCTION

With the burgeoning of the Internet and advancements in information technology, an exponential surge in data across network platforms has been observed [1]. Such proliferation not only furnishes users with a wealth of intriguing information but paradoxically engenders a quandary of information overload. Consequently, sifting through this deluge to pinpoint pertinent information has become an arduous task [2]. This conundrum is mirrored in the film industry, where recommendation systems have been posited as an efficacious remedy. At the heart of these systems lie recommendation algorithms, meticulously developed to tailor to users' specificities [3]. Optimal algorithms are pivotal as they not only streamline users' quest for intriguing films, thereby catalysing their consumption vigour, but also cascade into monumental economic windfalls for movie purveyors.

Historically, binary relations between users and projects have predominantly been leveraged for modelling in traditional recommendation systems [4]. Despite the commendable efficacy of some of these conventional algorithms, a glaring oversight is the neglect of context, a

pivotal facet influencing user decisions. Such contexts are indispensable for enhancing recommendation accuracy, as they frequently shape user predilections [4]. For instance, a solitary movie viewer might be well-served by existing systems, but the same might falter when the viewer is accompanied by children, with an inclination towards animation genres. Similarly, prevailing emotional states, such as despondency, might skew choices towards comedic films, while romantic companions might prompt a tilt towards romance-themed films. Evidently, the immediate milieu in which a user is ensconced invariably impacts their final verdict [5].

It becomes unequivocally apparent that the infusion of contextual features into recommendation algorithms is paramount to bolster both the precision and relevancy of recommendations. In this vein, the CAWTF algorithm is introduced, designed meticulously to discern correlations between context and ratings, subsequently determining the weights attributed to various contextual features. Through such a procedure, the conundrums of data sparsity endemic to context-aware systems are assuaged. Given the prowess of deep learning in discerning intricate feature relationships, a

fusion of CAWTF with neural networks culminated in the birth of the CAW-NeuMF model.

This manuscript is structured to first delineate the theoretical foundation underpinning context-aware recommendation systems. The ensuing section elucidates the high-order tensor decomposition algorithm predicated on context weight and presents the context-aware recommendation system delineated herein. The penultimate section is dedicated to empirical validation of the introduced algorithm and model, attesting to their efficacy and practicality.

2. RELATED WORK

Within the domain of recommendation research, heightened interest has been observed in the nexus between contextual information and user predilections. The pioneering work by Adomavicius and Tuzhilin [6] led to the conceptualisation of Context Aware Systems (CARS). A rigorous method was proposed by him to distil salient factors influencing ratings, striving to offer bespoke services tailored to individual users. It is contended that, in context-aware systems, the nature of contextual information captured invariably dictates the precision of ensuing preference extrapolation [7]. Hence, the methodologies employed in the curation and discernment of such information bear profound implications for the overarching efficacy of the recommendation system [8, 9].

Historically, temporal attributes have been ubiquitously harnessed as features, primarily attributed to their ease of capture [10]. The advent and ubiquity of mobile intelligent terminals have ushered in an era where a plethora of contextual data can be effortlessly amassed. It has been noted that, through mobile device sensors, a myriad of high-dimensional contextual metrics are readily accessible [11]. For instance, mobile Global Positioning System (GPS) can pinpoint user locales, such as workplaces or residences, while accelerometers can divulge user activity states, be it ambulatory or sedentary. Although the facilitation of such diverse contextual data acquisition is undeniably advantageous, the inadvertent integration of irrelevant contexts can deleteriously attenuate the efficiency of contextual recommendation systems. This not only exacerbates the rigours associated with data collection but also augments computational overheads related to processing superfluous contexts. Furthermore, a poignant observation is that context impact varies across different application paradigms, accentuating the paramountcy of judicious context selection.

Significant strides have been made by scholars in refining context-aware recommendation systems [12]. There is a consensus that the amalgamation of deep learning with context-aware paradigms necessitates further exploration to elevate recommendation precision [13]. In some investigations, topic models were advanced to sift through copious user context logs, endeavouring to unearth prevalent context-aware inclinations [14]. Tensor factorization (TF), a methodology to pare down the dimensions of multi-faceted matrices, has also been examined. Despite its demonstrable superiority over matrix decomposition in navigating multi-dimensional matrices, an uptick in iteration counts was discerned, potentially compromising efficiency in contextual information feature handling [15]. Conversely, employing clustering algorithms to structure a contextual feature model

and collate user and project-related contextual feature data was found to effectively mitigate data sparsity issues. Deep learning models, encompassing Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) architectures, were also juxtaposed against conventional recommendation algorithms [16]. Such endeavours substantiated that deep learning paradigms can adeptly handle abundant contextual feature data and decipher intricate interrelations therein [17]. Notwithstanding these advancements, a lacuna persists. While context-aware systems do factor in contextual attributes, in many real-world scenarios, the correlation between users and most contextual instances remains non-existent, thereby underscoring the exigency for meticulous contextual feature allocation. It is also posited that context-aware systems grapple with more acute data sparsity challenges than their two-dimensional recommendation system counterparts.

3. RESEARCH METHOD

In the exploration of context-aware recommendation systems, it has been observed that the focus on contextual features was not exhaustive [18]. All contextual features were considered, yet the treatment of these features lacked thoroughness. Additionally, the introduction of context presented significant sparsity challenges [19]. Notably, the introduction of contextual information, based on users and projects, expands the system from a two-dimensional matrix to a multidimensional one, as illustrated in Figure 1. This section initially deliberates upon the context-based weight allocation algorithm, subsequently introducing the CAW-NeuMF model, which amalgamates user, project, and contextual feature information to establish a deep learning recommendation system.

Traditionally, the dimensionality was reduced mainly through tensor decomposition or matrix decomposition. In this discourse, a novel method for dimensionality reduction is presented: a high-order tensor decomposition algorithm centred on contextual features to determine weight allocation. The algorithm's emphasis, first and foremost, is the calculation of context feature weights and the subsequent weight allocation algorithm.

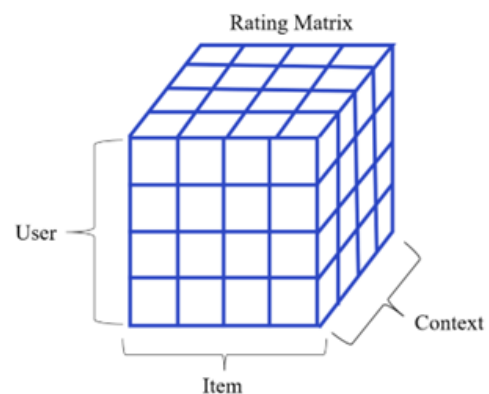


Figure 1. Multidimensional context space

The process is bifurcated into two distinct phases. Initially, the weight of each context is determined, followed by the alignment of the context features most closely associated with user relevance. It has been hypothesised that within identical

contextual settings, varying users' impacts on project ratings differ significantly. For instance, when an individual watches a film accompanied by a significant other, preferences may lean towards romantic or family genres. In such scenarios, correlations between users and the context of social relationships become paramount, while ties to other contextual features diminish.

By calculating the standard deviation of each user's context rating, the most pertinent context in relation to the instance user is discerned. If the standard deviation of the user in one context exceeds that in another, it is inferred that the former context exhibits higher correlation with the user, subsequently neglecting the least correlating contextual feature. Historical scoring data can be exemplified as follows in Table 1.

Table 1. Data instance 1

User	Item	Rating	Time	Weather
U ₁	I ₁	2	Morning	Sunny
U ₁	I ₂	4	Afternoon	Rainy
U ₁	I ₃	3	Night	Cloudy
U ₁	I ₄	3	Afternoon	Sunny

In the pursuit of discerning which context holds a robust correlation with the rating, the standard deviation of each user was calculated. The elucidation of the specific algorithmic steps is presented as follows:

Step 1: It is posited that the dataset t comprises $t \in \{1, 2, \dots, m\}$ contextual features, symbolised as C_t . Dependent on varying criteria, the dataset is partitioned into distinct $D \in \{D_1, D_2, \dots, D_x\}$ segments.

Step 2: From each instance of context t , data is arbitrarily selected from the partitions delineated in the first step. This data is subsequently divided into $|C_t|$ replicas. Within every instance of the context, the pertinent mean value r_{z_k} is computed.

Step 3: Following the acquisition of the mean values for each instance as delineated in the second step, the overall average score for all instances of the context C_t is computed in accordance with Eq. (1).

$$\bar{r}_{C_t} = \frac{1}{|C_t|} \sum r_{z_k} \quad (1)$$

Step 4: In alignment with Eq. (2), the standard deviation pertaining to the context C_t score for every specific item, denoted as i , is computed.

$$M_{U_{it}} = \frac{\left[\sum_{C_k \in C_t} (r_{z_k} - \bar{r}_{C_t})^2 / (|C_t| - 1) \right]^{1/2}}{\bar{r}_{C_t}} \quad (2)$$

To exemplify the calculation of standard deviations within the user-context interaction, consider the ratings of user U_1 for the context feature time. Initially, the data was divided into distinct sections based on varying items. The average ratings were ascertained for context time=Afternoon, $r_1=(4+3)/2=3.5$, time=Morning, $r_2=2$, and time=Night, $r_3=3$. Following this, the composite average rating for each instance of Time, $\bar{r}_{time} = (3.5 + 2 + 3)/3 = 2.83$ was computed. Utilising Eq. (3), the standard deviation pertaining to user U_1 's ratings in the context Time was identified as $(2.83-3)^2+(2-2.83)^2+(3-2.83)^2/(3-1)]^{1/2} \cdot 2.83=0.120$. In a parallel manner, the standard deviation associated with user U_1 in the context Weather was calculated, yielding a value of 0.107. This result indicated a

pronounced correlation between the ratings of user U_1 and the context Time. Drawing from these insights, the top K contextual features demonstrating the highest relevance to the score were identified. Subsequently, weights corresponding to each context were determined using Eq. (3). It was observed that an elevated weight signified heightened importance of the respective context.

$$W_{Ct} = \frac{M_{U_{it}}}{\sum_{t=1}^{t=k} M_{U_{it}}} \quad t \in [1, k] \quad (3)$$

3.1 High-order tensor decomposition algorithm based on context weight

Previously, the method for context weight calculation was delineated, aligning the most pertinent context features based on this weight. However, post-introduction of context in recommendation systems, it has been noted that many users exhibit no connection with most contextual instances, leading to pronounced data sparsity. Conventional matrix decomposition approaches become unsuitable. High-order tensor decomposition has been efficaciously utilised for high-dimensional data processing and analysis and finds application in the domain of personalized recommendation systems [20]. Wang et al. [20] utilised higher-order singular value decomposition to address data sparsity in context-aware systems, while Karatzoglou and others incorporated tensor decomposition into context-aware systems, treating all context features collectively. This approach did not account for varied impacts of distinct contexts on recommendation systems. Hence, with an escalating number of context features, tensor decomposition not only becomes computationally intricate but also encounters data sparsity challenges.

Grounded on the context selection method delineated earlier, the most rating-relevant top K contextual features are ascertained. Weights for each context, as per Eq. (3), are then procured. In the ensuing discourse, the second segment of the algorithm will be addressed. By integrating context weights, the third-order tensor corresponding to each context is decomposed, yielding user, item, and context feature matrices.

For elucidative purposes, it is assumed that user historical rating data encapsulating contextual information is presented in Table 2.

Table 2. Data instance 2

User	Item	Rating	C1	C2	...	Cp
U ₁	I ₂	5	1	2	...	1
U ₂	I ₃	2	3	3	...	3
...
U _i	I _j	3	2	1	...	2

To construct the third-order tensor of context C_i from the dataset, each element $a_{u,i,c}$ in the tensor is considered to represent the user u rating of an item i under a given context $C_i=c$. For instance, $a_{1,2,1}$ is indicative of a rating of 5 given by user 1 for the item 2 under the context $c=1$. For an N -order tensor $A \in R^{1 \times 1 \times 1 \times \dots \times 1 \times 1 \times 1 \times \dots \times 1 \times 1 \times 1}$, it is expanded into a matrix prediction, $A_{(n)} \in R^{1 \times (n+1) \times n+2 \times \dots \times n+1 \times 1 \times 2 \times \dots \times 1 \times n-1}$. In the matrix's specified line i_n , the value residing in column $(i_{n+1}-1)I_{n+2}I_{n+3} \dots I_{n-1}I_2 \dots I_{n-1}+(i_{n+2}-1)I_{n+3}I_{n+4} \dots I_{n-1}I_2I_{n-1}+\dots+(i_{n-1}-1)I_1I_2 \dots I_{n-1}+(i_1-1)I_2I_3 \dots I_{n-1}+(i_2-1)I_3I_4 \dots I_{n-1}+\dots+i_{n-1}$ was identified as a_{i_1, i_2, \dots, i_N} . Guided by the previously delineated expansion protocols, three distinct two-dimensional feature

matrices were derived. Subsequently, tensor decomposition was executed on each of these matrices. As a result, the feature matrix $U \in R^{I \times m}$ symbolising the user, the feature matrix $I \in R^{J \times n}$ characterising the project, and the feature matrix $C \in R^{K \times p}$ denoting the context were all acquired. Additionally, the core vector, referred to as $core = R^{m \times n \times p}$, was ascertained.

3.2 Model network structure

Upon analysis of existing technologies and limitations within the recommendation field, deep learning has been applied, resulting in the introduction of the CAW-NeuMF model. As illustrated in Figure 2, this model employs the user feature matrix U , item feature matrix I , and context feature matrix C as neural network inputs. Initially, basic data encompassing users, items, and ratings are utilised to determine the first K context-weight relationships associated with users via context allocation algorithms. Following dimensionality reduction via high-order tensor decomposition, the comprehensive user, item, and context feature matrices are input into the NeuMF neural network for learning and training.

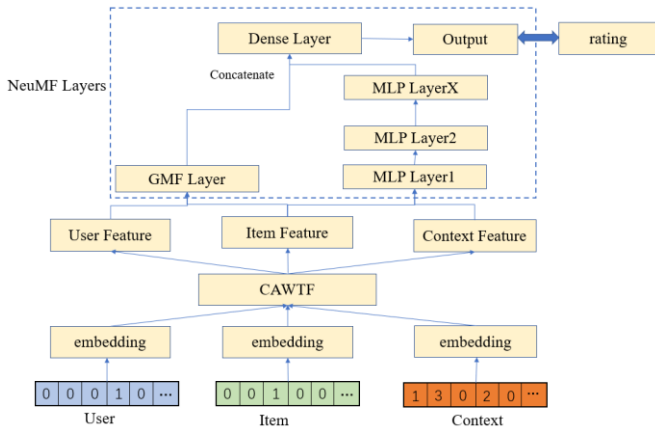


Figure 2. A personalized recommendation model based on deep learning

The NeuMF model, as proposed by He et al. [21], is a variant of Neural Collaborative Filtering (NCF). The General Matrix Decomposition (GMF) extracts low-order (user-item) relations, whilst the Multi-Layer Perceptron (MLP) learns (user-context) relations. The original NeuMF decomposition model, lacking the extraction of efficient auxiliary information for feature modelling, has been enhanced in this study. Contextual auxiliary information is integrated, leading to the formation of CAW-NeuMF, encompassing the refined CAW-NeuGMF and the improved MLP CAW-NeuMLP.

In the CAW-GMF model, the extracted feature vector P_u of user u and feature vector Q_i of item i are element-wise multiplied. Subsequently, the prediction score is derived through a fully connected layer and an activation function as per the given equation.

$$\varphi_o(P_u, Q_i) = P_u \odot Q_i = [u_1 i_1, u_2 i_2, \dots, u_m i_m] \quad (4)$$

$$\hat{y}_{ui} = \sigma((P_u \odot Q_i)^T h) \quad (5)$$

where, n is the dimension of the characteristic matrix, σ is the activation function, and h is the weight matrix of the full connection layer.

Within the CAW-MLP model, it becomes essential to

concatenate the extracted user feature vector P_u , item i feature vector, and context feature vector R_c with weight data. The Eq. (6) is as follows:

$$z_1 = \varphi_1(P_u, Q_i, R_c) = [u_1, \dots, u_m, i_1, \dots, i_n, c_1, \dots, c_t] \quad (6)$$

Post concatenation, the vector of length $m+n+t$ undergoes MLP processing, and prediction scores are then derived using a fully connected layer alongside an activation function. The calculation process is shown in the following Eqs. (7)-(9).

$$\varphi_2(z_1) = \sigma_2(z_1^T w_2 + b_2) \quad (7)$$

$$\dots$$

$$\varphi_X(z_{X-1}) = \sigma_X(z_{X-1}^T \omega_X + b_X) \quad (8)$$

$$\hat{y}_{uic} = \sigma_X(\varphi_X(z_{X-1})^T h) \quad (9)$$

where, σ_i is the activation function. For each layer, the RELU function was selected as the activation function. ω_i and b_i are denoted as the weight and bias of the fully connected layer, respectively.

The NeuMF model, designed to amalgamate the strengths of GMF and MLP models, was implemented. Vectors φ_{GMF} and φ_{MLP} , derived through GMF and MLP, respectively, were concatenated and subsequently channeled into the terminal NeuMF layer. The underlying principle of this operation was to map the vectors to a one-dimensional space, culminating in the ensuing predictive equations:

$$Z_{ij} = [\varphi_{ij}^{GMF}, \varphi_{ij}^{MLP}] \quad (10)$$

$$\gamma_{u,i,c} = \sigma(h^T(Z_{ui})) \quad (11)$$

To circumvent the pitfall of overfitting, a loss function enriched with an L2 regularization term was employed in training the model parameters. Within this framework, ω and b represent the discrepancies in weights across various layers of the neural network, while λ serves as the coefficient of the regularization term. The specific configuration of the loss function is delineated in:

$$Loss = \frac{\sum_n (\gamma_{u,i,c} - \hat{y}_{u,i,c})}{n} + \lambda(\omega^2 + b^2) \quad (12)$$

For optimization of u_i and v_j , the Adam optimizer was harnessed. Its mode of updating is explicated in the subsequent formula.

$$u_i = u_i - \eta \frac{\partial L(U, V)}{\partial u_i} \quad (13)$$

$$v_j = v_j - \eta \frac{\partial L(U, V)}{\partial v_j} \quad (14)$$

Within the referenced formula, the learning rate, denoted as η , is initially set at 0.001. Unlike conventional random gradient descent optimisation algorithms in machine learning, which maintain a constant learning rate throughout the training, Adam is distinguished by its capacity to prescribe

adaptive learning rates for individual parameters. This trait gives Adam an edge over the Stochastic Gradient Descent (SGD). In essence, the Adam optimisation algorithm assimilates the merits of both the adaptive gradient algorithm and the root mean square propagation algorithm. Table 3 is the training procedure predicated on the deep learning context weight algorithm:

Table 3. Training of the CAW-NeuMF personalized recommendation model via deep learning

Input: User-Item Rating Data, Contextual Data
Output: Predicted User-Item Rating
1. An embedding matrix for users was constructed, and the user feature matrix was subsequently extracted utilising CAWTF.
2. Similarly, an embedding matrix for items was formulated, from which the item feature matrix was extracted with the aid of AWTF.
3. A context embedding matrix was devised. The premier K context features were selected and their respective weights were computed.
4. Low-level user-project relationships were discerned using Eq. (4).
5. High-level relationships, encompassing users, projects, and context, were comprehended by referring to Eqs. (6)-(8).
6. Features were concatenated and subsequently mapped onto a one-dimensional space, facilitating predictions as delineated in the corresponding formulas.
7. The loss function was computed. Iterative updates of the user and project feature matrices were carried out until convergence was achieved, as explicated in Eqs. (12)-(14).

4. EXPERIMENTAL RESULTS AND ANALYSIS

The LDOS-CoMoDa, a movie rating dataset renowned for its rich contextual information, was employed [22]. Unlike conventional datasets, this particular one collates ratings and 12 types of contextual information directly after the movie viewing experience, ensuring that the data reflects real-time user responses rather than historical records or speculative reactions [23]. A comprehensive distribution of the dataset is depicted in Table 4.

Table 4. LDOS-COMODA dataset distribution

Dataset	Rating	User	Item	Variable	Score	Sparsity (%)
LDOS-CoMoDa	2296	121	1232	30	1-5	98.45

Experiments were conducted in a controlled environment: Linux operating system with 16GB memory, Intel (R) Core (TM) i7-8700k CPU 3.70 GHz. Python was the chosen programming language, and TensorFlow was utilised as the deep learning framework.

To gauge the precision of the prediction score post-model training, two primary metrics, Root Mean Square Error (RMSE) and Mean Absolute Error (MAE), were employed. The underlying premise was to predict users' proclivity scores for movies not yet viewed based on the users' and movies' characteristic data. Collaborative filtering further buttressed these predictions, offering user-centric movie recommendations and thereby mitigating the adverse effects of the initial sparse scoring data on the recommendation system [24].

MAE denotes the average magnitude of errors between predicted and actual outcomes, whereas MSE (Mean Squared

Error) signifies the average squared differences between predictions and true values. Emphasising the penalisation of larger deviations, MSE is particularly informative. The mathematical formulations for MAE and RMSE are elucidated as:

$$MAE = \frac{\sum_{u,i,c} |y_{uic} - \hat{y}_{uic}|}{Num} \quad (15)$$

$$RMSE = \sqrt{\frac{\sum_{u,i,c} (y_{uic} - \hat{y}_{uic})^2}{Num}} \quad (16)$$

In the aforementioned formulas, \hat{y}_{uic} stands for the anticipated rating value, y_{uic} signifies the factual rating of item i by user u within context c , and Num indicates the total ratings in the test set.

The LDOS-CoMoDa dataset encapsulates features stemming from 12 distinct contexts [25]. Leveraging the algorithm delineated in Section 3, correlations between these contexts and ratings were determined via Eq. (2). The ensuing results are vividly portrayed in Figure 3.

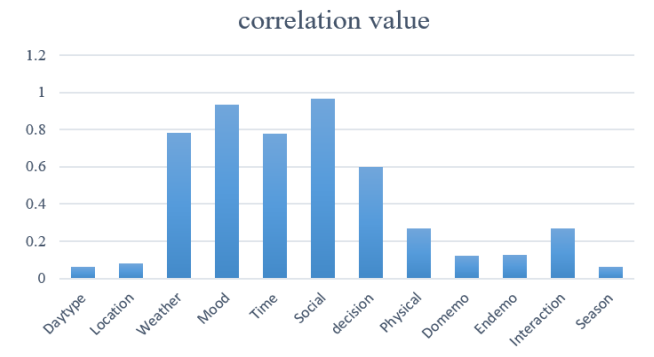


Figure 3. Contextual relevance diagram

According to Figure 3, pivotal factors such as weather, mood, time, social milieu, decision context, physical setting, and interaction type prominently influence user ratings. Using the context weight computation (Eq. (3)), weightings corresponding to each context were ascertained, as represented in the subsequent Table 5.

Table 5. Contextual feature weight distribution

Context	Numbering	Weight
Endemo	w_1	0.35
Domemo	w_2	0.32
Social	w_3	0.89
Mood	w_4	0.80
Time	w_5	0.71
Location	w_6	0.10
Physical	w_7	0.41
Season	w_8	0.06
Decision	w_9	0.55
Interaction	w_{10}	0.42
Daytype	w_{11}	0.08
Weather	w_{12}	0.72

For the purpose of curtailing data sparsity, records of users with fewer than 5 ratings were excluded [26]. As a result, a refined dataset of 2,175 ratings pertaining to 1,302 movies from 56 distinct users was obtained. A rigorous ten-fold cross-validation technique was implemented, segregating 90% of

data for training and the remaining 10% for testing. Throughout the course of 10 experimental runs, regularization coefficients were set at $\lambda=0.01$, the preliminary learning rate was fixed at 0.001, and the upper limit for iterations was capped at 500. The collective mean of MAE and RMSE across these 10 iterations was chosen as the definitive evaluation metrics.

A perusal of Figure 4 suggests that post 100 training iterations, convergence patterns emerged. Notably, the validation set's influence predominantly remained inferior to the training set's MSE, underscoring the model's efficacy.

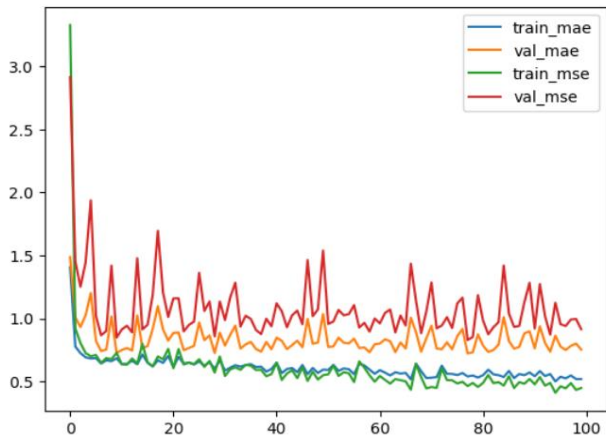


Figure 4. Trend analysis of experimental data

Insights from Figure 4 further reveal a swift model convergence within the initial five training iterations. With the MAE and MSE values of the training set represented by blue and green trajectories respectively, the model's commendable performance becomes apparent. Preventative measures against overfitting were instituted, enhancing the code's generalization. Concomitantly, the MAE in the validation set surpassed the MSE, exhibiting a decline analogous to the training set.

In assessing the efficacy of the model, a comparative analysis was executed with several prominent recommendation models, namely: BiasedMF [27], an offset term matrix decomposition model heralded during the Netflix Prize; NeuMF [28], an innovative neural matrix decomposition model that synergistically integrates GMF and MLP; and CARS [29], a context-aware recommendation model. Furthermore, a novel tensor decomposition algorithm based on context weight, termed CAW NeuMF, was introduced in this study.

A schematic representation, illustrated in Figure 5, underscores the analogous precision distribution across the trio of primary recommendation model algorithms. Evidently, the matrix decomposition model generally lags behind the tensor decomposition model in terms of predictive accuracy. This discrepancy can be attributed to the tensor decomposition model's superior performance when grappling with high-dimensional data. Furthermore, the divergence between the actual values and the predicted outcomes for CARS exceeds that of CAW NeuMF. This suggests that the incorporation of WACTF, which duly considers context weight and judiciously allocates contextual features, holds a distinct advantage over indiscriminately introducing all contextual attributes into the model.

An empirical evaluation of the aforementioned

recommendation algorithms, juxtaposed with the proposed CAW-NeuMF method, was embarked upon. The purpose of this investigation was to substantiate the claim that, upon assimilating contextual features, the CAW-NeuMF approach eclipses other recommendation methodologies. The experimental findings have been encapsulated in Table 6 below.

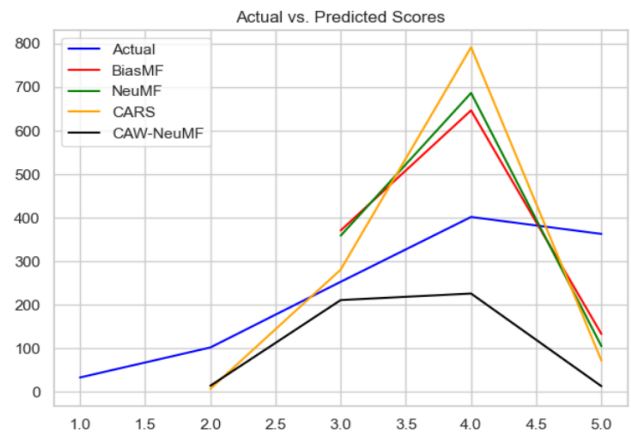


Figure 5. Schematic representation of actual vs. predicted value comparisons

Table 6. Comparative performance on RMSE and MAE metrics for LDOS-COMODA dataset

Model	Evaluation Metric		
	MAE	MSE	Good/Bad
BiasMF	0.76	0.88	Bad
NeuMF	0.75	0.94	Bad
CARS	0.75	0.89	Bad
CAW-NeuMF	0.72	0.82	Good

Insights derived from Table 6 manifest that when employing RMSE and MAE as metrics for recommendation systems, especially within the LDOS-COMODA dataset, the Deep-AIRS algorithm showcases a pronounced improvement relative to its peers. Specifically, in terms of the MAE metric, the Deep-AIRS algorithm exhibits enhancements of 4.00% in comparison with the BiasMF model, 5.236% with the NeuMF algorithm, and 4.000% vis-à-vis the CARS model. Analogously, the RMSE metric reveals that the Deep-AIRS algorithm trumps the BiasMF model by 12.766%, the NeuMF model by 6.818%, and the CARS by 7.865%. These empirical results cogently argue in favour of the superior accuracy of the CAW-NeuMF algorithm within the contextual recommendation domain, thereby demonstrating its capacity to better satiate user-specific requirements.

Within the presented table, several recommended algorithm models are evaluated based on their MAE and MSE metrics. From the data, it is discerned that the models under examination all exhibit commendable performance, with MAE values observed to lie between 0.72 and 0.76 and MSE values registering between 0.82 and 0.94.

Upon closer inspection, the NeuMF, BiasMF, and CARS models display congruent results in terms of both MAE and MSE scores, suggesting parallel efficacy in the prediction of target variables. Notably, the CAW-NeuMF model's performance is marginally superior, as evidenced by its reduced scores across both evaluative criteria.

5. CONCLUSIONS

With the inexorable ascent of the internet economy, an exponential augmentation in network data has been observed, leading recommendation systems to proffer enhanced applicability and value for users. Significant ramifications have been imparted upon recommendation systems across myriad sectors, encompassing finance, education, and entertainment, owing to the burgeoning market catalysed by user data. While judicious employment of contextual information stands to bolster the precision of recommendation systems, the integration of such features can concurrently introduce an array of challenges. Among these challenges, increased data dimensionality and concomitant data sparsity have been identified.

Addressing the aforementioned challenges, the focus of this study gravitated towards enhancing the precision of contextual recommendation systems. Noteworthy innovations unveiled within this study encompass, firstly, the proposal of a high-dimensional tensor decomposition algorithm rooted in context weight. Designed to redress the challenges posed by data sparsity and the disparate allocation of context features within recommendation systems, this algorithm adeptly computes the correlation between context and user ratings, culminating in an astute allocation of contextual features. In a subsequent stride, deep learning neural networks were employed to seize both user and contextual nuances. The culmination of this approach, denominated as CAW-NeuMF and predicated on the NeuMF model, serves to accentuate the accuracy of the recommendation system, catering to nuanced user predilections.

Empirical evaluation of the CAW-NeuMF, undertaken using the LDOS-COMODA dataset, evinced its superior performance relative to preceding recommendation models. Notably, it adeptly assimilated contextual feature information, bolstering recommendation accuracy. The revelations from this study proffer a foundational bedrock for subsequent inquiries within the realm of context-aware recommendation systems.

REFERENCES

- [1] Lavanya, R., Singh, U., Tyagi, V., (2021) A comprehensive survey on movie recommendation systems. In 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), Coimbatore, India, pp. 532-536. <https://doi.org/10.1109/ICAIS50930.2021.9395759>
- [2] Tejal, T., Sheetal, A. (2016). YouTube video recommendation via cross-network collaboration. *Foundation of Computer Science*, 146(11): 9-17. <https://doi.org/10.5120/ijca2016910896>
- [3] Chouiref, Z., Hayi, M.Y. (2022). Toward preference and context-aware hybrid tourist recommender system Based on machine learning techniques. *International Information and Engineering Technology Association*, 2022, 36(2): 195-208. <https://doi.org/10.18280/ria.360203>
- [4] Zhu, H., Chen, E., Xiong, H., Yu, K., Cao, H., Tian, J. (2014). Mining mobile user preferences for personalized context-aware recommendation. *Association for Computing Machinery*, 5(4): 1-27. <https://doi.org/10.1145/2532515>
- [5] Winarko, E., Heryawan, L. (2022) Context-aware recommendation system survey: Recommendation when adding contextual information. In 2022 6th International Conference on Information Technology, Information Systems and Electrical Engineering, Yogyakarta, Indonesia, pp. 7-13. <https://doi.org/10.1109/ICITISEE57756.2022.10057940>
- [6] Adomavicius, G., Tuzhilin, A. (2011). Context-aware recommender systems. In *Recommender Systems Handbook*. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-85820-3_7
- [7] Harshali, D. (2021). Movie recommendation system through movie poster using deep learning technique. *International Journal for Research in Applied Science and Engineering Technology*, 9(4): 1574-1581. <https://doi.org/10.22214/ijraset.2021.33947>
- [8] Jeong, S.Y., Kim, Y.K. (2021) Deep learning-based context-aware recommender system considering contextual Features. *Applied Sciences*, 12(1): 45. <https://doi.org/10.3390/app12010045>
- [9] Chouiref, Z., Hayi, M. Y. (2022). Toward preference and context-aware hybrid tourist recommender system based on machine learning techniques. *International Information and Engineering Technology Association*, 36(2): 195-208. <https://doi.org/10.18280/ria.360203>
- [10] Vu, S.L., Le, Q.H. (2023). A deep learning-based approach for context-aware multi-criteria recommender systems computers. *Computer Systems Science and Engineering*, 44(1): 471-483. <https://doi.org/10.32604/csse.2023.025897>
- [11] Mu, Y., Wu, Y. (2023). Multimodal movie recommendation system using deep learning. *Mathematics*, 11(4): 895. <https://doi.org/10.3390/math11040895>
- [12] Fang, J., Li, B., Gao, M. (2020). Collaborative filtering recommendation algorithm based on deep neural network fusion. *International Journal of Sensor Networks*, 34(2): 71-80. <https://doi.org/10.1504/IJSNET.2020.110460>
- [13] Szymdt, M. (2021) Contextual personality-aware recommender system versus big data recommender system. In 2021 24th International Conference on Business Information Systems, Hannover, Germany, pp. 163-173. <https://doi.org/10.52825/bis.v1i1.38>
- [14] Le Q.H., Vu, S.L., Le A.C. (2022) A comparative analysis of various approaches for incorporating contextual information into recommender systems. *Science Publications*, 18(3): 187-203. <https://doi.org/10.3844/jcssp.2022.187.203>
- [15] Zhai, L.L., Xing, H.L., Zhang, S.C. (2016). Research on collaborative filtering recommendation in mobile e-commerce based on context clustering optimization. *Information Theory and Practice*, 39(8): 106-110.
- [16] Iqbal, H.S., Alan, C., Jun, H., Watters, P. (2021). *Context-Aware Machine Learning and Mobile Data Analytics*. Springer International Publishing, pp. 137-146.
- [17] Vu, S.L., Le, Q.H. (2023). A deep learning based approach for context-aware multi-criteria recommender systems computers. *Materials and Continua (tech Science Press)*, 44(1): 471-483. <https://doi.org/10.32604/csse.2023.025897>
- [18] Wang, H. (2022). MovieMat: Context-aware movie recommendation with matrix factorization by Matrix

- Fitting. In 2021 7th International Conference on Computer and Communications (ICCC), Chengdu, China, pp. 1642-1645, <https://doi.org/10.1109/ICCC54389.2021.9674549>
- [19] Cheng, S.L., Jiang, H.M., Wang, W.Y., Jiang, W. (2023). Research on multi-context-aware recommendation methods based on tensor factorization. *Multimedia Systems*, 29: 2253-262. <https://doi.org/10.1007/s00530-023-01103-z>
- [20] Wang, L.C., Meng, X.W., Zhang, Y.J. (2012). Research on several key technologies of context-aware recommendation systems Beijing University of Posts and Telecommunications, 2012.
- [21] He, X.N., Zhang, H.W., Kan, M.Y., Chua, T.S. (2016). Fast matrix factorization for online recommendation with implicit feedback. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, New York, USA, pp. 549-558. <https://doi.org/10.1145/2911451.2911489>
- [22] Manimurugan, S., Almutairi, S. (2022) A user-based video recommendation approach using CAC filtering, PCA with LDOS-CoMoDa. *Journal of Supercomputing*, 78(7): 9377-9391. <https://doi.org/10.1007/s11227-021-04213-5>
- [23] Mu, Y., Wu, Y. (2023). Multimodal movie recommendation system using deep learning. *Mathematics*, 11(4): 895. <https://doi.org/10.3390/math11040895>
- [24] Hartatik, Winarko, E., Heryawan, L. (2022). Context-aware recommendation system survey: Recommendation when adding contextual information. In 2022 6th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE), Yogyakarta, Indonesia, pp. 7-13. <https://doi.org/10.1109/ICITISEE57756.2022.10057940>
- [25] Liu, N.N., Bin, C., Zhao, M. (2010) Adapting neighborhood and matrix factorization models for context aware recommendation. In *Proceedings of the Workshop on Context-Aware Movie Recommendation*, New York, USA, pp. 7-3. <https://doi.org/10.1145/1869652.1869653>
- [26] Abinaya, S., Devi, M.K. (2021). Enhancing top-N recommendation using stacked autoencoder in context-aware recommender system. *Neural Process Letters*, 53: 1865-1888. <https://doi.org/10.1007/s11063-021-10475-0>
- [27] Tejashree, M., Surekha, K. (2022), Recommendation system using matrix factorization. *International Journal for Research in Applied Science and Engineering Technology*, 10(9): 355-359. <https://doi.org/10.22214/ijraset.2022.46615>
- [28] Fang, J., Li, B.C., Gao, M.X. (2020). Collaborative filtering recommendation algorithm based on deep neural network fusion. *Inderscience Publishers*, 34(2): 71-80. <https://doi.org/10.1504/IJSNET.2020.110460>
- [29] Nawara, D., Kashef, R. (2021). Context-aware recommendation systems in the IoT environment (IoT-CARS)—A comprehensive overview. *Institute of Electrical and Electronics Engineers*, 9: 144270-144284. <https://doi.org/10.1109/ACCESS.2021.3122098>