Journal of information and
communication convergence engineering
http://jicce.org

J. Inf. Commun. Converg. Eng. 22(1): 44-55, Mar. 2024 Regular paper

Deep Learning Framework with Convolutional Sequential Semantic Embedding for Mining High-Utility Itemsets and Top-N Recommendations

Siva S[1]* and Shilpa Chaudhari[2]

[1]Department of Computer Science and Applications, Reva University, Bangalore 560064, India
[2]Department of Computer Science and Engineering, M S Ramaiah Institute of Technology, Bangalore 560054, India

Abstract

High-utility itemset mining (HUIM) is a dominant technology that enables enterprises to make real-time decisions, including supply chain management, customer segmentation, and business analytics. However, classical support value-driven Apriori solutions are confined and unable to meet real-time enterprise demands, especially for large amounts of input data. This study introduces a groundbreaking model for top-N high utility itemset mining in real-time enterprise applications. Unlike traditional Apriori-based solutions, the proposed convolutional sequential embedding metrics-driven cosine-similarity-based multilayer perception learning model leverages global and contextual features, including semantic attributes, for enhanced top-N recommendations over sequential transactions. The MATLAB-based simulations of the model on diverse datasets, demonstrated an impressive precision (0.5632), mean absolute error (MAE) (0.7610), hit rate (HR)@K (0.5720), and normalized discounted cumulative gain (NDCG)@K (0.4268). The average MAE across different datasets and latent dimensions was 0.608. Additionally, the model achieved remarkable cumulative accuracy and precision of 97.94% and 97.04% in performance, respectively, surpassing existing state-of-the-art models. This affirms the robustness and effectiveness of the proposed model in real-time enterprise scenarios.

Index Terms: High utility itemset mining, Semantic sequential deep learning, MLP, Cosine similarity, Top-N HUI Recommendation

I. INTRODUCTION

In recent years, software computing, big data analytics, and decentralized computing technologies have emerged. These technologies, in sync with gigantic digital data, have helped enterprises understand user behavior, consumer perceptions, and preferences to improve business decisions. Consumers' purchase behavior and transaction analyses have helped enterprises understand supply chain demands, high-utility items, and periodic demand patterns, thereby playing a decisive role in business communities, including e-commerce, manufacturing, and supply chain industries (e.g., the global value chain). However, identifying a set of high-utility items for the aforementioned digital data or transaction details, remains challenging. Advanced computing technologies, such as pattern mining [1-4], have enabled the identification of high-utility itemsets for business decisions. Typically, in sync with the business ecosystem, pattern mining technologies exploit existing transaction details to understand consumer preferences and socioeconomic behavior, thus helping enterprises improve their marketing mix decisions. They also help consumers identify the intended product or service that meets the respective demands [5]. In other words, the pattern-mining technique can help both enterprises and consumers with certain optimistically identified sets of products or items for corresponding use. In recent years, machine learn-

Received 24 May 2023, **Revised** 16 October 2023, **Accepted** 4 November 2023
***Corresponding Author** Siva S (E-mail: sivaraju.reva@gmail.com)
Department of Computer Science and Applications, Reva University, Bangalore 560064, India

Open Access https://doi.org/10.56977/jicce.2024.22.1.44 print ISSN: 2234-8255 online ISSN: 2234-8883

ing (ML) and artificial intelligence (AI) have played decisive roles in pattern mining-based recommendation systems [5]. An improved computing ecosystem called natural language programming (NLP) can also identify the target output by learning raw input patterns or transaction details [5]. These techniques exploit different input patterns and associated features to make prediction decisions [5], learning over different periodic and sequential patterns, as well as frequent itemsets.

In business ecosystems, pattern analysis methods have gained widespread attention for market basket analysis and business intelligence (BI). BI approaches exploit sequential or periodic transaction details (or data) to identify the frequent itemsets with high utility values. This helps in performing personalized predictions for users. In BI ecosystems, the identification of high-utility itemsets (HUIs) enables the segmentation of N demanding products and services. Classical approaches often exploit intercommodity associations and frequency of item purchases or transactions, to identify HUIs for further prediction [6-8]. Notably, an item is stated to be a frequent itemset when its frequency is greater than a predefined threshold, called the support value [9]. Different approaches have been proposed for frequent high-utility itemset mining (FHUIM); however, merely applying a predefined threshold over large nonlinear features or patterns cannot yield optimal accuracy for generalization. Techniques such as the Apriori [6][7] method determine frequent itemsets by assessing iterative level-based searches to identify HUIs. Notably, these methods employ the downward closure method, in which an a priori characteristic is applied to prune redundant or less redundant items. Apriori-based methods ensure that itemsets possessing a low support value do not become an HUI. However, the iterative estimation of the support value can be computationally demanding, especially over a large search space, which limits the robustness of Apriori-based methods. In recent years, several other approaches such as equivalence class clustering and bottom-up lattice traversal (ECLAT) [2], frequent pattern-growth (FP-Growth) [3][10], and hyper-structure mining (Hmine) [11] were proposed to improve pruning and the associated data structure for HUI identification. These approaches that employ frequent itemset mining, only consider the frequency of the itemsets in relation to other items rather than their respective utility or co-occurrence probability. This limits their utility in contemporary business decisions, where identifying co-occurrence items is as important as identifying HUIs. The literature indicates that other details, such as transaction counts, co-purchased items, their frequency, and high profit value, can help identify HUIs in value-based itemset predictions [12]. Unlike Apriori or frequency-based solutions, high-utility itemset mining (HIUM) [13,14] techniques which employ both volume as well as profit per unit for HUI estimation, have gained widespread attention. These methods aim to improve both accuracy and computational costs to improve scalability [12]. Many state-of-the-art studies have applied the utility factor, which includes total and unit profits of the itemset, in estimating HUIs from a large search space based on transaction details. These approaches perform better than Apriori methods; However, their suitability for generalization to larger dynamic transactions is disputable [12-15]. Studies have revealed that pruning insignificant items from transaction search spaces can minimize computational cost and delay and improve learning-based prediction accuracy [12]. This prompted the development of transaction-weighted utility (TWU) which focuses on improving pruning, whereby the upper threshold is estimated to prune an itemset. To improve accuracy and reduce the complexity over large inputs, two-phase methods have been proposed [16-19], which are hypothesized to be more effective than single-phase solutions [20,21]. In two-phase methods, highly correlated items are first identified. Subsequently, their utility is assessed to label them as HUIs. Despite claims of higher accuracy, the associated computational cost remains a challenge, particularly in iterative database scanning and level-wise utility estimation [20]. The use of a list structure (utility list) performs better. However, the utility value remains proportional to the length of the itemset, making the computation more complex. To alleviate this, the concept of average utility was recently proposed, which focuses on reducing the impact of length on HUI prediction [22-24], applying the average support value to perform HUI estimation. Interestingly, none of these methods can exploit the itemset probability or contextual details among the sequential or periodic transactions in performing HUI itemset prediction.

In this paper, a novel convolutional sequential embedding metric-driven cosine-similarity-based multi-layer perception learning model is proposed for top-N high-utility itemset recommendations. Unlike traditional deep learning solutions, such as convolutional neural network (CNN)-based top-N HUIM models that apply local features for prediction, the proposed model exploits contextual or global features, including semantic features, over sequential transactions to perform top-N recommendations. First, semantic features are extracted from the transaction details encompassing the itemsets and their respective co-occurrence probabilities using the semantic CNN (SCNN) model. The extracted semantic features are processed by a multilayer perceptron (MLP), which retrieves the linear relationship among the itemsets and their corresponding co-occurrence probabilities. Subsequently, the cosine-similarity method is applied over the MLP-predicted linear associations to perform top-N HUI predictions. In MATLAB-based simulations over different datasets, the proposed HUIM model achieved a precision of 0.5632, mean absolute error (MAE) of 0.7610, hit rate (HR)@K of 0.5720, and normalized discounted cumulative gain (NDCG)@K of 0.4268. Additionally, it exhibited an average MAE of 0.608 on different datasets and latent dimensions, achieving a cumulative performance accuracy and precision of 97.94% and 97.04%, respectively. Relative performance characterization revealed that the proposed top-N HUIM model surpasses other state-of-art methods, including CNN, in different federated learning environments, which confirms the robustness of the proposed model and its suitability for real-time enterprise purposes.

The remainder of this paper is organized as follows. Section II discusses related works, followed by the research questions in Section III. Section IV presents the overall proposed model and its implementation. The simulation results and relevant inferences are provided in Section V. Section VI discusses the overall research contributions in the context of future scope. The references used in this study are provided at the end of the manuscript.

II. RELATED WORK

Pattern mining [4] methods were initially designed based on the frequency of itemsets [4]. Frequent itemset mining methods exploit threshold conditions such as support value and profit score to perform HUI estimation. However, applying standalone threshold-based pruning alone cannot yield a robust solution. A few improved methods such as ECLAT [2], FP-Growth [3], Apriori [4], HMine [11], and HUIM have been designed in recent years for HUI estimation. An Apriori-based itemset mining method was proposed in [6], where a level-wise search method is used to help estimate frequent itemsets. However, the computational cost associated with a large search space and iterative pruning make these methods laborious. The algorithm of FP-growth [3] applies a tree-structure to detect HUIs, whereby an FP-tree structure is first obtained by traversing across data space, searching for frequent itemsets over the tree structure. An improved FP growth method called HMine [11] has been developed, with an additional supplementary pointer-based hyperlink to represent items with a high frequency across the search space. ECLAT [2] was designed by using a vertical database structure called “Transaction ID list.” Unlike conventional methods in which a unit pointer is applied to detect each data element, the ECLAT method exploits the transaction ID to minimize the scanning cost, applying a support count value to each itemset to prune the search space for HUI estimation. Other methods [6,26-29] have used the support value and/or mean profit score for HUI estimation.

Unlike frequency-based itemset mining methods, HUIM methods have desirable performance [13,14] because of their ability to exploit and learn over a large transaction volume, and the corresponding profit makes them suitable for HUI estimation. Two-phase HUI methods [16] begin by identifying itemsets with higher frequencies, followed by an estimation of their utility for final HUI prediction. However, these methods have been criticized for their reliance on a standalone threshold when dealing with non-linear sequential data, which is often inadequate for HUI estimation [16]. Thus, the authors in [16] designed a transaction-weighted utility (TWU) function to minimize the iterative data-scanning cost. This TWU method was further enhanced in [17] by introducing a flexible upper threshold and the capability to skip high-utility itemsets, thereby improving the efficiency of the search space. The authors employed a two-phase method with pruning called the isolated itemset discarding strategy (IIDS) to improve delay performance. Other tree-based methods include incremental high-utility pattern (IHUP) [18], HUP-tree [30], UP-Growth [19], UP-Growth+ [31], mining utility(MU)-Growth [32], and projection-based (PB) indexing approach [33]; however, the computational costs involved and lack of co-occurrence probability remain unexplored. In [20], an HUI miner was proposed using a utility-list (UL) data structure. The UL contains details of the itemset required for pruning, followed by HUI identification. However, despite claims of time efficiency, it cannot address HUI identification using correlated itemsets. In [21], HUI-Miner was designed to reduce the number of joins between the utility and its efficiency functions. The authors employed estimated-utility co-occurrence pruning (EUCP) on a matrix structure called the estimated-utility co-occurrence structure (EUCS). The EUCP encompasses the TWU values of the two item sets arranged in the EUCS matrix. The estimated itemsets were used to prune low-significance items without estimating the utility value. A number of pruning algorithms have been developed to enhance HUIM [34]; however, they fail to address semantic relatedness among itemsets over the search space. The authors in [35] designed efficient itemset mining (EFIM) with predefined upper bounds, considering subtree and local utility factors. To reduce scanning costs, they used the transaction-merging concept. The HMiner [36] model was applied using utility information storage with allied pruning. Approaches such as BAHUI [37], the HUIM-BPSO sign [38], MinHUIs [39], and FHM+ [40] have also been used for HUI estimation.

The two-phase average-utility (TPAU) method [22] applies an average utility-based upper threshold condition, whereby a level-wise search method is applied to enhance the time efficiency. The projection-based average-utility (PBAU) [23] method applies an indexing structure. By applying PBAU, a hard upper limit is defined using the prefix concept [23] to reduce search costs. In [41], a tree-based high-average-utility itemset (HAUI) mining method was designed. The HAUI-Growth model [42] was developed using a tree-based method that efficiently reduces iterative data scanning, and the HAUI miner was designed as a one-phase concept [43] by applying an average utility-based list structure. The efficient high average-utility pattern mining (EHAUPM) in [44] was designed by amalgamating two upper thresholds: looser upper-bound utility (LUB) and revised tighter upper bound (RTUB). Mining of high average-utility (MHAI) [45] retains suitable HUIs based on a high average-utility itemset (HAI) list structure. A closed high-utility itemset (CHUI) with DGU, REG, RML, and DGM was designed in [51] to retain decisive itemsets over the input data structure. In [52], CHUI-Miner was designed as a single-phase model by applying the EU-list structure. This reduces unexpected iterative search costs. The CHIU-Miner, called EFIM-closed [53], was developed with two strictly defined upper thresholds with forward-backward examination. This method uses local and subtree utility values to prune the search space. The CLS-Miner [54] was designed with supplementary coverage and LBP. Despite numerous efforts [46-50], no significant work has examined the probability of coexisting items as HUIs for top-N HUI predictions [55]. Van et al. [56] used FP growth to examine the association between available features to append new features with a certain threshold to perform HUI estimation. In [56], sequential-to-sequential learning methods were applied to top-N target balanced recommendations. In [57], a deep learning method was applied to top-N recommendations, considering the user's interest level and frequency. A deep reinforcement learning model was used in [58] for top-N recommendations. Similarly, interest-related item set learning and similarity measures were applied to perform top-N recommendations [59]. In [60], a trust-aware sequential recommendation model that exploits frequency information was designed. Unfortunately, there has been no viable effort that considers sequential co-occurrence probability or the semantically connected co-occurrence feature to perform top-N HUI recommendations. This was the key driving force in this research.

III. SYSTEM MODEL

This section discusses the proposed convolutional sequential embedding-driven cosine similarity-based MLP learning model for top-N HUI recommendations. As the name indicates, the proposed system comprises three key components: a semantic sequential convolutional encoding (SSCE) also called semantic CNN (SCNN), MLP, and cosine similarity for top-N HUI predictions. The SSCE model comprises a multidirectional filtering-assisted semantic embedding framework that learns over sequential input items or transactions to generate a semantic embedding matrix. In sync with the enterprise application environment, transaction details (user preferences) and allied frequent itemset patterns (buying patterns) are considered. In this manner, unlike conventional deep learning approaches [64-67] that apply local item-wise embedding metrics for learning, the proposed model employs both item-level and corresponding group-level (co-occurrence probability) information to perform top-N HUI prediction or recommendation. This approach enables the model to achieve a higher accuracy while maintaining low computational cost and delays during the scanning of search space (or feature space) to support estimation. An MLP network is applied in conjunction with an adaptive model optimizer (ADAM) learning model to perform training and obtain linear associations among the items, item frequency, and associated co-occurrence probabilities over the sequential transaction inputs. To refine the predicted results and increase accuracy, MLP was deployed in conjunction with the cosine similarity method, to help predict (accurately) the top-N HUIs for recommendation. The proposed model encompasses the following phases:

1. Semantic sequential convolutional encoding (SSCE),
2. Multilayer perceptron (MLP) learning
3. Cosine similarity-driven top-N HUI prediction.

The detailed discussion of these (functional) components is provided as follows:

A. Semantic Sequential Convolutional Encoding (SSCE)

Let the transaction data U have L consecutive itemsets. The objective of this work is to identify the top-T items with both high frequency and co-occurrence probability over sequential transaction data S^u. Here, the transaction dataset has input items $S_1{}^u$, ..., $S_{|S^u|}{}^u$. Therefore, the proposed SSCE model first converts input transactions into an equivalent semantic word-embedding form. More specifically, the Word2Vec embedding method is applied to convert sequential input transactions into equivalent word-embedded metrics (P_M). This metric states the probability of occurrence of each itemset in the transaction data. Once semantic (word-embedded) metrics, also called semantic features, are obtained for the input transaction data, multidirectional convolutional filtering (MDCF) is applied to retrieve the contextual features over the itemsets along with the corresponding co-occurrence probability. Subsequently, the retrieved contextual details are used to perform learning using the MLP network, which, in conjunction with cosine similarity, predicts the top-N HUIs. A schematic of the implementation of the proposed model is shown in Fig. 1.

By applying the aforementioned Word2Vec semantic word-embedding method over the input sequential itemsets and their occurrences across the transaction data, an embedding matrix that possesses embedded information related to the itemsets and their occurrence probability is obtained. Here, the embedding matrix for the input sequential transactions is obtained from the semantic feature space by inputting L-traversed itemsets into a neural network. Let, i denote an item; then the corresponding embedding matrix represent-

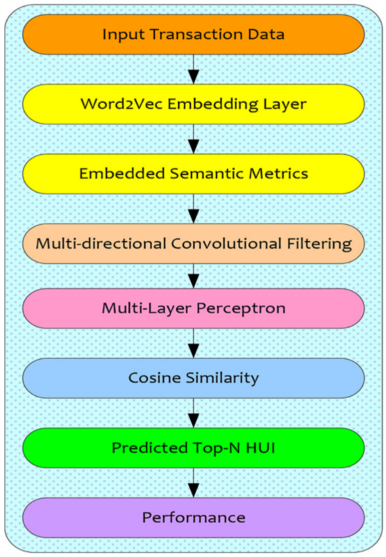

Fig. 1. Proposed top-N HUI prediction model

ing the latent feature is obtained as $Q_i \in R^d$, where d is the latent dimension. In this manner, Word2Vec embedding transforms input sequential transactions containing L itemsets and stacks them to yield an embedding matrix $E^{(u,t)} \in R^{l \times d}$ for u transactions at time t. The embedding metrics thus obtained are defined as:

$$E^{(u,t)} = \begin{bmatrix} Q_{S^u_{t-L}} \\ . \\ . \\ . \\ Q_{S^u_{t-2}} \\ Q_{S^u_{t-1}} \end{bmatrix} \quad (1)$$

In addition to the above derived itemset embedding, the proposed model applies a supplementary embedding model to derive latent information for u users, $P_u \in R^d$, where P_u is the user specific embedding matrix in latent feature space. This is achieved using multidirectional convolutional filtering (MDCF). In CNN-based feature extraction models, convolutional filters retrieve local features over Word2Vec embedding metrics to learn and predict. In our proposed method, the deployed CNN model retrieves the itemset sequential pattern and corresponding co-occurrence probability in $L \times d$ dimensions for L items. In this manner, the latent embedding metrics $L \times d$ or E enable convolution filters to perform frequent sequential pattern estimations for learning. In this study, we applied two distinct CONV filters, a horizontal and vertical filter, to learn sequential patterns over the input embedding metric E. A horizontal filter was applied to generate two group-level sequential itemset patterns signifying the co-occurrence probability. Horizontal filters were used in the form of $h \times d$ matrices, where h is the height ($h = 2$), and d is the width. The horizontal filters choose the itemsets by sliding over the rows of the embedded metrics or latent space E. In contrast, the vertical filter ($L \times 1$ matrix) selects sequential frequent itemsets by sliding over the columns of E.

Let $F^k \in R^{h \times d}$, $1 \le k \le n$ be the horizontal convolutional filter with $h \in \{1, \ldots, L\}$ as the filter height. In four-dimensional latent space $L = 4$, a total number of $n = 8$ filters were deployed. The horizontal filter F^k slides down from top to the bottom of E and continues over all horizontal dimensions of E for item i, $1 \le i \le L-h+1$. In this manner, the interaction outputs the ith convolution value obtained using (2).

$$c_1^k = \varnothing c(E_{i:i+h-1} \odot F^k) \quad (2)$$

where $\odot$ refers to the inner product or multiplication function, whereas $\varnothing c(\cdot)$ represents the activation function. c_1^k yields the inner product between F^k and the submatrix obtained from the ith row to the ($i - h + 1$)-th raw of E (i.e., $E_{i:i+h-1}$). Hence, the final convolutional result for F^k is a vector (3).

$$c^k = [c_1^k \ c_2^k \ \ldots \ c_{l-h+1}^k] \quad (3)$$

We applied a max-pooling layer to the extracted embedding metrics or vector c^k (3). This helps retain the high-resolution features with maximum values, retrieved over the features generated by the convolutional filter. Thus, the proposed method retrieves the set of vectors or significant features over the deployed n filters that eventually yield the output as $o \in c^k$ (4).

$$o = \{\max(c^1), \max(c^2), \ldots, \max(c^n)\} \quad (4)$$

Similar to horizontal filter-driven feature extraction (4), we also applied vertical convolutional filters. Let, $\tilde{n}$ be the total number of vertical filters, then $\tilde{F}^k \in R^{l \times 1}$, where $1 \le k \le \tilde{n}$. Here, each vertical filter $\tilde{F}^k$ correlates the columns of E by traversing from left to right, d-times, obtaining output $\tilde{c}^k$ in (5).

$$\tilde{c}^k = [\tilde{c}_1^k \ \tilde{c}_2^k \ \ldots \ \tilde{c}_d^k] \quad (5)$$

The retrieved feature vector (5) shows that the result is the same as the weighted sum over L rows of the embedding metrics E, obtained by examining the weights $\tilde{F}^k$. Mathematically,

$$\tilde{c}^k = \sum_{l=1}^{L} \tilde{c}_d^k \cdot E_l \quad (6)$$

In (6), E_l represents the lth row of E. Thus, the vertical filters help in learning the embeddings of the L subsequent itemsets having the same value as the weighted sum, in identifying the latent information of L itemsets. Vertical filters generate an entity-level sequential itemset pattern by using weighted sums over the latent information of the traversed itemsets.

Traditional deep learning-based recommendation models use a single weighted sum to predict an itemset. By contrast, we used $\tilde{n}$ global vertical filters to estimate $\tilde{n}$ weighted sums $\tilde{n} \in R^{d\tilde{n}}$ to further search and improve the learning for better prediction results. Mathematically,

$$\tilde{o} = [\tilde{c}^1 \ \tilde{c}^2 \ \dots \ \tilde{c}^{\tilde{n}}] \tag{7}$$

Horizontal and vertical filters were used to extract the global features from the global average max-pooling layer, thereby applying the max function to estimate the global features (8).

$$\Omega_z^j = \max\{\max(o_x^j), \max(\tilde{o}_y^j)\} \tag{8}$$

The proposed model generates a final feature vector by applying a nonlinear-activation-driven fully connected layer. The output of the fully connected layer is given by (9).

$$o_j = \tanh(W_0 \cdot (\tanh(W_h \cdot \Omega_z^j + b_h^j) + b_o^j)) \tag{9}$$

In (9), W_h and W_h represent the weight metrices, whereas b_h^j and b_o^j are the bias components. The feature representation $o_j \in \mathbb{R}^d$ (9) was used as the latent feature for further learning and HUI prediction. To learn the extracted latent features or semantic search space, an MLP network was applied. A brief description of the deployed MLP network follows.

B. Multi-Layer Perceptron (MLP) Layer

As stated earlier, the features from both the horizontal and vertical filters (9) were projected onto a fully connected neural network. The MLP network was applied as a fully convolutional neural network (FCNN) to represent the nonlinear relationship between the itemsets and the associated cooccurrence probability. The deployed MLP network is a multilayer feed-forward neural network (MFNN) that learns nonlinear itemsets, aligning them with co-occurrence probability interactions. In this study, the combined embedding feature (9) was fed into the MLP to obtain the latent embedding feature in (10).

$$\begin{gathered} X_0^p = U_e^p \\ X_1^p = f(W_1^X \cdot X_0^p + B_1^X) \\ X_l^p = f(W_l^X \cdot X_{l-1}^p + B_l^X) \\ X^s = f(W_{Lm}^X \cdot X_{Lm-1}^p + B_{Lm}^X) \end{gathered} \tag{10}$$

In (10), $X_1^p, \dots, X_{Lm}^p$ and $B_1^X, \dots, B_{Lm}^X$ denote the weight and bias vectors, respectively. *Lm* represents the total number of MLP layers, and $f(*)$ represents the rectified linear unit (ReLU) activation function. $X^p \in \mathbb{R}^{m \times c}$ states the latent feature metrics learned by the MLP network. The latent (semantic) embedding metrics for the itemsets were obtained using (11).

$$\begin{gathered} Y_0^p = V_e^p \\ Y_1^p = f(W_1^Y \cdot Y_0^p + B_1^p) \\ Y_l^p = f(W_l^Y \cdot X_{l-1}^p + B_l^p) \\ Y^s = f(W_{Lm}^Y \cdot X_{Lm-1}^p + B_{Lm}^p) \end{gathered} \tag{11}$$

In (11), $W_1^Y, \dots, W_{Lm}^Y$ and $B_1^Y, \dots, B_{Lm}^Y$ state the weights and bias vectors for the deployed MLP network, respectively. Here, $Y^s \in \mathbb{R}^{n_p \times c}$ states the item latent embedding metrics.

The MLP training model over latent metrics was designed based on the loss between the predicted and measured itemset relationships. In this work, the cost function (12) was used to train the network for the prediction of target itemsets q using cost function p (12).

$$\min_{P^p, Q^p, \Theta^p} \sum_{y \in Y^{p+} \cup Y^{p-}} l(y, \hat{y}) + \lambda(\|P^p\|_F^2 + \|Q\|_F^2) \tag{12}$$

In (12), $l(y, \ddot{y})$ represents the loss function between the observed itemsets and the associated co-occurrence probability y and predicted interaction $\ddot{y}$. In (12), Y^{p+} and Y^{p-} refer to the measured and predicted itemsets, respectively. In (12), $\|P^p\|_F^2 + \|Q\|_F^2$ denotes the regularizer, whereas λ is the hyperparameter controlling the level of significance of the regularizer. To alleviate overfitting of Y^+, a fixed number of unobserved itemsets was selected as the negative instance, given by $Y^-_{sampled}$. $Y^-_{sampled}$ to replace Y^-. This helped achieve swift and accurate learning of the input features. Thus, using sequential itemsets, associated co-occurrence probability y_{ij} were obtained (13).

$$y_{ij} = \begin{cases} r_{ij} & \text{if} \quad y_{ij} \in Y^+ \\ 0 & \text{if} \quad Y^-_{sampled} \\ null & \text{otherwise} \end{cases} \tag{13}$$

In this study, a normalized cross-entropy loss function (14) was applied to perform the learning.

$$l(y, \hat{y}) = \frac{y}{\max(R)} \log \hat{y} + \left(1 - \frac{y}{\max(R)}\right) \log(1 - \hat{y}) \tag{14}$$

In (14), $\max(R)$ denotes the highest rating of the itemsets across the input latent embedding matrix. We applied MLP to represent a nonlinear relationship between the itemsets and their co-occurrence (utility). In the MLP network, let the input embedding matrices for the itemsets be $M_{in}^p = [\tilde{U}_p; U^{pd}]$, where $\tilde{U}_p$ represents the combined embedding matrix pertaining to the common itemsets with a high co-occurrence probability p. The embedding matrix of the distinct metrics is U^{pd}. For item v_j, the corresponding embedding matrices in the output layer of the MLP are retrieved according to (15-16).

$$M_i^p = M_{outi}^p = f(\ldots f(f(M_{in_i}^p \cdot W_{M_1}^p) \cdot W_{M_2}^p)) \tag{15}$$

$$N_j^p = N_{outj}^p = f(\ldots f(f(N_{in_j}^p \cdot W_{N_1}^p) \cdot W_{N_2}^p)) \tag{16}$$

In (15-16) the rectified linear unit (ReLU) is represented by the activation function $f(*)$, whereas $W_{M_1}^p$ and $W_{M_2}^p \ldots$, and $W_{M_1}^p$, $W_{N_2}^p \ldots$ are the weights of the multilayer networks in the different layers for $M_{in_i}^p$ and $N_{in_j}^p$, respectively. Finally, in the output layer, the predicted top-N HUI itemsets $\ddot{y}_{i,j}$ between itemsets u_i and v_j are obtained using the cosine similarity function (17).

$$\hat{y}_{i,j}^p = \text{cosine}(M_i^p, N_j^p) = \frac{M_i^p \cdot N_j^p}{\|M_i^p\|\|N_j^p\|} \tag{17}$$

Notably, unlike traditional top-N prediction models, where the learning model (e.g. MLP) predicts the top N selected items, the proposed model applies cosine similarity to the initially predicted items. This helps identify the optimal set of top-N items with high inter-item similarity and corresponding co-occurrence probability. This can be of great significance to enterprises, for highly accurate HUI prediction and inventory management. Thus, by applying this method, the proposed model performs top-N HUI predictions for enterprises or applications.

IV. RESULTS AND DISCUSSION

This study proposes a robust convolutional sequential embedding metric-driven cosine similarity-assisted multilayer perception learning model for top-N high-utility itemset recommendation. Unlike classical approaches, the proposed model exploits both sequential itemset frequency and co-occurrence probability information to perform top-N HUI predictions. First, the sequential transaction data are input into word embedding using the Word2Vec method. Subsequently, the retrieved semantic embedding matrix and multidirectional filters encompassing the horizontal and vertical filters are used to derive global features. The composite embedded matrix features are projected as input onto the MLP layer, which, in conjunction with cosine similarity, performs top-N HUI prediction and recommendations. The deep learning model was executed at an initial learning rate of 0.0001. The overall proposed model was developed using the MATLAB software tool, performing simulation over central processing units arranged using 8 GB RAM and a 3.2 GHz processor. To assess efficacy, different benchmark datasets were applied [61,62]. The dataset encompasses sequential transaction details and itemsets. The proposed model enables intensity estimation over sequential itemsets, as expressed in (18).

$$\text{Frequent Sequential Intensity Per Itemset (FSII)} = \frac{\text{\#Total Counts of the Itemsets}}{\text{\#Number of Transactions}}$$

In (18), the numerator represents the frequency (S_{t-p}^u, ..., S_{t-2}^u, S_{t-1}^u)$\rightarrow S_t^u$ of the itemsets. In this study, itemset frequency and associated significance were measured using a minimum threshold based on support value and confidence (here, 50%) using an L-order Markov chain. In (18), the denominator represents the number of transactions. Thus, by applying (18), the intensity of the sequential itemset pattern is estimated for each input dataset. Data elements or itemsets with an FSII (Frequent Serach Itemsets) smaller than 0.0025 were removed before performing Word2Vec embedding. The input data were divided into two subsets: training and testing, with 70% set aside for the training data and 30% for the test data. Parameters such as *Precision@*, *Recall@N*, maximum average precision (MAP), and MAE were used to examine performance. The derivation of the aforementioned performance parameters are provided in (19-21). The average performances for the different datasets (average d = 25) are shown in Fig. 2.

$$\text{Precision} = \frac{|R \cap \widehat{R_{1:N}}|}{N} \tag{19}$$

$$\text{Recall} = \frac{|R \cap \widehat{R_{1:N}}|}{|R|} \tag{20}$$

$$\text{AP} = \frac{\Sigma_{N=1}^{|\hat{R}|}\text{Precision}(N) \times F(N)}{|\hat{R}|} \tag{21}$$

In the above equations, N denotes the list of N predicted (search) itemsets (i.e., $R_{1:N}$), whereas R signifies the test dataset. In (19), *Precision@N* denotes the precision of N predicted itemsets. To assess the relative performance over different embedding dimensions, we performed simulations using different d values 4, 8, 16, 32, and 64). The results are summarized in Table 1.

Among the state-of-the-art approaches considered for top-N itemset recommendation, a few methods [57-60] have applied deep learning methods for latent feature learning and prediction; however, state-of-the-art methods can be laborious and time consuming. For instance, the authors in [57] applied a CNN followed by a denoising autoencoder (DAE) to perform top-N recommendation, with the CNN applied for latent feature extraction, followed by fractional maximization and DAE-based top-N recommendation. The use of successive deep models can incur high computational costs,

Table 1. Simulated results over different datasets

Dataset	*d*	*Prec.*	*Recall*	*MAP*	*MAE*
	4	0.341	0.014	0.126	0.983
	8	0.420	0.024	0.174	0.734
Tmall	16	0.451	0.042	0.204	0.832
	32	0.563	0.059	0.234	0.721
	64	0.498	0.059	0.293	0.701
	4	0.174	0.039	0.199	0.863
	8	0.192	0.059	0.201	0.723
Gowalla	16	0.198	0.098	0.223	0.758
	32	0.210	1.235	0.224	0.634
	64	0.218	1.249	0.300	0.698
	4	0.356	0.032	0.167	0.899
	8	0.380	0.052	0.198	0.983
Repeat_Buyer	16	0.404	0.159	0.201	0.799
	32	0.428	0.210	0.223	0.652
	64	0.439	0.245	0.247	0.678
	4	0.319	0.087	0.199	0.857
	8	0.334	0.042	0.199	0.699
Kaggle	16	0.3452	0.073	0.259	0.694
	32	0.3569	0.099	0.299	0.621
	64	0.3452	0.139	0.301	0.6873

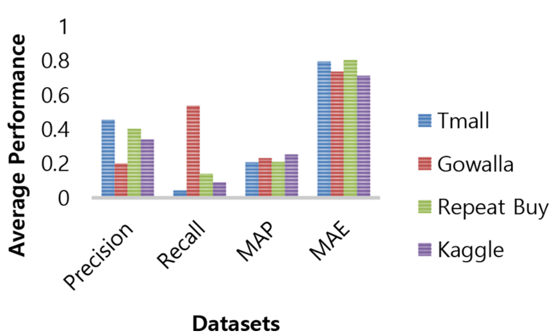

Fig. 2. Average performance over different datasets

exhaust memory, and delay performance. In addition, in observing the empirical simulation results over different datasets, the proposed approach showed a relatively lower error value than \existing methods [57]. The highest precision obtained by DMLR-DAE [57] was 0.32. By contrast, the proposed method achieved the highest precision of 0.5632, which is higher than that of the other existing methods. The MAE obtained using the DMLR-DAE [57] was 0.611 at d = 10 and 0.609 at d = 30. In contrast, the proposed method achieved an average MAE of 0.7610 for the different benchmark datasets.

In addition, the hit rate was estimated for the number of target HUI predicted in the top N predicted lists. Mathematically, HR is measured using (22).

$$HR@K = \frac{No.\ of\ Hits@K}{|T|} \tag{22}$$

In (22), $|T|$ denotes the number of itemsets and associated co-occurrence probabilities or interactions in the test set. In addition, the normalized discounted cumulative gain was applied to assess the hit position by assigning higher scores to hits (especially fort the top K ranks). Mathematically, NDCG@K is obtained as described previously (23).

$$NDCG@K = \sum_{i=1}^{K} \frac{2^{r_i}-1}{\log_2(i+1)} \tag{23}$$

In (23), r_i denotes the ranked relevance of the target item at the i-th position (hence, $r_i = 1$). Otherwise, $r_i = 0$. In addition, the root mean square error (RMSE) was measured using (24), where T states the total number of test ratings, whereas $R^s(i, j)$ represents the real rating, with the measured or predicted rating being $R^s(i, j)$.

$$\text{RMSE} = \sqrt{\frac{1}{T}\sum_{s,i,j}(R^s(i,j)+\hat{R}^s(i,j))^2} \tag{24}$$

The proposed model was simulated using embedding dimensions of $k = \{4, 8, 16, 32, 64\}$, where k represents the latent embedding dimension. The results (Table 2) infer that with increasing k, HR also increases. This implies that with increasing latent dimensions, both HR and NDCG increase.

Table 2. Performance over the different latent embedding dimensions

Latent Dimensions (d)	Data	HR@K	NDCG@K	RMSE
	Tmall	0.453	0.283	1.409
4	Gowalla	0.682	0.410	1.389
	Repeat Buyer	0.543	0.312	1.004
	Kaggle	0.459	0.321	0.953
	Tmall	0.486	0.299	1.091
8	Gowalla	0.672	0.461	1.077
	Repeat Buyer	0.578	0.532	0.997
	Kaggle	0.613	0.523	0.096
	Tmall	0.482	0.299	0.987
16	Gowalla	0.677	0.498	0.988
	Repeat Buyer	0.578	0.487	0.902
	Kaggle	0.689	0.512	0.904
	Tmall	0.501	0.374	0.938
32	Gowalla	0.698	0.460	0.874
	Repeat Buyer	0.582	0.490	0.921
	Kaggle	0.600	0.498	0.184
	Tmall	0.512	0.377	0.880
64	Gowalla	0.701	0.481	0.871
	Repeat Buyer	0.564	0.499	0.911
	Kaggle	0.510	0.412	0.910

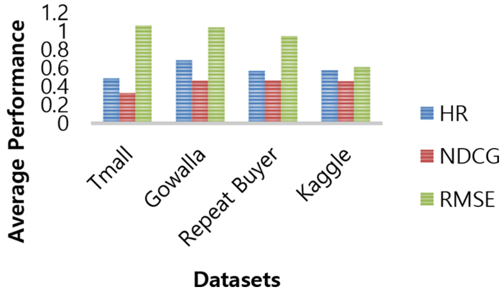

Fig. 3. Average performance over the different datasets

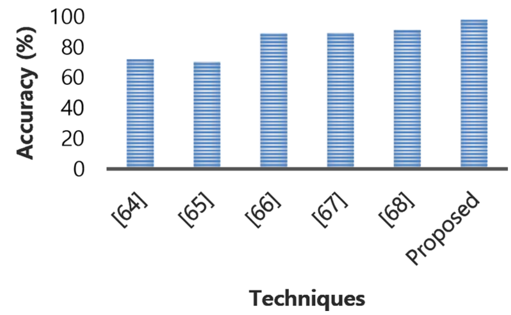

Fig. 4. Top-N HUI prediction accuracy

The mean HR ratio was 0.57923. In contrast, average NDCG was 0.4268 and RMSE was 0.9143. A higher HT rate indicates a higher accuracy and reliability of the solution, whereas, a lower RMSE indicates a better performance of the proposed model. In comparison with state-of-the-art methods [63], the results confirm the superiority of the proposed model for top-N HUI recommendations. The existing method [63] achieved an HR@64 of 0.5126; in contrast, the average HR@K of 0.5720, of the proposed method was significantly higher than that of the state-of-the-art methods. This confirms the robustness of the proposed model for top-N real-time HUI predictions.

Additionally, we compared the proposed model in terms of top-N recommendation accuracy. Table III presents the comparative results of different state-of-the-art techniques. The information in the confusion matrix was used to assess prediction accuracy.

Observing the results, clearly unlike traditional deep learning-based approaches [64-67], where feature extraction is based on CNN, in the proposed method, semantic feature extraction is followed by improved learning-driven feature extraction, which is subsequently processed using cosine-similarity to obtain top-N HUI prediction. Using this approach, the proposed model refined the predicted (top-N HUI) output, achieving higher accuracy, which can easily be visualized in Fig. 4.

Similarly, in terms of cumulative precision, which is derivedfrom the confusion matrix, the proposed itemset top-N HUI prediction model yielded higher precision (97.04%) than the existing method [68] (87%), clearly indicating that the proposed model is more robust compared to the state-of-the art models for enterprise HUI prediction. Conclusions alongwith inferences are presented in the following section.

Table 3. Accuracy Prediction

Source	Techniques	Accuracy (%)
[64]	Federated NN, CNN	71.68
[65]	CNN	70.00
[66]	Random Forest, CNN and XGBoost	88.84
[67]	NN, Logistic Regression	89.00
[68]	Association rule as feature selection over CNN for top-N itemset prediction	91.00
Proposed	convolutional sequential semantic embedding driven MLP with Cosine similarity-based top-N HUI recommendation	97.94

V. CONCLUSION

Most existing HUIM methods often face limitations due to high computational costs, delays, and reduced accuracy when processing extensive sequential transaction data prevalent across various industries. Moreover, the pruning costs and lack of contextual details representing co-occurrence probability limit the efficacy of available state-of-the-art methods. Unlike HUIMs that rely on support value and unit price threshold for pruning, deep learning-based HUI identification is a better alternative. This approach, however, necessitates the exploitation of contextual details, including semantic sequential embedding features, to perform HUI prediction. Motivated by this, this study proposed a novel and robust convolutional sequential semantic embedding-driven multi-layer perceptron learning environment, in sync with cosine similarity, to predict top-N HUI recommendations. In the proposed method, first, two filters are applied along the horizontal and vertical directions simultaneously in convolutional sequential deep learning, to extract the semantic embedding matrix over the transaction details. The use of multiple convolutional filters allows for the retention of a substantial amount of semantic information for further learning and classification. Using the extracted semantic features, the MLP neurocomputing model, which is designed using a ReLU regulation layer and ADAM non-linear optimization function, obtains the linear relationship among the itemsets available across the search space. The proposed MLP model was executed in conjunction with a cosine similarity function to predict the top-N HUI for further recommendations. The use of a semantic embedding matrix with MLP learning and cosine similarity measures helped refine the top-N HUI itemset predictions, which can be highly accurate for any enterprise solution(s). The proposed model exhibited the highest precision of 0.5632, MAE of 0.7610, HR@K of 0.5720, and NDCG@K of 0.4268. Additionally, it exhibited an average MAE of 0.608 over four different datasets, indicating robustness for real-time HUI predictions. The higher cumulative accuracy (97.94%) and precision (97.04%) confirm the efficacy and suitability of the proposed model for real-time enterprise solutions.

REFERENCES

[1] P. Fournier-Viger, J. C. W Lin, R. U. Kiran, Y. S. Koh, and R. Thomas, "A survey of sequential pattern mining," *Data Science and Pattern Recognition*, vol. 1, no. 1, pp. 54-77, Feb. 2017.

[2] M. J. Zaki, "Scalable algorithms for association mining," *IEEE Transactions on Knowledge and Data Engineering*,, vol. 12, no. 3, pp. 372-390, 2000. DOI: 10.1109/69.846291.

[3] J. Han, J. Pei, and M. Kamber, "Data mining: concepts and techniques," *Elsevier*, Amsterda, 2011.

[4] R. Agrawal and R. Srikant, "Mining sequential patterns," In *Proceedings of the Eleventh International Conference on Data Engineering*, Taipei, Taiwan, pp 3-14, 1995. DOI: 10.1109/ICDE.1995.380415.

[5] K. K. Sethi and D. Ramesh, "A fast high average-utility itemset mining with efficient tighter upper bounds and novel list structure," *The Journal of Supercomputing, Springer*, vol. 76, no. 12, pp. 10288-10318, Mar. 2020. DOI: 10.1007/s11227-020-03247-5..

[6] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," in *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, Washington, USA, pp 207-216, 1993. DOI: 10.1145/170035.170072.

[7] R. Agrawal and R. Srikant; "Fast algorithms for mining association rules," in *Proceedings of the 20th International Conference on Very Large Data Bases*, pp 487-499, 1994.

[8] P. Fournier-Viger, J. C.-W. Lin, B. Vo, T. T. Chi, J. Zhang, and H. B. Le, "A survey of itemset mining," *WIREs Data Mining and Knowledge Discovery*, vol. 7, no. 4, Apr. 2017. DOI: 10.1002/widm.1207.

[9] T. Wei, B. Wang, Y. Zhang, K. Hu, Y. Yao, and H. Liu, "FCHUIM: Efficient Frequent and Closed High-Utility Itemsets Mining," *IEEE Access*, vol. 8, pp. 109928-109939, 2020. DOI: 10.1109/ACCESS.2020.3001975.

[10] G. Grahne and J. Zhu, "Fast algorithms for frequent itemset mining using fp-trees," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 10, pp. 1347-1362, Oct. 2005. DOI: 10.1109/TKDE.2005.166.

[11] J. Pei, J. Han, H. Lu, S. Nishio, S. Tang, and D. Yang, "H-mine: hyper-structure mining of frequent patterns in large databases," in *ICDM 2001, Proceedings IEEE International Conference on Data Mining*, San Jose, USA, pp. 441-448, 2001. pp 441-448, DOI: 10.1109/ICDM.2001.989550..

[12] V. S. Tseng, B.-E. Shie, C.-W. Wu, and P. S. Yu, "Efficient algorithms for mining high utility itemsets from transactional databases," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 8, pp. 1772-1786, Aug. 2013. DOI: 10.1109/TKDE.2012.59.

[13] R. Chan, Q. Yang, Y. D. Shen, "Mining high utility itemsets," in *IEEE International Conference on Data Mining*, Melbourne, Florida, pp 19-26, 2003, DOI: 10.1109/ICDM.2003.1250893.

[14] H. Yao, H. J. Hamilton, C. J. Butz; "A foundational approach to mining itemset utilities from databases," in *Proceedings of the 2004 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics*, Lake Buena Vista, USA, pp 482-486 2004. DOI: 10.1137/1.9781611972740.51.

[15] W. Song, Y. Liu, and J. Li, "BAHUI: Fast and memory efficient mining of high utility itemsets based on bitmap," *International. Journal of. Data Warehousing Mining*, vol. 10, no. 1, pp. 1-15, Jan. 2014. DOI: 10.4018/ijdwm.2014010101.

[16] Y. Liu, W. K. Liao, and A. N. Choudhary, "A two-phase algorithm for fast discovery of high utility itemsets," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (*PAKDD*), Hanoi, Vietnam, pp. 689-695, 2005. DOI: 10.1007/11430919_79.

[17] Y. C. Li, J. S. Yeh, and C. C. Chang, "Isolated items discarding strategy for discovering high utility itemsets," *Data Knowledge Engineering*, vol. 64, no. 1, pp.198-217, Jan. 2008. DOI: 10.1016/j.datak.2007.06.009.

[18] C. F. Ahmed, S. K. Tanbeer, B. S. Jeong, and Y. K. Lee, "Efficient tree structures for high utility pattern mining in incremental databases," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 12, pp. 1708-1721, Dec. 2009. DOI: 10.1109/TKDE.2009.46.

[19] V. S. Tseng, C. W. Wu, B. E. Shie, and P. S. Yu, "UP-growth: an efficient algorithm for high utility itemset mining," in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington DC, USA, pp 253-262, 2010. DOI: 10.1145/1835804.1835839

[20] M. Liu and J. Qu, "Mining high utility itemsets without candidate generation," in *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, Maui, USA, pp 55-64, 2012. DOI: 10.1145/2396761.2396773.

[21] P. Fournier-Viger, C. W. Wu, S. Zida, and V. S. Tseng, "FHM: faster high-utility itemset mining using estimated utility co-occurrence pruning," in *Foundations of Intelligent Systems: 21st International Symposium, ISMIS 2014,*, Roskilde, Denmark, pp 83-92, 2014. DOI: 10.1007/978-3-319-08326-1_9.

[22] T. P. Hong, C. H. Lee, and S. L. Wang, "Effective utility mining with the measure of average utility," *Expert Systems Applications*, vol. 38, no. 7, pp. 8259-8265, 2011. DOI: 10.1016/j.eswa.2011.01.006.

[23] G. C. Lan, T. P. Hong, and V. S. Tseng, "A projection-based approach for discovering high average utility itemsets," *Journal of Information Science and Engineering*, vol. 28, no. 1, pp. 193-209, 2012.

[24] C. W. Lin, T. P. Hong, and W. H. Lu ,"Efficiently mining high average utility itemsets with a tree structure," in *Asian Conference on Intelligent Information and Database Systems*, Hue City, Vietnam,, pp 131-139, 2010. DOI: 10.1007/978-3-642-12145-6_14.

[25] A. Y. Peng, Y. S. Koh, and P. Riddle, "mHUIMiner: A fast high utility itemset mining algorithm for sparse datasets," in *Advances in Knowledge Discovery and Data Mining*, Jeju, Korea, pp. 196-207, 2017. DOI: 10.1007/978-3-319-57529-2_16.

[26] J. Pei, J. Han, and L. V. Lakshmanan, "Pushing convertible constraints in frequent itemset mining," *Data Mining and Knowledge Discovery*, vol. 8, no. 3, pp. 227-252, May 2004. DOI: 10.1023/B:DAMI.0000023674.74932.4c.

[27] K. K. Sethi and D. Ramesh, "HFIM: a Spark-based hybrid frequent itemset mining algorithm for big data processing," *The Journal of Supercomputing*, vol. 73, no. 8, pp. 3652-3668, Jan. 2017. DOI: 10.1007/s11227-017-1963-4.

[28] G. Pyun, U. Yun, and K. H. Ryu, "Efficient frequent pattern mining based on linear prefix tree," *Knowledge-Based Systems*, vol. 55, pp. 125-139, Jan. 2014. DOI: 10.1016/j.knosys.2013.10.013.

[29] U. Yun, G. Lee, and K. H. Ryu, "Mining maximal frequent patterns

by considering weight conditions over data stream," *Knowledge. Based Systems* vol. 55, pp. 49-65, Jan. 2014. DOI: *10.1016/j.knosys. 2013.10.011*.

[30] C. W. Lin, T. P. Hong, and W. H. Lu, "An effective tree structure for mining high utility itemsets," *Expert Systems with Applications*, vol. 38, no. 6, pp. 7419-7424, Jun. 2011. DOI: 10.1016/j.eswa.2010.12.082.

[31] V. S. Tseng, B. E. Shie, C. W. Wu, and P. S. Yu, "Efficient algorithms for mining high utility itemsets from transactional databases,*" IEEE Trans Knowl Data Engineering*, vol. 25, no. 8, pp. 1772-1786, Aug. 2013. DOI: 10.1109/TKDE.2012.59.

[32] U. Yun, H. Ryang, and K. H. Ryu, "High utility itemset mining with techniques for reducing overestimated utilities and pruning candidates," *Expert Systems with Applications*, vol. 41, no. 8, pp. 3861-3878, 2014. DOI: 10.1016/j.eswa.2013.11.038

[33] G. C Lan, T. P. Hong, and V. S Tseng, "An efficient projection-based indexing approach for mining high utility itemsets," *Knowledge Information Systems*, vol. 38, no. 1, pp. 85-107, Aug. 2013. DOI: 10.1007/s10115-012-0492-y.

[34] S. Krishnamoorthy "Pruning strategies for mining high utility itemsets," *Expert Systems with Applications*, vol 42, no. 5, pp. 2371-2381, Apr. 2015. DOI: 10.1016/j.eswa.2014.11.001.

[35] S. Zida, P. Fournier-Viger, J. C. W. Lin, C. W. Wu, and V. S. Tseng, "EFIM: a highly efficient algorithm for high-utility itemset mining," in *14th Mexican International Conference on Artificial Intelligence*, Cuernavaca, Mexico, pp 530-546, 2015. DOI: 10.1007/978-3-319-27060-9_44.

[36] S. Krishnamoorthy, "HMiner: efficiently mining high utility itemsets," *Expert Systems with Applications*, vol. 90, pp. 168-183, Dec. 2017. DOI: 10.1016/j.eswa.2017.08.028.

[37] W. Song, Y. Liu, J. Li, "BAHUI: fast and memory efficient mining of high utility itemsets based on bitmap,". *International Journal Data Warehouse Mining (IJDWM)*, vol. 10, no. 1, pp. 1-15, Jan. 2014. DOI: 10.4018/ijdwm.2014010101.

[38] J. C. W. Lin, L. YangL, P. Fournier-Viger, J. M. T. Wu, T. P.. Hong, L. S. L. Wang, and J. Zhan, "Mining high utility itemsets based on particle swarm optimization," *Engineering Applications Artificial Intellegence*, vol. 55, pp. 320-330, Oct. 2016. DOI: 10.1016/ j.engappai.2016.07.006.

[39] P. Fournier-Viger, J. C. W. Lin, C. W.. Wu, V. S. Tseng, and U. Faghihi "Mining minimal high-utility itemsets," in *International Conference on Database and Expert Systems Applications*, Porto, Portugal, pp 88-101, 2016. DOI: 10.1007/978-3-319-44403-1_6.

[40] P. Fournier-Viger, J. C. W. Lin, Q. H. Duong, and T. L. Dam "FHM +: faster high-utility itemset mining using length upper-bound reduction," in *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, Morioka, Japan,pp 115-127, 2016. DOI: 10.1007/978-3-319-42007-3_11.

[41] T. Lu, B. Vo, H. T. Nguyen, and T. P. Hong, "A new method for mining high average utility itemsets," in *13th IFIP TC 8 International Conference, Computer Information Systems and Industrial Management 2014*, Ho Chi Minh City, Vietnam,, pp 33-42, 2014. DOI: 10.1007/978-3-662-45237-0_5.

[42] C. W. Lin, T. P. Hong, W. H. Lu, "Efficiently mining high average utility itemsets with a tree structure", in *Asian Conference on Intelligent Information and Database Systems*, Hue City, Vietnam,, pp 131-139, 2010. DOI: 10.1007/978-3-642-12145-6_14.

[43] J. C. W. Lin, T. Li, P. Fournier-Viger, T. P. Hong, J. Zhan, and M. Voznak, "An efficient algorithm to mine high average-utility itemsets," *Advanced Engineering Information*, vol. 30, no. 2, pp. 233-243, Apr. 2016.DOI : 10.1016/j.aei.2016.04.002.

[44] J. C. W. Lin, S. Ren, P. Fournier-Viger, and T. P. Hong, "EHAUPM: efficient high average-utility pattern mining with tighter upper bounds," *IEEE Access*, vol. 5, pp. 12927–12940, 2017. DOI: 10.1109/ACCESS.2017.2717438.

[45] U. Yun and D. Kim "Mining of high average-utility itemsets using novel list structure and pruning strategy," *Future Generation Computer Systems*, vol. 68, pp. 346-360, Mar. 2017. DOI: 10.1016/ j.future.2016.10.027.

[46] J. C. W. Lin, S. Ren, P. Fournier-Viger, T. P. Hong, J. H Su, and B. Vo, "A fast algorithm for mining high average-utility itemsets," *Application Intellegence Systems*, vol. 47, no. 2, pp. 331-346, Mar. 2017. DOI: 10.1007/s10489-017-0896-1.

[47] J. C. W. Lin, S. Ren, and P. Fournier-Viger, "MEMU: more efficient algorithm to mine high average utility patterns with multiple minimum average-utility thresholds," *IEEE Access*, vol. 6, pp. 7593-7609, 2018. DOI: 10.1109/ACCESS.2018.2801261.

[48] J. M. T. Wu, J. C. W. Lin, M. Pirouz, and P. Fournier-Viger, "TUB-HAUPM: tighter upper bound for mining high average-utility patterns," *IEEE Access*, vol. 6, pp. 18655-18669, 2018. DOI: 10.1109/ACCESS.2018.2820740.

[49] T. Truong, H. Duong, B. Le, and P. Fournier-Viger, "Efficient vertical mining of high average-utility itemsets based on novel upper-bounds,*" IEEE Transactions on Knowledge and Data Engineering*, 2018, vol. 31, no. 2, pp. 301-314, Feb. 2019. DOI: 10.1109/TKDE.2018.2833478.

[50] T. Truong, H. Duong, B. Le, P. Fournier-Viger, and U. Yun, "Efficient high average-utility itemset mining using novel vertical weak upper-bounds," *Knowledge-Based Systems,* vol. 183, pp. 104847, Nov. 2019. DOI: 10.1016/j.knosys.2019.07.018.

[51] V. S. Tseng, C.-W. Wu, P. Fournier-Viger, and P. S. Yu, "Efficient algorithms for mining the concise and lossless representation of high utility itemsets," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 3, pp. 726-739, Mar. 2015. DOI: 10.1109/ TKDE.2014.2345377.

[52] C.-W. Wu, P. Fournier-Viger, J.-Y. Gu, and V. S. Tseng, "Mining closed+ high utility itemsets without candidate generation," in *2015 Conference on Technologies and Applications of Artificial Intelligence (TAAI)*, Tainan, Taiwan, pp. 187-194, 2015. DOI: 1109/ TAAI.2015.7407089.

[53] P. Fournier-Viger, S. Zida, J. C.-W. Lin, C.-W. Wu, and V. S. Tseng, "EFIM-closed: Fast and memory efficient discovery of closed high-utility itemsets," in *12th International Conference, Machine Learning and Data Mining in Pattern Recognition*, New York, USA, pp. 199-213, 2016. DOI: 10.1007/978-3-319-41920-6_15.

[54] T.-L. Dam, K. Li, P. Fournier-Viger, and Q.-H. Duong, "CLS-Miner: Efficient and effective closed high-utility itemset mining," *Frontiers Computer. Science*., vol. 13, no. 2, pp. 357-381, Apr. 2019, DOI: 10.1007/s11704-016-6245-4.

[55] L. T. Hong Van, P. Van Huong, L. D. Thuan, and N. Hieu Minh, "Improving the feature set in IoT intrusion detection problem based on FP-growth algorithm," in *International Conference on Advanced Technologies for Communications (ATC)*, Nha Trang, Vietnam, pp. 18-23, 2020. DOI: 10.1109/ATC50776.2020.9255431.

[56] M. I. M. Ishag, K. H. Park, J. Y. Lee, and K. H. Ryu, "A pattern-based academic reviewer recommendation combining author-paper and diversity metrics," *IEEE Access*, vol. 7, pp. 16460-16475, 2019. DOI: 10.1109/ACCESS.2019.2894680.

[57] X. Wang, Y. Sheng, H. Deng and Z. Zhao, "Top-N-targets-balanced recommendation based on attentional sequence-to-sequence learning," *IEEE Access*, vol. 7, pp. 120262-120272, 2019. DOI: 10.1109/ ACCESS.2019.2937557.

[58] W. Zhou, J. Li, M. Zhang, Y. Wang, and F. Shah, "Deep learning modeling for top-N recommendation with interests exploring," *IEEE Access*, vol. 6, pp. 51440-51455, 2018. DOI: 10.1109/ACCESS.

2018.2869924.

[59] V. Baghi, S. M. Seyed Motehayeri, A. Moeini, and R. Abedian, "Improving ranking function and diversification in interactive recommendation systems based on deep reinforcement learning," *26th International Computer Conference, Computer Society of Iran (CSICC)*, Tehran, Iran, pp. 1-7, 2021. DOI: 10.1109/CSICC52343. 2021.9420615.

[60] J. Lv, B. Song, J. Guo, X. Du, and M. Guizani, "Interest-related item similarity model based on multimodal data for top-N recommendation," *IEEE Access*, vol. 7, pp. 12809-12821, 2019. DOI: 10.1109/ ACCESS.2019.2893355.

[61] Y. Zeng, Z. Qu, and B. Zhou, "Trust-aware sequence recommendation based on attention mechanism, in *IEEE 5th international conference on cloud computing and big data analytics*, Chengdu, China, pp. 45-50, 2020. DOI: 10.1109/ICCCBDA49378. 2020.9095580.

[62] IJCAI. Repeat Buyers Prediction Competition [Online], Available: https://ijcai-15.org/repeat-buyers-prediction-competition/.

[63] Kaggle. Association Rules Mining/Market Basket Analysis [Online], Available: https://www.kaggle.com/datatheque/association-rules-mining-market-basket-analysis.

[64] H.-J. Xue, X. Dai, J. Zhang, S. Huang, and J. Chen, "Deep matrix factorization models for recommender systems," in *Proceedings of 26th International Joint Conference on Artificial Intelligence (IJCAI-17)*, pp. 3203-3209, 2017.

[65] V. Umayaparvathi and K. Iyakutti, "Automated feature selection and churn prediction using deep learning models," *International Research Journal of Engineering and Technology (IRJET*), vol. 4, no. 3, pp. 1846-1854, Mar. 2017.

[66] P. Ghadekar and A. Dombe, "Image- Based Product Recommendations Using Market Basket Analysis," in *2019 5th International Conference On Computing, Communication, Control And Automation (ICCUBEA)*, Pune, India, pp. 1-5, 2019. DOI: 10.1109/ICCUBEA47591.2019.9128524.

[67] O. F. Seymen, O. Dogan, and A. Hiziroglu, "Customer churn prediction using deep learning," in *Proceedings of the 12th International Conference on Soft computing and Pattern Recognition (SoCPaR 2020)*, Online, pp. 520-529, 2021. DOI: 10.1109/ ICCUBEA47591.2019.9128524.

[68] N. Pazhaniraja, S. Sountharrajan, E. Suganya, and M. Karthiga, "Top 'N' Variant Random Forest Model for High Utility Itemsets Recommendation," *EAI Endorsed Transactions on Energy Web*, | vol. 8, no. 35, pp. 1-7, Jan. 2021. DOI: 10.4108/eai.25-1-2021. 168225.

Siva S

He received his Master of Computer Applications (M.C. A) from the University of Madras, India, in 2004. With over 16 years of experience in software research and development, he is currently employed as a senior engineering manager in a company in Bangalore, India, where a healthcare mobile solution is being developed using IoT and AI. Additionally, he is pursuing his Ph.D. at REVA University, Bangalore, India, and holds membership in IEEE.

Dr. Shilpa Chaudhari

Currently, she is working as an Associate Professor at the Department of CSE, MSRIT, Bangalore. She has been a technology educator and corporate trainer since 1999. Over the last 18 years, she has held various academic positions in technical institutes in Maharashtra and Karnataka. Her areas of research and teaching include network security, RTOS, computational intelligence, wireless networks, and embedded system development. She is an active professional member of the Computer Society of India (CSI), having been a Life member since 2013, and also holds membership with IEEE Member (#94611631), Bangalore Section.