IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 3, April 2024

An ORL-DLNN and IFIM-MST Framework for Data Quality Improvement in Modern Master Data Management

Shantanu Indra
Austin, Texas, United States of America

Abstract: *The process of generating and maintaining a master record for each person in a business is called Master Data Management (MDM). But, during Modern Master Data Management (MMDM), the prevailing methodologies did not focus on data loss. Hence, an Optimal with Recover Layer-based Deep Learning Neural Network (ORL-DLNN) and Indexed Fisher Information Matrix-based Minimum Spanning Tree-based Modern Master Data Management (IFIM-MST-based MMDM) is proposed in this paper. The source data is taken and pre-processed during training. Then, the products are clustered, and the attributes are extracted. Thereafter, by utilizing Dynamic Function-based Gold Rush Optimization (DF-GRO), the optimal attributes are selected. Afterward, for performing format classification with error detection, ORL-DLNN is introduced. Then, by utilizing the Reconciliation Rule (RR), the detected errors are corrected. After that, by utilizing the IFIM-MST, the text records are stored in the primary drive and the video and images are stored in the external drive. The details are entered by the information stewards into the classifier during testing. The data will be corrected if correction is present. Then, for verifying whether the data must be stored or not, some conditions are checked. Hence, the outcomes illustrated that the proposed system obtained a high accuracy of 98.3%, thus outperforming the existing techniques.*

Keywords: Modern MDM (MMDM), Dynamic Function-based Gold Rush Optimization (DF-GRO), Indexed Fisher Information Matrix-based Minimum Spanning Tree (IFIM-MST), Exponential Probability Function-based Affinity Propagation (EPF-AP), Reconciliation Rule (RR), Information stewards, and Business.

I. INTRODUCTION

The Internet application entered into a fresh big data era with the advancement of technology (Xiaojing&Minghai, 2021) (Sawadogo & Darmont, 2021). The technological development forced the organization to update new systems (Miklosik& Evans, 2020). The evolution of these technologies led to the generation of huge amounts of data from touchpoints (Warnke-Sommer&Damann, 2021). One of the valuable resources for corporate success is Data (Mositsa et al., 2023). Hence, to maintain data quality, consistency, and uniqueness, the MMDM system for larger data is very essential (Behera& Panda, 2023).

The process of creating a single master record of customers, products, and business entities is called MMDM (Jaskó et al., 2020). By ensuring the accuracy as well as consistency of the data, MMDM improves data quality (Yaqoob et al., 2022); also, it is used to establish a single source of truth for critical data (Pansara, 2020b) (Hannila et al., 2022). For improving data security and efficiency, various prevailing works concentrated on the MMDM with Machine Learning (ML) techniques. However, owing to integrating disparate data sources, the prevailing approaches had complexity issues. Therefore, in this work, an efficient ORL-DLNN and IFIM-MST-based MMDM are proposed.

1.1. Problem Statement

Limitations of prevailing frameworks are;

- During the MMDM process, none of the prevailing works focused on data loss.
- Owing to manual processing, the error rate prevailing (Song et al., 2023) is high during data storage.
- The prevailing research (Pansara, 2020a) considers the group of data, thus creating complexities.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 3, April 2024

- Redundant information is repeatedly stored on the existing drive (Kuznetsov et al., 2022).
- Most of the conventional works have an unaligned data process.

The key objectives of the proposed approach are,

- To avoid data loss during MMDM, external storage drives are used
- For effectively classifying the formats and detecting the errors, the ORL-DLNN classifier is introduced.
- EPF-AP-based product clustering is done.
- To avoid the storage of redundant information, an efficient condition-based approach is used.
- For aligning the data, pre-processing is performed.
- The paper is systemized as: in section 2, the prevailing works are demonstrated; in section 3, the proposed methodology is conveyed; in section 4, the outcomes are exemplified; lastly, in section 5, the proposed work is concluded.

II. LITERATURE SURVEY

(Song et al., 2023) propounded a master data-based production planning forecasting system. Here, to predict the dynamic data based on the baseline information, the M5 Prime (M5P) algorithm was utilized. For efficient production planning, the research combined the ML approaches. Yet, in the research, a manual process was performed during storage by which the error rate was increased.

(Pansara, 2020a) proffered graph databases and MDM. Primarily, the integration of graph databases with MDM systems was investigated by the presented research. Moreover, this work uncovered a wealth of insights into the optimization of relationships. Hence, the dynamics of data management were effectively reshaped by this work. Also, the research considered the group of data for the MDM process, thus creating complexities.

(Kuznetsov et al., 2022) explicated an MMDM platform. Here, a unidata platform was introduced for creating various MDM solutions, which were constructed for a specific domain as well as case requirements. Moreover, for effective MDM, the data storage approaches were presented. In addition, this work repeatedly stored redundant information in the drive, which might result in storage issues.

(Akter et al., 2020) explained the building of dynamic service analytics capabilities for the digital marketplace. Here, for managing environmental dynamism brought upon by big data, the dimensions were identified. Moreover, the presented framework enclosed management, technology, and service innovation capabilities. Nevertheless, the research did not address the intricacies involved with service systems, thus diminishing performance.

(Ying et al., 2021) supervised the management of big data in the retail industry of Singapore. For managing the big data, this work utilized the fork in a road strategy. Here, to reduce the barriers, the course of hiring skilled resources within different analytics was chosen. Therefore, the big data management analytics' feasibility was supremely clarified by the research. However, this model utilized manual processing in big data management, which created complexities.

III. PROPOSED MMDM SYSTEM

For performing the MMDM process, the ORL-DLNN and IFIM-MST are employed in the proposed MMDM system. Figure 1 displays the proposed model's architectural diagram

3.1 Training phase

The processes comprised in the training phase are explained below.

3.1.1 Source Data

Primarily, from business applications, namely Dynamic 365, custom apps, cloud apps, and legacy apps, the data are extracted and provided as input to the MMDM system. The extracted input data is expressed as,

$$\wp_\ell \xrightarrow{input} [\wp_1, \wp_2, \wp_3, \ldots\ldots, \wp_k] \quad Where \quad \ell \to (1,2,\ldots,k) \tag{1}$$

Here, the total number of $\wp_\ell$ is exemplified as $\wp_k$.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 3, April 2024

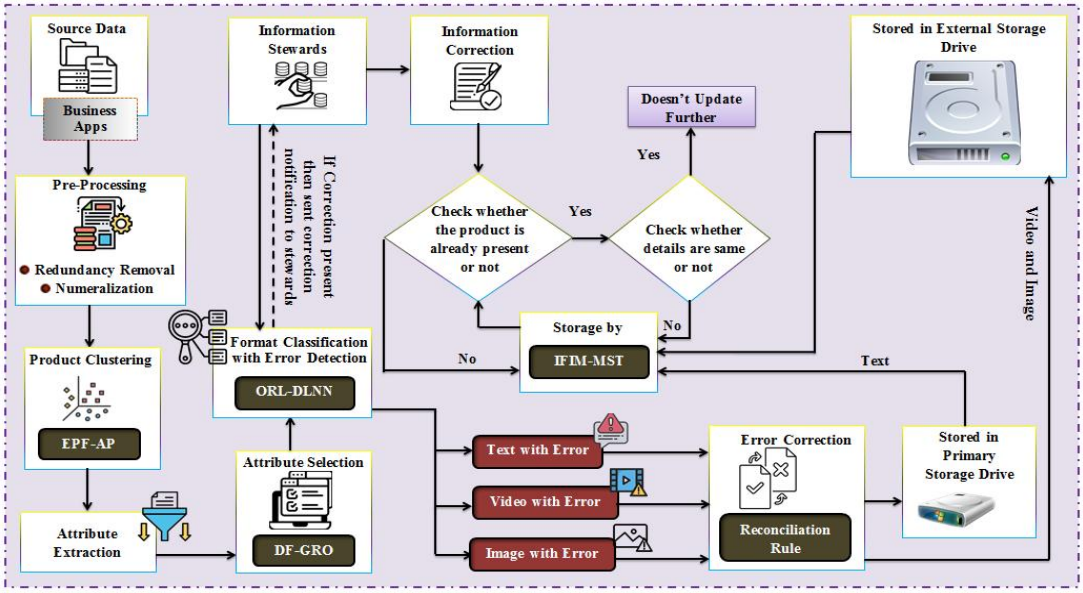


Fig 1: Architectural diagram of the proposed model

3.1.2 Pre-Processing

Then, to reduce the complexity of MMDM, the products are clustered by utilizing Exponential Probability Function-based Affinity Propagation (EPF-AP). For statistics-related values, superior outcomes can be provided by Affinity Propagation (AP). But, for updating the clusters, AP used a log-probability function that isn't highly supported for large-scale data. The Exponential Probability Function (EPF) is added with AP to overcome this issue. The data points (ϑ_r) are initialized, which are provided by,

$$\vartheta_r(\Re_b^{\bullet}) = (\vartheta_1, \vartheta_2, \vartheta_3, \ldots, \vartheta_\kappa) \quad Here \quad r = (1,2,\ldots,\kappa) \tag{3}$$

Afterward, the similarity and negative squared distance are estimated. Then, the EPF-AP has two message-passing steps: the responsibility matrix $R(u,w)$ and the availability matrix $A(u,w)$. Thereafter, both matrices are initialized to zero. For improving the cluster updation process, the EPF function $S(u,w)$ is added here, which is determined as,

$$S(u,w) = \begin{cases} \tilde{\lambda} e^{-\tilde{\lambda} r'} & r' \geq 0 \\ 0 & r' < 0 \end{cases} \tag{4}$$

Here, the rate parameter is exemplified as $\tilde{\lambda}$, and the random variable is signified as r'. Primarily, responsibility matrix $R(u,w)$ updation is performed by the EPF-AP, which is articulated as,

$$R(u,w) \leftarrow S(u,w) - \max_{w' \neq w} \{A(u,w') + S(u,w')\} \tag{5}$$

Here, the matrix indices are notated as u, v, and w. After that, the availability matrix is updated, which is determined by,

$$A(u,w) \leftarrow \min\left(0, R(w,w) + \sum_{u' \notin \{u,w\}} \max(0, R(u',w))\right) \quad for \quad u \neq w \quad and$$
$$A(w,w) \leftarrow \sum_{u' \neq w} \max(0, R(u',w)) \tag{6}$$

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 3, April 2024

This updation is continued until either of the clusters remains unchanged. Lastly, the clustered data (Ω_e^*) is articulated as,

$$\Omega_e^* \xrightarrow{clustered} \left\{\Omega_1^* + \Omega_2^* + \Omega_3^* + \ldots\ldots + \Omega_{\hbar}^*\right\} \tag{7}$$

Here, the total number of Ω_e^* is implied as $e = (1,2,\ldots,\hbar)$.

3.1.4 Attribute Extraction

Then, from the Ω_e^*, the attributes, namely customer id, customer initial and last name, customer mail id, product manufacturing date, and utilization area are extracted, which are provided by,

$$\chi_{\varepsilon}^{att} \rightarrow \left(\chi_1^{att}, \chi_2^{att}, \chi_3^{att}, \ldots\ldots, \chi_{\phi}^{att}\right) \quad where \quad \varepsilon = (1,2,\ldots,\phi) \tag{8}$$

Here, the extracted attributes are indicated as χ_{ε}^{att}.

3.1.5 Attribute Selection

Thereafter, by utilizing DF-GRO, the optimal attributes are selected from χ_{ε}^{att}. Gold Rush Optimization (GRO) effectively gives solutions for real-world problems. Yet, the convergence does not rely on the dynamic convergence component. Therefore, the dynamic function is added with GRO. Primarily, the gold-searching operators' population is initialized. The initialized population (P_m) is considered as the extracted attributes, which is expressed as,

$$P_m(\chi_{\varepsilon}^{att}) \xrightarrow{population} \langle P_1, P_2, P_3, \ldots\ldots, P_{\upsilon}\rangle \quad where \quad m = (1,2,\ldots,\upsilon) \tag{9}$$

Every operator stands randomly $(rand)$ in one spot in the local search space (L), which is articulated as,

$$L(s) = lb_s + (ub_s - lb_s) * rand \tag{10}$$

Here, lb_s and ub_s imply the lower and upper bound, correspondingly. Next, by considering the classification accuracy as a maximum $(\max(Acc))$, the fitness function (F) is computed, which is formulated as,

$$F(s) = \max(Acc) * \frac{K_s}{\varpi_s} * \frac{sound(highest\ volume) - sound(s)}{(sound(highest\ volume) - sound(lowest\ volume) + \varepsilon')} \tag{11}$$

Here, the coefficients are notated as K_s and ϖ_s, and the small positive number is signified as ε'. Subsequently, each operator can move towards or against the loud sound. For improving the convergence problem, the dynamic function (Df) is included here, which is determined by,

$$Df = (Df_{ub} - Df_{lb}) \times \frac{\max_{it} - it}{\max_{it}} + Df_{lb} \tag{12}$$

Here, the upper and lower bounds of Df are exemplified as Df_{ub} and Df_{lb}. The updated location $(Nl(s))$ is given as,

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 3, April 2024

$$Nl(s) = L(s) + Z \bullet \left[\left(F(t) - F(s)\right) * \left(L(t) - L(s)\right) * rand\right] \bullet Df \tag{13}$$

Here, the movement direction is notated as Z . Thereafter, the new location is updated by the below equation,

$$Nl'(s) = \begin{cases} rand < \beta & neighboring\ location\ choosed \\ \beta < rand < \gamma & new\ location\ randomly\ selected \\ \gamma < rand & don't\ move \end{cases} \tag{14}$$

This location update process is continued until the final iteration. Eventually, the selected attributes $\left(B_{\kappa'}\right)$ are indicated by,

$$B_{\kappa'} \xrightarrow{selected} \left\{B_1, B_2, B_3, \ldots\ldots, B_{\varphi}\right\} \quad where \quad \kappa' \to (1,2,\ldots,\varphi) \tag{15}$$

The pseudocode for DF-GRO is provided below,

PSEUDOCODE FOR DF-GRO

Input: Extracted attributes $\left(\chi_{\varepsilon}^{att}\right)$

Output: Selected attributes $\left(B_{\kappa'}\right)$

Begin

Initialize population $\left(P_m\right)$

While $\left(it \le \max_{it}\right)$

For each $\left(P_m\right)$

Compute $L(s) = lb_s + (ub_s - lb_s) * rand$

Estimate $F(s)$

$$Df = \left(Df_{ub} - Df_{lb}\right) \times \frac{\max_{it} - it}{\max_{it}} + Df_{lb}$$

Discover

Update new location

$$Nl(s) = L(s) + Z \bullet \left[\left(F(t) - F(s)\right) * \left(L(t) - L(s)\right) * rand\right]$$

Evaluate movement direction

Update new location

If $rand < \beta$

$neighboring\ location\ choosed$

Else If $\beta < rand < \gamma$

$new\ location\ randomly\ selected$

Else

$don't\ move$

End If

End For

End While

Obtain $\left(B_{\kappa'}\right)$

End

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 3, April 2024

After that, the selected attributes are subjected to the ORL-DLNN classifier.

3.1.6 Format classification with error detection

To avoid error rates, the format of $B_{\kappa'}$ is classified with errors using ORL-DLNN. For normal numerical values, Deep Learning Neural Network (DLNN) is highly suitable. Moreover, DLNN had exploding and dead neuron issues. The DF-GRO-based weight initialization and recovery layer is used in DLNN to solve the issues. Figure 2 displays the ORL-DLNN's structural representation.

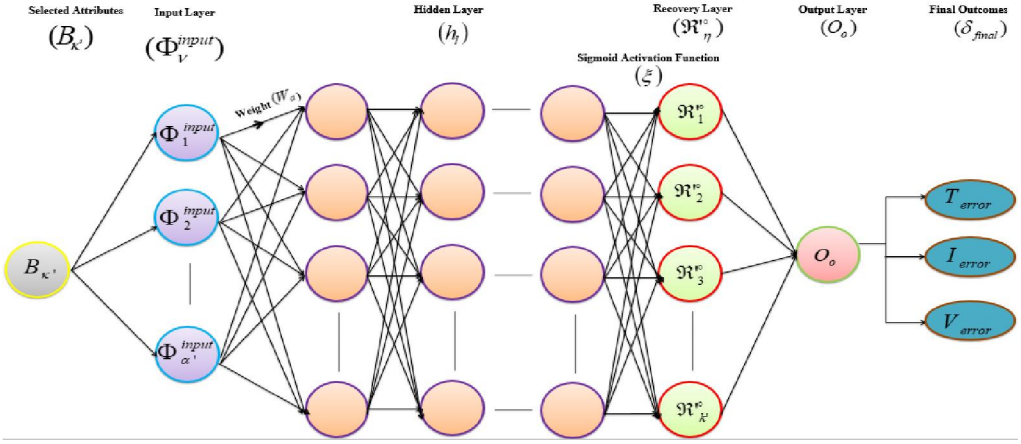


Fig 2: Structural representation of ORL-DLNN

Initially, the $B_{\kappa'}$ is provided as an input for the input layer $\left(\Phi_{\nu}^{input}\right)$, which is articulated as,

$$\Phi_{\nu}^{input} \rightarrow \left(\Phi_1^{input}, \Phi_2^{input}, \Phi_3^{input}, \ldots\ldots, \Phi_{\alpha'}^{input}\right) \quad where \quad \nu = (1,2,\ldots,\alpha') \tag{16}$$

Then, the input is provided to multiple hidden layers. The hidden layer processes the input by sharing their weight $\left(W_a\right)$ and bias value $\left(b'_a\right)$. Here, by utilizing DF-GRO, the weights are initialized, thus avoiding the exploding problems. The hidden layer $\left(h_l\right)$ operation is expressed as,

$$h_l \rightarrow \xi * \left\| W_a \cdot B_{\kappa'} \right\| + b'_a \quad where \quad l = (1,2,\ldots,\beta') \tag{17}$$

Here, the sigmoid activation function is notated as ξ, which is articulated as,

$$\xi = \frac{1}{1 + e^{-B_{\kappa'}}} \tag{18}$$

Thereafter, from the hidden layer, the recovery layer $\left(\Re'^{\circ}_{\eta}\right)$ filters the possible values, which is equated by,

$$\Re'^{\circ}_{\eta} \rightarrow \left[\Re'^{\circ}_1, \Re'^{\circ}_2, \Re'^{\circ}_3, \ldots\ldots, \Re'^{\circ}_{k'}\right] \quad Here \quad \eta = (1,2,\ldots,k') \tag{19}$$

Afterward, the recovery layer result is provided to the output layer $\left(\mathrm{O}_o\right)$, which is represented as,

$$\mathrm{O}_o = \sum \xi * \left\| W_a \cdot B_{\kappa'} \right\| + b'_a \tag{20}$$

astly, the proposed ORL-DLNN classified the formats with errors, such as text with error $\left(T_{error}\right)$, image with error $\left(I_{error}\right)$, and video with error $\left(V_{error}\right)$, which is articulated as,

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 3, April 2024

$$\delta_{final} \rightarrow \{T_{error}, I_{error}, V_{error}\} \tag{21}$$

The pseudocode for ORL-DLNN is provided below,

Pseudocode for ORL-DLNN

Input: Selected attributes
Output: Classification outcomes

Begin

Initialize, $B_{\kappa'}$, Φ_{v}^{input}, W_a and b'_a

Set $(iter = 1)$

While $[iter \leq \max(iter)]$

For each $B_{\kappa'}$

Estimate Φ_{v}^{input}

Initialize weights W_a

Perform $h_l \rightarrow \xi * \left\| W_a \cdot B_{\kappa'} \right\| + b'_a$

Compute $\xi = \dfrac{1}{1 + e^{-B_{\kappa'}}}$

Discover $\Re'^{\circ}_{\eta} \rightarrow \left[\Re'^{\circ}_{1}, \Re'^{\circ}_{2}, \Re'^{\circ}_{3}, \ldots\ldots, \Re'^{\circ}_{k'}\right]$

Estimate $O_o = \sum \xi * \left\| W_a \cdot B_{\kappa'} \right\| + b'_a$

End For

EndWhile

Obtain $\delta_{final} \rightarrow \{T_{error}, I_{error}, V_{error}\}$

End

Then, the detected errors are corrected, which is described below.

3.1.7 Error correction

Next, by utilizing the RR, the detected errors are corrected. Some error corrective actions, namely timing adjustments, amount mismatches, and removing duplicates are performed by the RR. Afterward, the records' alignments are ensured. Hence, the total c number of error-corrected data (T_x, I_y, V_z) is described as,

$$(T_x, I_y, V_z) \tag{22}$$

After that, the error-corrected texts are stored in the primary drive; also, the error-corrected images and videos are stored in the external drive.

3.1.8 Storage By

By utilizing the IFIM-MST, the (T_x, I_y, V_z) are stored in the drive. Matrix-based Minimum Spanning Tree (MST) has the edge-based approach; therefore, it covers all the points. However, more time is taken by MST; hence, an issue arises in edge loop checking. Fisher Information Matrix (FIM) is used to avoid this issue. Furthermore, the storage is handled with an index. Hence, the indexed FIM (U) is expressed as,

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 3, April 2024

$$U(T_x) = I'[0] * \left(-E\left[\frac{\partial^2 \ln p(T_x)}{\partial \theta^2} \right] \right)$$

Here, the indexed value is exemplified as $I'[0]$. The vertices $(\aleph_g)$ and edges (ζ_σ) in IFIM-MST are articulated as,

$$\aleph_g \xrightarrow{vertices} (\aleph_1, \aleph_2, \aleph_3, \ldots\ldots, \aleph_i) \quad Here \quad g = 1,2,3,\ldots,i \tag{23}$$

$$\zeta_\sigma \xrightarrow{edges} (\zeta_1, \zeta_2, \zeta_3, \ldots\ldots, \zeta_j) \quad Here \quad \sigma = 1,2,3,\ldots,j \tag{24}$$

After that, the edge weights (ω) are computed. Subsequently, for the T_x, the internal difference (d) is calculated, which is equated as,

$$d(T_x) = U * (\max(\omega(\zeta_\sigma))) \tag{25}$$

Then, the difference between the regions (J) is discovered. Afterward, grounded on the (d), (J) and T_x, the is stored, which is determined by,

$$J \le Min(d(T_x) + M(T_x)) \tag{26}$$

Here, the threshold function is symbolized as M. Likewise, by utilizing IFIM-MST, the I_y and V_z are stored. Two conditions are checked while storing. Whether the product is already present or not is checked by the first condition. If yes, the next condition (i.e., whether the details are the same or not) will be checked. The data will be stored if the first condition is no. The data will not be updated further if the second condition is yes; or else, the data is stored.

3.2 Testing

The customer id is entered by the information stewards into the classifier. Thereafter, the ORL-DLNN classifies the format with an error, which is described in section 3.1.6. A correction notification will be sent to the stewards if correction is present. Then, the information is corrected by the stewards. Next, by utilizing IFIM-MST, the details are stored concerning section 3.1.8.

IV. RESULTS AND DISCUSSIONS

Here, the proposed work's performance is assessed by implementing it in the working platform of PYTHON.

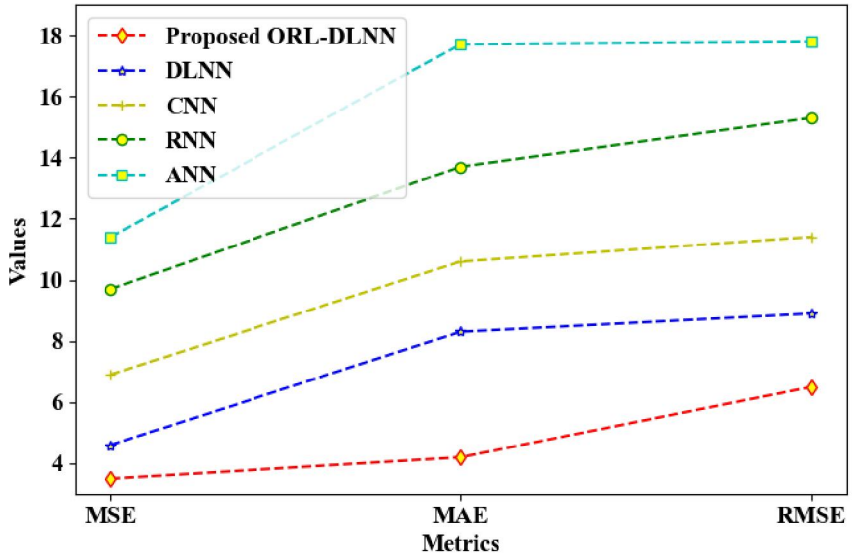
4.1. Database description

From Microsoft Azure, which is a public cloud computing platform, the databases are collected. Microsoft Azure gives access to cloud services and resources. Here, the dataset is gathered from different applications, namely Dynamics 365, Custom apps, Cloud apps, and Legacy apps.

4.2. Performance analysis of the proposed models

The performance evaluation of the proposed and existing works like DLNN, Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Artificial Neural Network (ANN) is exemplified in Figures 3 (a) and (b). The performance metrics are Mean Absolute Error (MAE), accuracy, recall, sensitivity, Mean Square Error (MSE), Root MSE (RMSE), precision, f-measure, and specificity. The proposed model overcomes the dead neuron and exploding gradient issues. Therefore, it attains the MSE, MAE, RMSE, accuracy, precision, recall, f-measure, sensitivity, and specificity of 3.5, 4.2, 6.5, 98.3%, 97.3%, 97.8%, 97.5%, 97.8%, and 97.3%, respectively. Yet, the existing models attained significantly lower performances compared to the proposed model. Therefore, the proposed technique's robustness is proven

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 3, April 2024



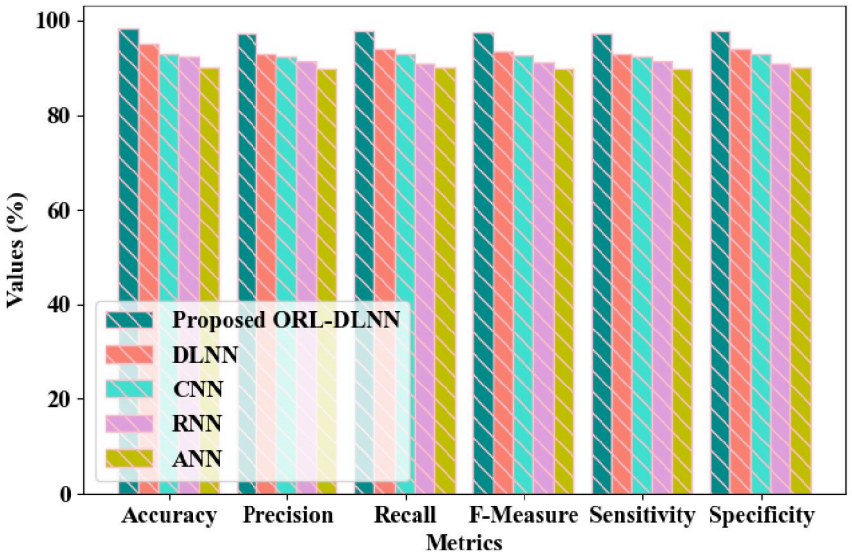


Fig 3 (a) and (b): Performance assessment of proposed ORL-DLNN

Table 1: Comparison of FPR and FNR

Techniques	FPR	FNR
Proposed ORL-DLNN	0.0463	0.0313
DLNN	0.0621	0.0498
CNN	0.0973	0.0578
RNN	0.1213	0.0893
ANN	0.2354	0.0997

The False Positive Rate (FPR) and False Negative Rate (FNR) analysis of the proposed and existing works is elucidated in Table 1. Here, the proposed ORL-DLNNachieved the FPR and FNR of0.0463 and 0.0313, respectively.The FPR and FNR of the existing DLNN are 0.0621 and 0.0498, respectively. Likewise, other existing models also had considerably higherFPR and FNR in contrast to the proposed model. Hence, the proposed ORL-DLNN's significance is estimated.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 3, April 2024

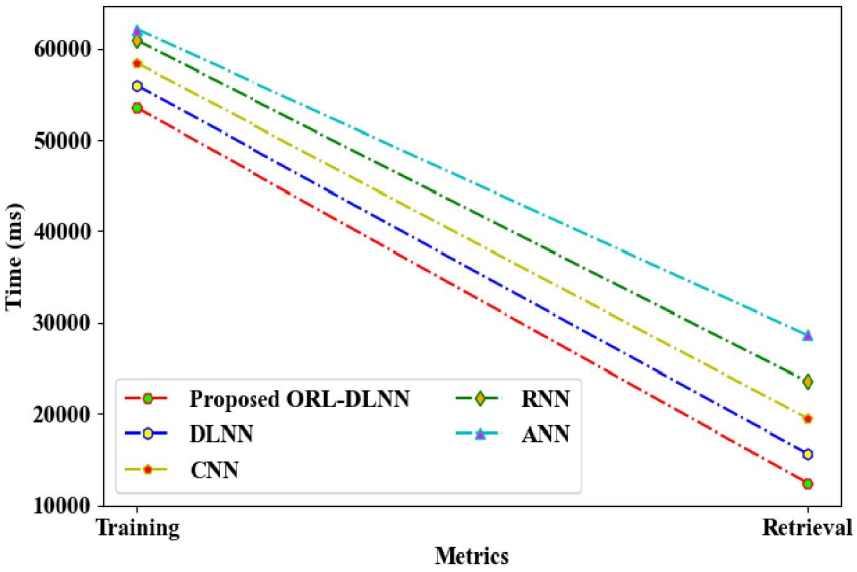


Fig 4: Training and Retrieval Time Analysis

Figure 4 displays the training and retrieval times of the proposed and conventional techniques. The prevailing CNN and RNN had a training time of 58422ms and 60912ms, and retrieval time of 19536ms and 23562ms, respectively. Yet, the proposed model takes a minimum of 53531ms and 12453ms for training and retrieval, respectively. This minimum time is owing to the integration of the modified DF-GRO with the proposed ORL-DLNN

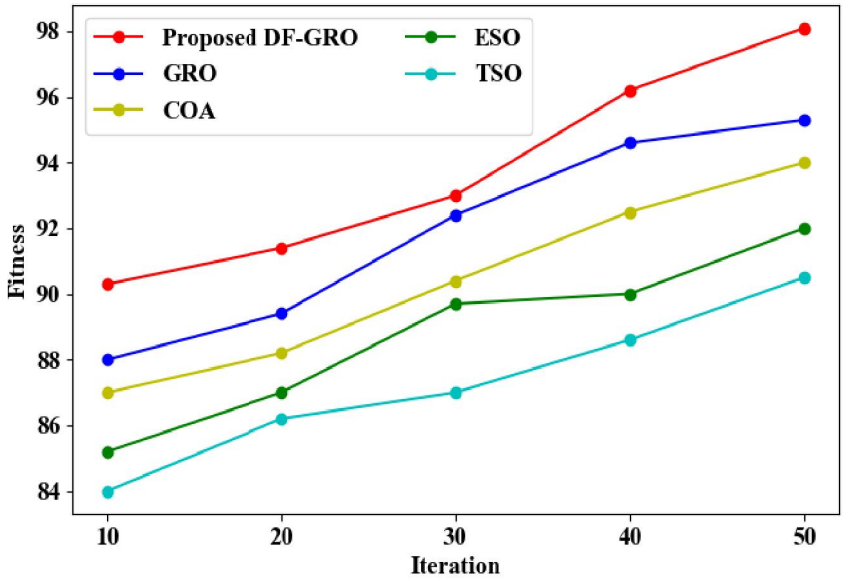


Fig 5: Fitness vs. Iteration Analysis

Figure 5 evaluates the fitness vs. iteration of the proposed DF-GRO and the existing GRO, Coati Optimization Algorithm (COA), Egret Swarm Optimization (ESO), and Tuna Swarm Optimization (TSO). For 10, 20, 30, 40, and 50 data, the existing TSO model attained 84%, 86.2%, 87%, 88.6%, and 90.5%, respectively. However, the proposed model had 90.3%, 91.4%, 93%, 96.2%, and 98.1% fitness for 10, 20, 30, 40, and 50 data, respectively. Here, when contrasted with the existing models, the fitness of the proposed model is better. Therefore, the significance of the proposed DF-GRO is well-defined by the convergence improvement.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 3, April 2024

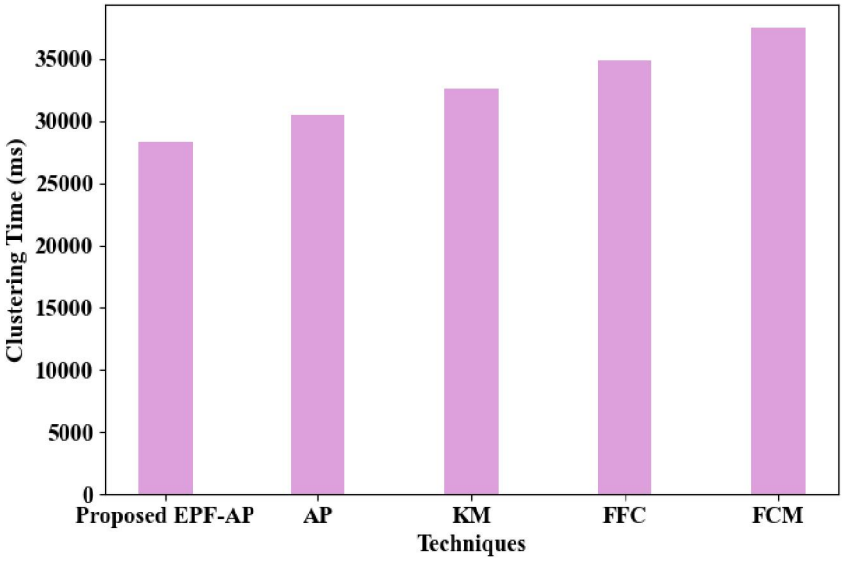


Fig 6: Clustering time analysis

The clustering time analysis of the proposed EPF-AP and the existing AP, K-Means (KM), Farthest First Clustering (FFC), and Fuzzy C-Means (FCM) is depicted in Figure 6. For clustering, the conventional AP had taken 30452ms. Therefore, for effectively supporting large-scale data, the exponential probabilities function is employed with AP. Hence, to cluster the products, the proposed EPF-AP takes a minimum of 28324ms, while other existing models had higher clustering times. Thus, the robustness of the proposed model is estimated

Table 2: Tree construction time analysis

Techniques	TCT (ms)
Proposed IFIM-MST	5478
MST	8341
BT	10432
ST	13421
RBT	15321

Table 2 illustrates the TCT analysis of the proposed IFIM-MST and the existing MST, Binary Tree (BT), Segment Tree (ST), and Red Black Tree (RBT). For the tree construction process, the existing MST, BT, ST, and RBT take 8341ms, 10432ms, 13421ms, and 15321ms. Nevertheless, the proposed IFIM-MST takes 5478ms for the tree construction. This minimum time is attained by avoiding the edge loop checking process, which takes more time. Hence, the analysis proves the proposed IFIM-MST's superiority.

V. CONCLUSION

In this framework, an efficient MMDM system grounded on ORL-DLNN and IFIM-MST is proposed. The proposed ORL-DLNN's assessment stated that this model efficiently classified the format and detected errors with an accuracy of 98.3% and a precision of 97.3%. Moreover, the proposed model also obtained the MSE, MAE, and RMSE rates of 3.5, 4.2, and 6.5, correspondingly. A minimum time of 5478ms is taken by the proposed IFIM-MST for tree construction. Thus, the proposed technique's robustness is proved

REFERENCES

[1]. Akter, S., Motamarri, S., Hani, U., Shams, R., Fernando, M., MohiuddinBabu, M., &NingShen, K. (2020). Building dynamic service analytics capabilities for the digital marketplace.Journal of Business Research, 118, 177–188.https://doi.org/10.1016/j.jbusres.2020.06.016

[2]. Behera, T. K., & Panda, B. S. (2023). Master Data Management using Machine Learning Techniques: MDM Bot. TechRxiv, 1–15. https://doi.org/10.36227/techrxiv.21818040

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.53
Volume 4, Issue 3, April 2024

[3]. Hannila, H., Silvola, R., Harkonen, J., &Haapasalo, H. (2022). Data-driven Begins with DATA; Potential of Data Assets. Journal of Computer Information Systems, 62(1), 29–38. https://doi.org/10.1080/08874417.2019.1683782

[4]. Jaskó, S., Skrop, A., Holczinger, T., Chován, T., &Abonyi, J. (2020). Development of manufacturing execution systems in accordance with Industry 4.0 requirements: A review of standard- and ontology-based methodologies and tools. Computers in Industry, 123, 1–18. https://doi.org/10.1016/j.compind.2020.103300

[5]. Kuznetsov, S., Tsyryulnikov, A., Kamensky, V., Trachuk, R., Mikhailov, M., Murskiy, S., Koznov, D., &Chernishev, G. (2022).Unidata-A Modern Master Data Management Platform.In EDBT/ICDT Workshops.2022., 1–13.http://ceur-ws.org

[6]. Miklosik, A., & Evans, N. (2020). Impact of Big Data and Machine Learning on Digital Transformation in Marketing: A Literature Review. IEEE Access, 8, 101284–101292. https://doi.org/10.1109/ACCESS.2020.2998754

[7]. Mositsa, R. J., Van der Poll, J. A., &Dongmo, C. (2023).Towards a Conceptual Framework for Data Management in Business Intelligence.Information (Switzerland), 14(10), 1–26. https://doi.org/10.3390/info14100547

[8]. Pansara, R. R. (2020a). Graph Databases and Master Data Management : Optimizing Relationships and Connectivity. International Numeric Journal of Machine Learning and Robots, 1(1), 1–10.

[9]. Pansara, R. R. (2020b). NoSQL Databases and Master Data Management : Revolutionizing Data Storage and Retrieval. International Numeric Journal of Machine Learning and Robots, 4(4), 1–11.

[10]. Sawadogo, P., &Darmont, J. (2021).On data lake architectures and metadata management.Journal of Intelligent Information Systems, 56(1), 97–120.https://doi.org/10.1007/s10844-020-00608-7

[11]. Song, H., Gi, I., Ryu, J., Kwon, Y., &Jeong, J. (2023). Production Planning Forecasting System Based on M5P Algorithms and Master Data in Manufacturing Processes. Applied Sciences (Switzerland), 13(13), 1–24. https://doi.org/10.3390/app13137829

[12]. Warnke-Sommer, J. D., &Damann, F. E. (2021). An improved machine learning application for the integration of record systems for missing US service members. International Journal of Data Science and Analytics, 11(1), 57–68. https://doi.org/10.1007/s41060-020-00236-y

[13]. Xiaojing, L., &Minghai, L. (2021).Application and Research of the Intelligent Management System Based on Internet of Things Technology in the Era of Big Data.Mobile Information Systems, 2021, 1–6. https://doi.org/10.1155/2021/6515792

[14]. Yaqoob, I., Salah, K., Jayaraman, R., & Al-Hammadi, Y. (2022).Blockchain for healthcare data management: opportunities, challenges, and future recommendations. Neural Computing and Applications, 34(14), 11475–11490. https://doi.org/10.1007/s00521-020-05519-w

[15]. Ying, S., Sindakis, S., Aggarwal, S., Chen, C., & Su, J. (2021). Managing big data in the retail industry of Singapore: Examining the impact on customer satisfaction and organizational performance. In European Management Journal (Vol. 39, Issue 3). Elsevier Ltd. https://doi.org/10.1016/j.emj.2020.04.001