Computers and Electrical Engineering 92 (2021) 107132

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

Performance analysis of the dynamic trust model algorithm using the fuzzy inference system for access control☆

G Abirami*, Revathi Venkataraman

Department of Computer Science and Engineering, SRMIST, Kattankulathur, India

ARTICLE INFO

Keywords:
Trust model
Access control
Attribute Based Access Control
Performance
Direct Observation
Expected Trust
Feedback
Dynamic Trust Model Algorithm

ABSTRACT

Accessing company resources such as personal data, financial data, and company networks in the dynamic business environment is a vital task. This paper proposes a Dynamic Trust Model Algorithm (DTMA) using fuzzy inference rules for access control. The novelty is finding the unsteady behaviour of an employee in a varying period using trust mathematical computation. Based on four parameters such as Performance (P), Direct Observation (DO), Expected Trust (ET) and Feedback (F), the trust value is calculated. To manage the deliberate altering behaviour of the hostile employees, a dynamic Trust Value (TV) has been calculated and restrict their harmful actions. Also, the performance and accuracy of the DTMA have been assessed and compared to other models such as DLATrust, DyTrust and SecTrust. The proposed DTMA gives better results in terms of accuracy, precision, recall, F-Score and Receiver Operating Characteristics (ROC).

1. Introduction

To provide access control for an organisation's resources in a dynamic environment is a challenging task due to changes in human behaviour and trustworthiness. Hence, trust is an important aspect to be considered in a business environment. The current enterprise information system scenarios have various users such as organizational personnel, customers, and visitors. The organizational personnel interact directly with the IT and other enterprises; hence, there is a maximum vulnerability of misusing the resources. Customers are important users and the basic goal of the enterprise is to satisfy customer needs. Though less access to resources is possible to a customer, an untrusted customer may cause harm to the intangible assets of an enterprise, by showing unsatisfied feedback in online [1,2]. Hence, trust is one of the security concepts in pervasive computing. Though many dynamic trust computation models have been designed, still, there is a demand for accomplishing trust management in industries and business contexts. So, this paper proposes a novel Dynamic Trust Model Algorithm (DTMA) to address the dynamicity problems of trust to access the resources.

The trust model has been constructed by using various mathematical approaches and parameters. Also, the proposed DTMA has measuring parameters, such as Performance (P), Direct Observation (DO), Expected Trust (ET), and Feedback (F). The Performance (P) tells the knowledge, skillset, and effectiveness of a person. 'DO' is monitoring the behaviour activities, effectiveness in work and skillset of a person. The 'ET' is calculated between Recent trust (R_t) and Historic trust (H_t). Finally, Feedback (F) has been calculated by two additional parameters, such as Recommendation (*Rec*) and Reputation (*Rep*). The number of recommenders' 'n' provides recommendations about a person and the recommendations are classified based on their trust weightage '*w*'. Preferences to

☆ This paper is for special section VSI-ain. Reviews processed and recommended for publication by Guest Editor Dr. S. Smys.

* Corresponding author.

E-mail addresses: abiramig@srmist.edu.in (G. Abirami), revathin@srmist.edu.in (R. Venkataraman).

https://doi.org/10.1016/j.compeleceng.2021.107132
Received 22 April 2020; Received in revised form 22 September 2020; Accepted 15 March 2021
Available online 20 April 2021

recommenders are based on the trust weightage of a user. The *Rep* is computed based on Triple Exponential Smoothing (TES) which has trend and seasonal value. Here, the trend is considered as the current context and seasonal value is taken as a periodicity or time interval of trust calculation. Hence, TES predict the unsteady behaviour of a person. Whereas in the existing model the Bounded Double Exponential Smoothing is used to calculate the *Rep* which has shown predicted value with only a certain period. After, using the Mamdani fuzzy inference system, 30 rules have been proposed to calculate the dynamic Trust Value (TV) of a person. The computed TV is also considered as one of the attributes in ABAC to allow or deny a person to access the resources. Hence, the proposed method gives the dynamic TV, according to trust parameters, and Mamdani fuzzy inference rules. The dynamic TV predicts the hostile employee's behaviour to restrict the access privilege of resources. Also, trustworthy employees are allowed to access the resources in time critical situations according to TV to sustain and maximize the business productions and profits.

This paper is organized as follows: Section 2 summarizes the related work for the proposed model. Section 3 provides the proposed trust model. Trust mathematical calculations are computed in Section 4. The Dynamic Trust Model Algorithm (DTMA) is proposed in Section 5. Section 6 discusses the Mamdani rules on fuzzy inference systems. The Experimental setup is provided in Section 7. Results and Discussion are shown in section 8. Finally, Section 9 provides the conclusion and future work.

2. Related work

Currently, access control based on the trust model has been increased in ubiquitous computing. Das et al. [3] proposed a secured trust of dynamic computation model for finding the trust changing in the behaviour of a malicious agent and provided only in generic application not in specific application. Zhong et al. [4] proposed a dynamic trust model that differentiates integrity trust from competence trust for an unstable user. Assessing the changes of trustee behaviour, a confidence value is added to the trust score. Hoogendoorn et al. [5] validated a relative trust to predict human trust-based behaviour which shows better results than a benchmark model. This model is nonlinear in trust and experience, and difficult to analyse. Ghavipour et al. [6] conducted experiments on a real trust network dataset to infer the trustworthiness of an unknown user using Distributed Learning Automata (DLA). Gupta et al. [7] provided a reputation based trust model to collect the feedback about the past behaviour of a person that helps to find whom to trust and whom to distrust. The system is for static and not for dynamic changes of user behaviour. Rajganesh et al. [8] proposed a fuzzy based intelligent cloud computing service to evaluate customer feedback for providing privacy to data, but a context metric to evaluate trust is not shown. Shreya Shashi and Kakali Chatterjee. [9] proposed a framework which identified the set of customer's trusts and the weight of each customer has been calculated in E-commerce. But they failed to show it in varying scenarios.

Braga et al. [10] provided a thorough review of computation trust and reputational model for human to computer interaction. Rashi Srivastava et al. [11] proposed a framework to provide a cloud service analysis with various sets of parameters, such as agility, finance, usability, security, and system performance using fuzzy logic. Kesarwani Abhishek and Pabitra Mohan Khilar [12] proposed a trust based access control model for cloud service providers using fuzzy logic. Ma Shunan et al. [13] proposed a dynamic access control model based on scenario trust for factors, such as access time, place, history behaviour and risk control strategy by applying fuzzy logic. Jayasinghe et al. [14] provided an algorithm based on machine learning to classify the trust features for decision making. Servos et al. [15] suggested current research challenges in ABAC. Boukerche et al. [16] explained a reputation based trust model system to track the behaviour of a network model. Abirami and RevathiVenkataraman [17,18] implemented ABAC -T in XACML for providing access control to achieve fine grained policy and provided a recommendation trust model. Jain et al. [19] explained a trust model with quality of service parameters using a fuzzy logic system; they have taken only a few parameters such as availability, reliability and turnaround time. However, security and time allocations are lacking. Hu et al. [20] discussed the ABAC principle and guidelines for a business environment but have not discussed the various trust metrics. Chen et al. [21] proposed a fuzzy inference trust model for a Peer to Peer (p2p) network environment and discussed the Mamdani type fuzzy inference technique in peer trust. Samari et al. [22,23] provided a collaboration graphical tool for ABAC policies for dynamic trust and privacy evaluation. Zhao et al. [24] provided a method for the assessment of recommendation trust and have given a solution to the problem of weight assignment to decision attributes by using relevant theories in expert investigation, fuzzy analysis and gray correlation analysis. But this is lacking in shows the dynamic problem. This dynamicity is overcome in the proposed model. Singh et al. [25] proposed trust assessment of medical data, though, not discussed about the crisis situation in access control. Hence, by exploring the above literature work, it is found that there is a lack in the vibrant users' behaviour. Also, it is a less consistent mathematical approach. The proposed system shows a general mathematical approach for trust model in a dynamic environment.

3. Proposed trust model

The major trust model is classified into direct trust and indirect trust [18]. Direct trust is observing the person's or entity's benevolence through direct contact and interactions. Indirect trust is getting information about a person through another person or entity. Recommendation and Reputation are two important methods of finding the trust of a person. Recommendation is provided by a third person who is known by both the trustor and the trustee. Reputation is a group of community provided trustworthiness about a person (trustee). A lot of research work has been done on trust calculation [2] in various areas and both static and dynamic trust models discussed. In the proposed trust model, the dynamic trust model has been shown using the mathematical approach to provide accuracy in the trust value which is assigned to users.

If an employee wants to access the company resources, such as financial, personal detail, and performance report of peer employee, then it has been permitted or denied according to the computed TV of an employee, as shown in Fig. 1. The parameters P, DO, ET and F are used to calculate the TV. 'p' is calculated based on knowledge, skillset and effectiveness of a person. 'DO' is the difference of the present and last satisfactory interactions. 'ET' is computed using recent trust and historic trust. 'F' is calculated by recommendation and reputation. By computing all four parameters TV has been assigned by linguistic variables, such as Ignored, Not_Trusted, Partially_Trused, Trusted and Fully_Trusted. According to the possession of these variables, the employees in a company can access the resources. Then the computed TV is included as one of the attributes in Attribute Based Access Control (ABAC) to achieve a fine grained mechanism for accessing the resources. The proposed system has been implemented on the Kaggle dataset at different time intervals and the performance of the DTMA is evaluated. The Kaggle dataset of 1000 records and 15 attributes each is run at different time periods 't' by changing the attribute values. Then the proposed DTMA is compared with the DLATrust, DYTrust, and SecTrust to show better results in terms of accuracy, precision, recall F-Score and ROC for the Trust Value (TV) by using the J48 decision tree algorithm which is run on the WEKA tool.

4. Trust mathematical calculation

To provide a dynamic trust computation with accurate TV, the mathematical model of each input parameter such as Performance (P), Direct Observation (DO), Feedback (F), and Expected Trust (ET) has been calculated with additional attributes. These redefined parameters are used to calculate the TV of a person.

4.1. Performance (P)

According to the unifying theory of human performance, it measures the ability of the Trustees (T_e) based on their Knowledge (K), Skillset (S_t) and Effectiveness (E_{ff}) in the workplace. These three attributes must be presented to achieve the expected target, where τ_i is the sum of the tasks in a given time period (CP_t). The P in Eq. (1) of the trustee can be calculated in different time periods. Then the value of P should be normalized in the range of 0 to 1. S_t in Eq. (2) is the ability to carry out a task with determination.

$$P = K + S_t + E_{ff} \tag{1}$$

Where,

$$S_t = \frac{\sum_{i=1}^{n} \tau_i}{CP_t} \tag{2}$$

4.2. Direct observation (DO)

'DO' is computed by satisfactory experience on T_e in a usual context. It is a method of satisfaction of Trustor (T_r) on collecting the

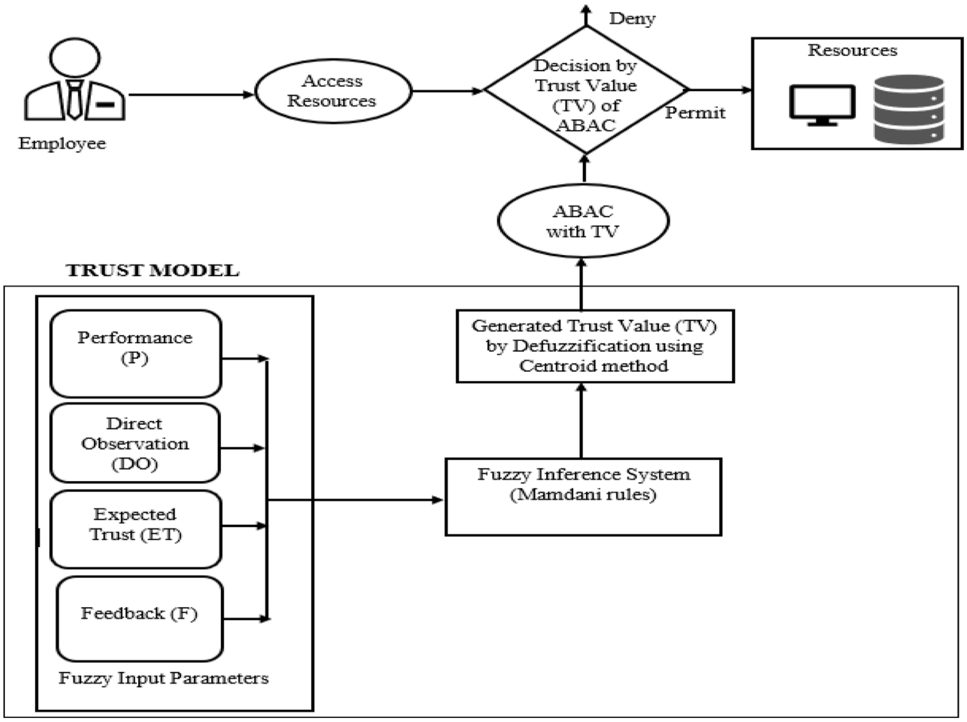


Fig. 1. Proposed dynamic trust model algorithm (DTMA).

ongoing behaviour of a T_e. The degree of satisfaction is the overall activities of T_e's attitude towards a work in a usual context and emotional reaction in a different situation. Let *Saf*(T_r,T_e) in Eq. (3) represent the amount of satisfaction T_r has upon T_e based on 'n' number of interactions in the ith time interval [3]. If the interactions are fully satisfied then *Saf*(T_r,T_e) is 1 or otherwise 0 which is represented in Eq. (4)

$$Saf(T_r, T_e) = \alpha \times Saf_{cur} + (1-\alpha) \times Saf^{t}_{n-1}\ (T_r, T_e) \tag{3}$$

Initially *Saf*(T_r,T_e) = 0; , it has been updated, after the ith time interval.

$$Saf(T_r, T_e) = \{\ \begin{array}{c} 0,\ \textit{if interaction is fully unsatisfactory} \\ 1,\ \textit{if interaction if fully satisfactory} \\ \epsilon(0,1),\ \textit{otherwise} \end{array} \tag{4}$$

$$\boldsymbol{\xi_n}\ (T_r, T_e) = c \times \delta_n\ (T_r, T_e) + (1-c) \times \boldsymbol{\xi_{n-1}}\ (T_r, T_e) \tag{5}$$

$$\boldsymbol{\xi_{last}}\ (T_r, T_e)\ and\ \xi_0(T_r, T_e) = 0 \tag{6}$$

The weight α change based on the accumulated deviation ξ_n(T_r,T_e)

$$\boldsymbol{\alpha} = Thr + c \times \frac{\delta_n(T_r, T_e)}{1 + \xi_n(T_r, T_e)} \tag{7}$$

$$\boldsymbol{\delta_n}\ (T_r, T_e) = \left|Saf^{t}_{n-1}\ (T_r, T_e) - Saf_{cur}\right| \tag{8}$$

Saf$_{cur}$ is the present satisfactory interaction and Saf^{t}_{n-1} the last interaction with T_e. ''*c*' is the user defined constant which interprets the deviation in the interactions. if '*c*' increases, the recent deviation δ_n (T_r,T_e) in Eq. (8) is more to be considered than the accumulated deviation ξ_n(T_r,T_e) in Eq.(5) and vise versa. The initial value and lasr values of accumulated deviation ξ_n(T_r,T_e) are set to 0 in Eq. (6). The threshold *Thr* is used to make the α dynamic in Eq. (7). By empirically tuned, *Thr* is set to 0.25 and α is set to 1.

4.2.1. Decay model

The TV is reduced due to the absence of interaction. If there is no interaction between T_r and T_e for a long time, then the TV is remaining idle without any update. Since it depends on DO the satisfaction metrics are to be considered.

$$\widehat{Saf}(T_r, T_e) = Saf^{t}_{n}\ (T_r, T_e)e^{-\lambda\Delta t} \tag{9}$$

$$\Delta t = t_{current} - t_{previous} \tag{10}$$

where $\widehat{Saf}(T_r, T_e)$ in Eq. (9) represents the value of satisfaction after decay. Here, λ is the decay constant and Δ*t* in Eq.(10) denotes the interval between the current interaction and the last interaction. So, if successive interactions increase due to a long time interval, the Δ*t* also increases. Hence, reliable TV is based on recent interactions.

4.3. Expected trust (ET)

ET is acquired from the computation of both Historic trust (H_t) and Recent trust (R_t).

4.3.1. Recent trust (R_t)

The Recent trust (R_t) in Eq. (11) is a weighted combination of direct trust and indirect trust. The weightage of direct trust is more because the trustor (T_r) has direct interaction with trustee (T_e). The trust is more robust on the own experience of T_e than on the recommendation from others [3].

$$R_{t\ =}\ \ \beta \times dt + (1-\beta) \times Id_t \tag{11}$$

Where, β represents the weight of direct trust that can be calculated by

$$\beta = \frac{I^t}{I^t + M^t} \tag{12}$$

$$M^t(T_r\ ,\ T_e) = \frac{\sum_{x\in W-\{T_r\}} FeCr\left(T_{r,}\ x\right) \times I^t(x,\ T_e)}{|W - \{T_r\}|} \tag{13}$$

Here, I^t represents the number of interactions, that 'T_r' has performed with 'T_e' in the ith interval. Hence, M^tdenotes the mean number of interaction that the other T_r entity has conducted with T_e. To compute M^t, in Eq. (13) the Feedback Credibility (*FeCr*) of the recommender and number of interaction count I^t has been considered. If it increases when compared to M^t, β in Eq. (12) also increases parallelly. '*W*' represents, that the number of the recommends have not interacted with T_e.

If $|W - \{Tr\}|$ = 0, then, set M^t = 0 and if $I^t + M^t$ = 0, then set β = 0.5 (default value).

4.3.2. Historic trust (H_t)

According to long term behavioural patterns and past experiences, the H_t is calculated. The H_t is given by using the exponential average function to reduce the storage overhead. Let H_t in Eq. (14) represents the historical trust that T_r has about T_e.

$$H_t\ (T_r, T_e) = \frac{\rho \times H_{t_{n-1}}(T_r, T_e) + R_{t_{n-1}}\ (T_r, T_e)}{2} \tag{14}$$

Here ρ $(0 \leq\ \ \rho \leq 1)$ is the forgetting factor (omitting older experience) and $H_t$of ($(T_{r,}\ T_e) = 0$. Also, good behaviour of T_e in the recent interaction has not been considered, when they had malicious activity in the past. Hence, T_e is to be considered as good only when they have a greater number of interactions with T_r.

The Expected Trust (ET) is calculated by the following Eq. (15)

$$E(T_r, T_e) = \begin{cases} 0,\ \textit{if neither } R_t \textit{ nor } H_t \\ \eta\ R_t(T_r, T_e) + (1-\eta)H_t(T_r,\ T_e) \\ \textit{if either } R_t \textit{ and / or } H_t \textit{ is available} \end{cases} \tag{15}$$

Initially, η is set to 0.5, but η in Eq. (16) is adjusted dynamically based on the difference of R_t and H_t (deviation factor is ξ set to 0.3, which is empirically tuned), The η value increases by 0.1(empirically tuned) when R_t is beyond H_t by Eq. (14) which means that ET is increasing based on R_t then H_t and vice versa. By adjusting the value of ξ, the recent value has been found. Hence ET is the sum of the recent R_t and H_t.

$$\eta = \begin{cases} \eta + 0.1\ \textit{if}\ R_t(T_r, T_e) - H_t(T_r,\ T_e) > \xi, \\ \eta - 0.1\ \textit{if}\ R_t(T_r, T_e) - H_t(T_r,\ T_e) < -\xi, \\ \eta\ \textit{if} -\xi < R_t(T_r, T_e) - H_t(T_r,\ T_e) < \xi, \end{cases} \tag{16}$$

4.4. Feedback (F)

'F' is calculated based on two metrics, such as Recommendation (*Rec*) and Reputation (*Rep*).

$$F = Rec + Rep \tag{17}$$

4.4.1. Recommendation (Rec)

The *Rec* of T_e is provided by 'n' number of trustworthy recommenders. The weight index of the recommenders varies according to the trustworthiness. W_i is the weight index of the *i*th recommender in Eq.(19). *Rec* [17,18] is computed by the simple exponential average of 'n' recommenders' value in Eq. (18).

$$Rec(T_e) = w_1 ptr(T_e) r_1 + w_2 ptr(T_e) r_2 + \ldots\ldots\ldots. + w_n ptr(T_e) r_n \tag{18}$$

$ptr(T_e)r_i = i^{th}$ received recommendation

$$\text{Here } w_1 + w_2 + w_3 + w_4 + \ldots\ldots\ldots. + w_n = 1 \tag{19}$$

The Eq.(20) shows error index $\boldsymbol{Error_i}$ is calculated on the observation value and recommendation value

$$Error_i = \left|ptr(T_e\ )_{obs} - ptr(T_e\ )_{r_i}\right| \tag{20}$$

The initial value of each weighing index should be the same, i.e. $1/n$. Those values which are more accurate have more weighing indices.

4.4.2. Reputation (Rep)

The *Rep* of T_e is calculated according to the rating sequence collected in different time series and different contexts. To predict the next rating based on a previous rating sequence [4] Double Bounded Exponential Smoothing (BDES) is able to catch only the trend behaviour and is not useful for unpredictable T_e such as random behaviour pattern. In this proposed DTMA system, Triple Exponential Smoothing (TES) is provided to predict the random behaviours of a person. In Eq. (22) the Observation of T_e is provided by the difference between the current observation and the previous context rating, given by trustworthy recommenders. ϕ_t is smoothed observation, ψ_t is the trend factor, and ν_t the seasonal or context values. These factors are calculated by Eq. (25) and the description is given in the Table 1. By means, it gives the random behaviour of a person and improves the accuracy of the predicted TV.

$$\varphi_t = \alpha\left(\omega_t - \nu_{t-p}\right) + (1-\alpha)(\varphi_{t-1} +\ \psi_{t-1}) \tag{21}$$

$$\psi_t = \beta(\varphi_t - \varphi_{t-1}) + (1-\beta)\psi_{t-1} \tag{22}$$

$$\nu_t = \gamma(\omega_t - \varphi_t) + (1-\gamma)\nu_{t-p} \tag{23}$$

The T_e random behaviours of different contexts or periodicity in a particular trend factor gives the TV and forecasts the predictable value with recent observations. There are three constant parameters in Eqs. (21)–(23) α, β, and γ respectively. These values range from 0 to 1, and the initial value is set by T_r based on the minimum mean square error (MSE) between the rating sequence and the predicted value. So, the values of $\alpha = 0.2$, $\beta = 0.1$, and $\gamma = 0.25$ are chosen in such a way that the MSE of the error is minimized [3]. F_{t+m}in Eq. (24) is the forecast values obtained by computing all the attributes, such as ϕ_t,ψ_t, and ν_t. Hence, *Rep* in Eq. (25) is a value of F_{t+m} in different time series to achieve the dynamic TV.

$$F_{t+m} = \varphi_t + m\psi_t + \nu_{t-p+1+((m-1)mod\ p)} \tag{24}$$

$$\text{Rep} = F_{t+m} \tag{25}$$

5. Proposed Dynamic Trust Model Algorithm (DTMA)

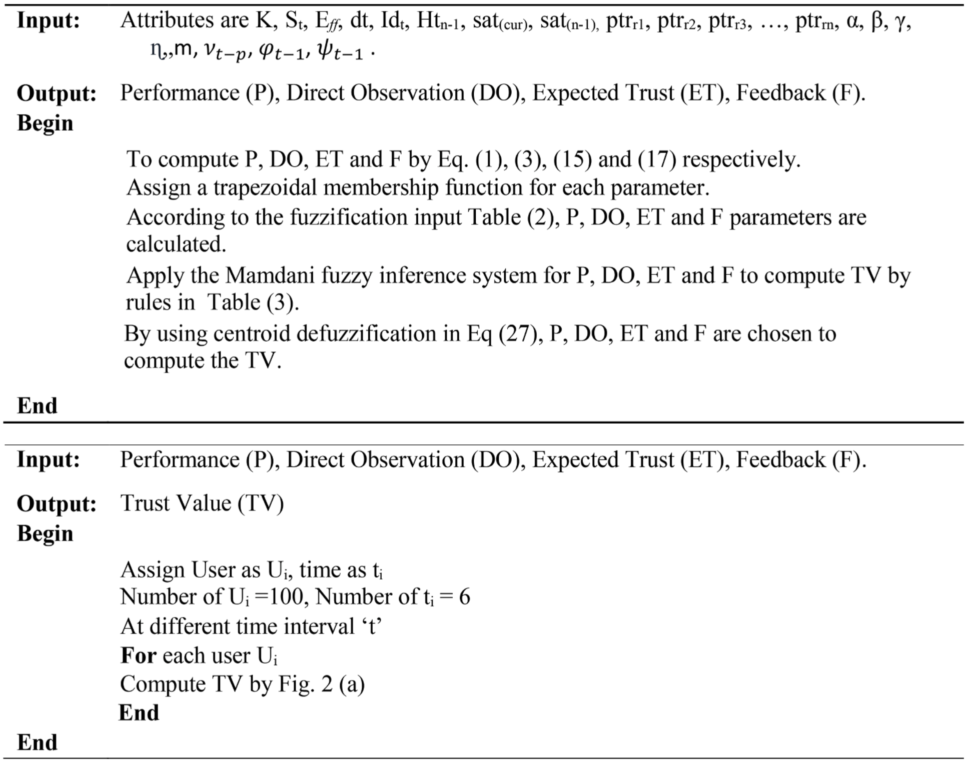
The algorithm for calculating the TV is shown in Fig. 2 (a) in which all the attributes for calculating the parameters such as P, DO, ET, and F are given as input. According to Eq. (1), (3), (15), and (17) the value of these parameters has been calculated. The trapezoidal membership function for each parameter is assigned and the membership values are range from 0 to 1 for fuzzification. The computed values of P, DO, ET, and F are given as input to calculate TV by applying Mamdani fuzzy inference rules. The value of TV is obtained by using centroid defuzzification.

The Algorithm for the proposed DTMA is given in Fig. 2 (b) which calculates P, DO, ET and F for a sample of 100 users with different time intervals ‘t’, and obtain TV based on Mamdani inference system. The TV is changing dynamically with respect to DTMA.

6. Fuzzy inference system

6.1. Input for fuzzification

The fuzzy logic toolbox is used to design and develop this model that contains the inference rules of the Mamdani inference system with trapezoidal membership functions (μ) for fuzzification in Eq. (26);, here, *R* is the real numbers ranging from a scaling factor of



```
Input:    Attributes are K, $S_t$, $E_{ff}$, dt, $Id_t$, $Ht_{n-1}$, $sat_{(cur)}$, $sat_{(n-1)}$, $ptr_{r1}$, $ptr_{r2}$, $ptr_{r3}$, …, $ptr_{rn}$, α, β, γ,
            $\eta_,$,m, $\nu_{t-p}$, $\varphi_{t-1}$, $\psi_{t-1}$ .
Output:   Performance (P), Direct Observation (DO), Expected Trust (ET), Feedback (F).
Begin
          To compute P, DO, ET and F by Eq. (1), (3), (15) and (17) respectively.
          Assign a trapezoidal membership function for each parameter.
          According to the fuzzification input Table (2), P, DO, ET and F parameters are
          calculated.
          Apply the Mamdani fuzzy inference system for P, DO, ET and F to compute TV by
          rules in  Table (3).
          By using centroid defuzzification in Eq (27), P, DO, ET and F are chosen to
          compute the TV.
End
```

```
Input:    Performance (P), Direct Observation (DO), Expected Trust (ET), Feedback (F).
Output:   Trust Value (TV)
Begin
          Assign User as $U_i$, time as $t_i$
          Number of $U_i$ =100, Number of $t_i$ = 6
          At different time interval ‘t’
          For each user $U_i$
          Compute TV by Fig. 2 (a)
          End
End
```

Fig. 2. (a) Proposed Algorithm for calculating Trust Value (TV). (b) The Algorithm for proposed DTMA.

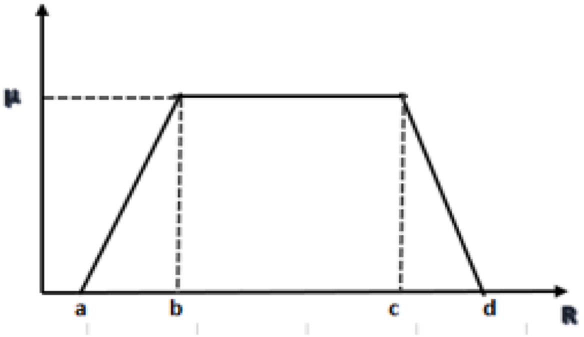


Fig. 3. Trapezoidal membership functon for fuzzification

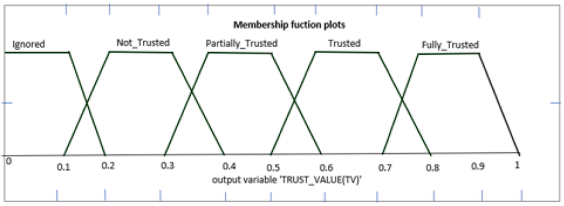


Fig. 4. Output Trust Value (TV) membership function.

Table 1
Additional Attributes in measuring parameters.

Attributes in Measuring parameters	
ϕ_t	*Smoothed observation*
ψ_t	*Trend*
ν_t	*Seasonal/context*
ω_t	*Current Observed rating*
ν_{t-p}	*Previous context rating by recommender*
ϕ_{t-1}	*Previous initial rating by recommender*
ψ_{t-1}	*Last context rating by recommender*
F_{t+m}	*Forecast or Estimated value at time* $t+m$, *where* $m > 0$
α	*Data factor,* $0 < \alpha < 1$
β	*Trend factor,*$0 \leq \beta < 1$
γ	*Seasonal/context factor,*$0 < \gamma < 1$
t	*The index that denote at time*
p	*Period*
m	*Set to 1*

Table 2
Fuzzification input parameters.

Parameters	**Fuzzification inputs**				
	0-0.2	**0.1 - 0.4**	**0.3-0.6**	**0.5-0.8**	**0.7-1**
Performance(P)	UnSkilled	Less_Skilled	Partially_Skilled	Skilled	Highly_Skilled
Direct Observation (DO)	Very_Low	Low	Average	High	Very_High
Expected Trust (ET)	Very_Low	Low	Average	High	Very_High
Feedback (F)	Poor	Average	Good	Very_Good	Excellent

0 to 1, and in Fig. 3 the trapezoidal shaped membership function $f(TV; \quad a, b, c, d, \mu)$ is used. The representation of linguistic variables such as Ignored, Not_Trusted, Partially_Trused, Trusted and Fully_Trusted of output TV membership function plot is shown in Fig. 4. Table 2 shows the crisp values of the fuzzification input, and centroid defuzzification has been used for obtaining a crisp output.

$$f(TV;a,b,c,d,\mu)=\begin{cases}0\ when\ TV<aandTV>d\\ \dfrac{(a-TV)\mu}{a-b}\ when\ a\leq TV\leq b\\ \mu\ when\ b\leq TV\leq c\\ \dfrac{(d-TV)\mu}{d-c}\ when\ c\leq TV\leq d\end{cases}\tag{26}$$

6.2. Fuzzification inference rules

The inference engine consisting of the knowledge base which contains 30 fuzzy inference rules using Mamdani's system, has been proposed, to evaluate the TV and is given in Table 3.

6.2. Defuzzification

The Center of Area (COA) or centroid defuzzification method is used to calculate the weighted average of 'n' fuzzy set as shown in Fig. 5. By using Eq. (27) the output crisp value of TV has been chosen. Also, the surface view of P and DO is shown in Fig. 6, implemented with MATLAB R2019b, Intel(R) Core (TM) i5-7200 CPU @2.5–2.71 GHz, 8 GB RAM running Windows 10.

$$TV=\frac{\sum_i y_i\times\mu_A\ .(y_i)}{\sum_i\mu_A\ .(y_i)}\tag{27}$$

7. Experimental setup

To evaluate the performance of the proposed DTMA, the experiment of the proposed system has been implemented in MATLAB R2019b on a dataset of 1000 records and 15 attributes each that run at different time periods with the dynamic data of the user. The proposed DTMA algorithm is run at 6 different time intervals to calculate the accurate trust value of each user.

The evaluation is done by calculating the different trust metrics such as Performance (P), Direct Observation (DO), Expected Trust (ET), and Feedback (F) of each user. The DTMA is run at different time periods to find better accuracy. The P metric is calculated by user knowledge, skillset and effectiveness as input, and the result obtained in Eq. (1). The DO is done by monitoring the users in the given context and compare the present activities with the past ones in Eq. (3). The ET is calculated by using both the Historic trust

Table 3
Mamdani fuzzy inference rules for computing Trust Value (TV).

1.	(P==Unskilled) & (DO==Very_Low) & (ET==Very_Low) & (F==Poor) => (TV==Ignored)
2.	(P==Less_skilled) & (DO==Low) & (ET==Very_Low) & (F==Poor) => (TV==Ignored)
3.	(P==Partially_Skilled) & (DO==Average) & (ET==Very_Low) & (F==Poor) => (TV==Ignored)
4.	(P==Skilled) & (DO==High) &(ET==Very_Low) & (F==Poor) => (TV==Ignored)
5.	(P==Highly_Skilled) & (DO==Very_High) & (ET==Very_Low) &(F==Poor) => (TV==Ignored)
6.	(P==Un_Skilled) & (DO==Very_Low) & (ET==Low) & (F==Average) => (TV==Ignored)
7.	(P==Less_Skilled) & (DO==Low) & (ET==Low) & (F==Average) => (TV==Ignored)
8.	(P==Partially_Skilled) & (DO==Average) & (ET==Low) & (F==Average) => (TV==Not_Trusted)
9.	(P==Skilled) & (DO==High) &(ET==High) &(F==Very_Good) =>(TV==Trusted)
10.	(P==Skilled) & (DO==High) & (ET==Average) & (F==Good) => (TV==Partially_Trusted)
11.	(P==Unskilled) & (DO==Very_Low) & (ET==Low) & (F==Good) => (TV==Ignored)
12.	(P==Less_Skilled) & (DO==Low) & (ET==Average) & (F==Good) => (TV==Not_Trusted)
13.	(P==Skilled) & (DO==Average) & (ET==Average) & (F==Good) => (TV==Partially_Trusted)
14.	(P==Highly_Skilled) & (DO==Very_High) & (ET==Very_High) & (F==Good) => (TV==Fully_Trusted)
15.	(P==Highly_Skilled) & (DO==Very_High) & (ET=Average) & (F==Good) => (TV==Trusted)
16.	(P==Unskilled) & (DO==Very_Low) & (ET==High) & (F==Very_Good) => (TV==Ignored)
17.	(P==Less_Skilled) & (DO==Low) & (ET==High) & (F==Very_Good) => (TV==Not_Trusted)
18.	(P==Partially_Skilled) & (DO==Average) &(ET==High) & (F==Very_Good) =>(TV==Partially_Trusted)
19.	(P==Skilled) & (DO==High) & (ET==High) & (F==Very_Good) => (TV==Trusted)
20.	(P==Highly_Skilled) & (DO==Very_High) & (ET==Very_High) & (F==Very_Good) => (TV==Trusted)
21.	(P==Unskilled) & (DO==Very_Low) & (ET==Very_High) & (F==Excellent) => (TV==Ignored)
22.	(P==Partially_Skilled) & (DO==Average) & (ET==Very_High) & (F==Excellent) => (TV==Partially_Trusted)
23.	(P==Skilled) & (DO==High) & (ET==Very_High) & (F==Excellent) => (TV==Trusted)
24.	(P==Highly_Skilled) & (DO==Very_High) & (ET==Very_High) & (F==Excellent) => (TV==Fully_Trusted)
25.	(P==Less_Skilled) & (DO==Low) & (ET==Very_Low) &(F==Excellent) => (TV==Not_Trusted)
26.	(P==Highly_Skilled) &(DO==High) & (ET==High) & (F==Very_Good) => (TV==Trusted)
27.	(P==Skilled) & (DO==Very_High) & (ET==Very_High) & (F==Good) => (TV==Trusted)
28.	(P==Partially_Skilled) & (DO==Very_High) & (ET==Very_High) & (F==Very_Good) => (TV==Trusted)
29.	(P==Highly_Skilled) & (DO==Very_High) & (ET==Very_High) & (F==Good) => (TV==Trusted)
30.	(P==Highly_Skilled) & (DO==Very_High) & (ET==Very_High) & (F==Excellent) => (TV==Fully_Trusted)

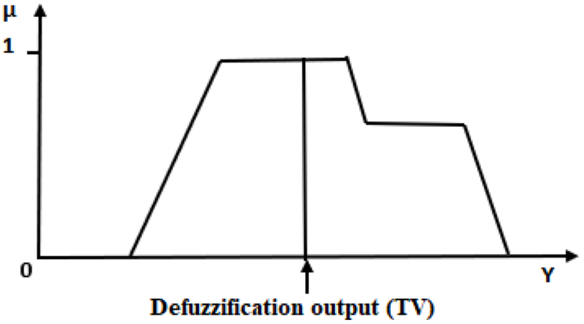


Fig. 5. COA defuzzification

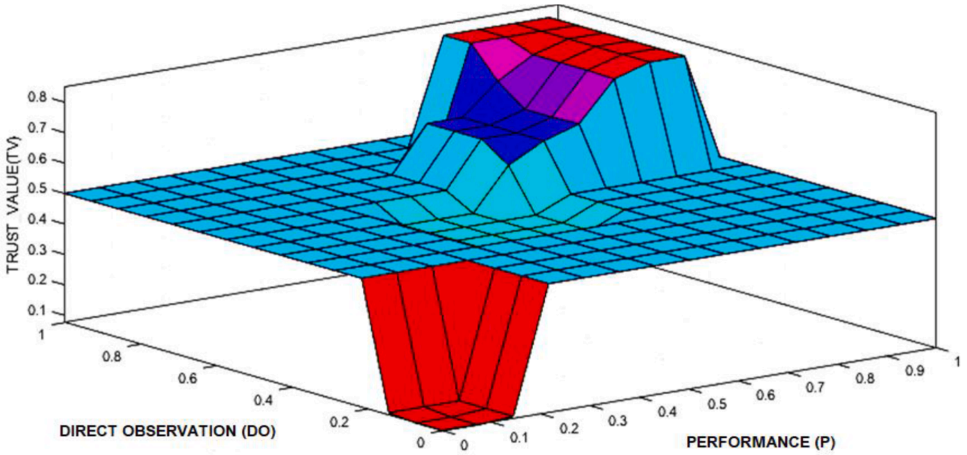


Fig. 6. Surface view of Performance (P) and direct observation (DO)

(H_t) and the Recent trust (R_t) of a person in Eq. (15). The Feedback metric is calculated by finding the Recommendation (*Rec*) and Reputation (*Rep*) of a person in Eq. (17). To get the dynamic value, the TES algorithm is used for calculating the TV which is unstable in nature.

The linguistic variable is assigned to each trust metric according to the scaling factors 0 to 1 and 30 inference rules using the Mamdani fuzzy inference system and written to calculate the TV of each user. Then, the TV is assigned as one of the attributes in ABAC to grant access control, such as to permit or deny.

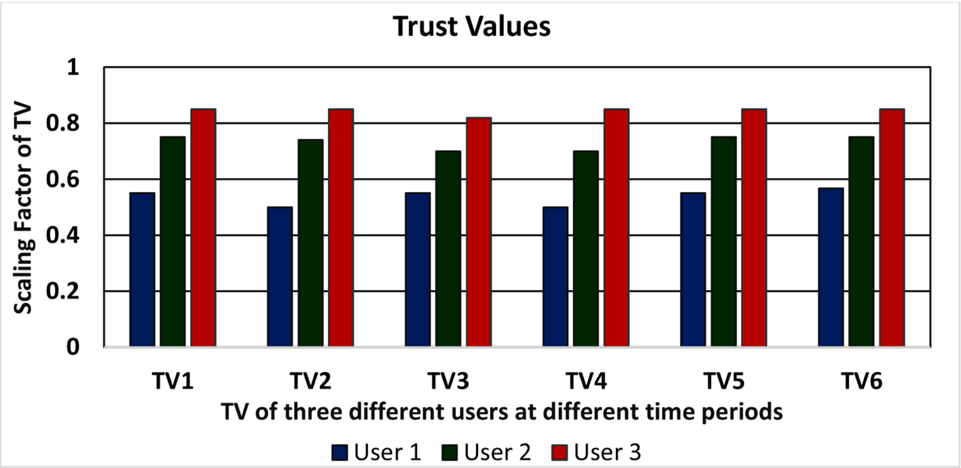
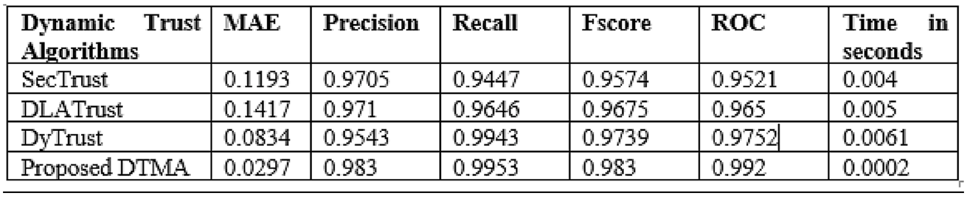


Fig. 7. Trust Values of three different users at various times

Table 4
Performance measuring metrics for DTMA.



Dynamic Trust Algorithms	MAE	Precision	Recall	Fscore	ROC	Time in seconds
SecTrust	0.1193	0.9705	0.9447	0.9574	0.9521	0.004
DLATrust	0.1417	0.971	0.9646	0.9675	0.965	0.005
DyTrust	0.0834	0.9543	0.9943	0.9739	0.9752	0.0061
Proposed DTMA	0.0297	0.983	0.9953	0.983	0.992	0.0002

8. Result and discussion

The TV of the sample of three different users at different time intervals has been chosen to check the oscillation of the predicted TV as shown in Fig. 7. This experiment illustrates that there is a medium change of their computed trust compared to different time periods. The performance measuring metrics such as precision, Mean Absolute Error (MAE), recall, F-Score and ROC have been found on computed trust value to check the accuracy of this proposed DTMA, as shown in Table 4. By using the J48 decision tree algorithm the classification of performance has been done to predict the accuracy of DTMA.

The resultant TV value that is generated by DTMA has been taken for training and testing for classification to get the performance. The samples of 450 TV for training and 150 TV for testing have been chosen to run on J48 decision tree algorithms for acquiring the performance metrics.

To measure the accuracy of the proposed DTMA, the generated TV is classified by using the J48 decision tree algorithm.

8.1. Mean absolute error (MAE)

The prediction error should be minimized by getting the difference of the actual value X_i and predicted value $\widehat{X}_i$ for a given TV dataset by Eq. (29). When compared with other dynamic trust algorithms, the proposed DTMA gives a minimum *MAE* of 0.0297 which is more accurate for prediction, as shown in Fig. 8.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|X_i - \widehat{X}_i\right| \tag{29}$$

8.2. Precision

The precision value is high, by finding the number of users predicted by the proposed DTMA to be trusted and are correctly trusted, as given in Eq. (30). Hence, it has achieved 98.3%, more than the other dynamic trust models shown in Fig. 9.

$$Precision = \frac{TP}{TP + FP} \tag{30}$$

Where *TP* is true positive and *FP* is false positive respectively [4].

8.3. Recall

In this measure, the users who actually trusted are predicted successfully as trusted by the DTMA algorithm by Eq. (31) given in Fig. 10.

$$Recall = \frac{TP}{TP + FN} \tag{31}$$

8.4. F-Score

It is used to measure the accuracy of the proposed DTMA by using precision and recall together, by Eq. (32). Here, the proposed DTMA has achieved 98.2% accuracy, more than the other models as shown in Fig. 11.

$$Fscore = \frac{2 \times Recall \times Precision}{Recall + precision} \tag{32}$$

8.5. ROC

The ROC for proposed DTMA attains the maximum coverage of all trust variables such as Ignored, Not_Trusted, Partially_Trused,

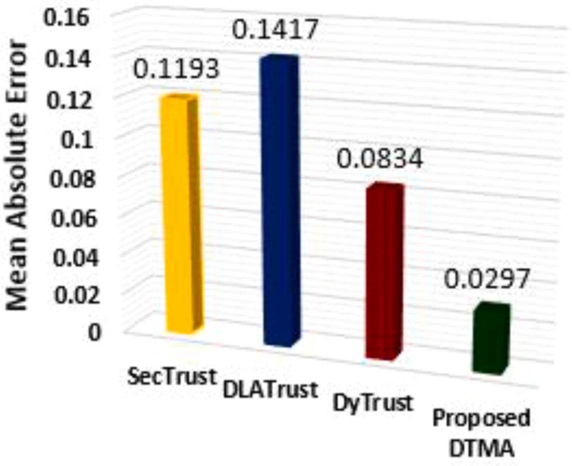


Fig. 8. Mean absolute error metric

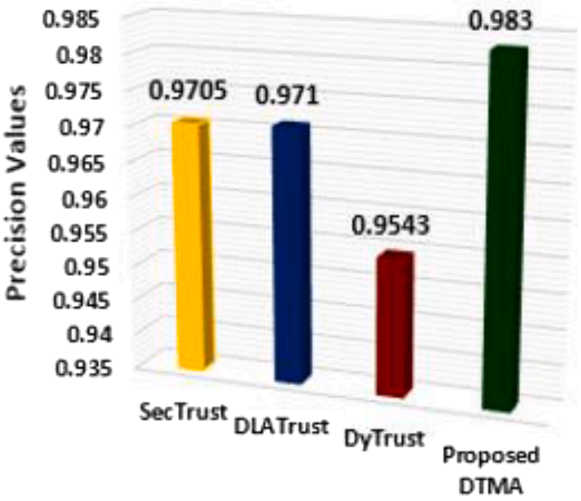


Fig. 9. Precision metric

Trusted and Fully_Trusted. Thus, the proposed DTMA is good in trust computation. Hence, it has been clearly observed from the Table. 4, MAE, precision, recall, F-Score, and ROC have given better results than other dynamic trust model algorithms such as DLATrust, DyTrust, and SecTrust.

8.6. Comparative analysis

The proposed DTMA has been implemented to verify the recent trust value of 100 users that run at different time intervals for obtaining the accuracy of TV. An average comparative analysis has been done for trust parameters, such as Performance (P), Direct Observation (DO), Expected Trust (ET), and Feedback (F) are shown in Fig. 13. It is drawn between the number of users and metric values and it has been clearly observed that ET and F have gradual variations of each person at different time intervals, which shows the minimum MAE in Fig. 13.

Therefore, the performance of the proposed DTMA has been obtained by running the resultant TV for 600 samples of 450 training sets and 150 testing sets on the J48 decision tree algorithm. The resultant metrics such as MAE, precision, recall, F-Score, and ROC are obtained for DTMA and compared with existing algorithms, namely SecTrust, DLATrust, and DYTrust are shown in Table 4. According to this result, the proposed DTMA provides the highest accuracy of 98%, which is more than that of all other dynamic trust models.

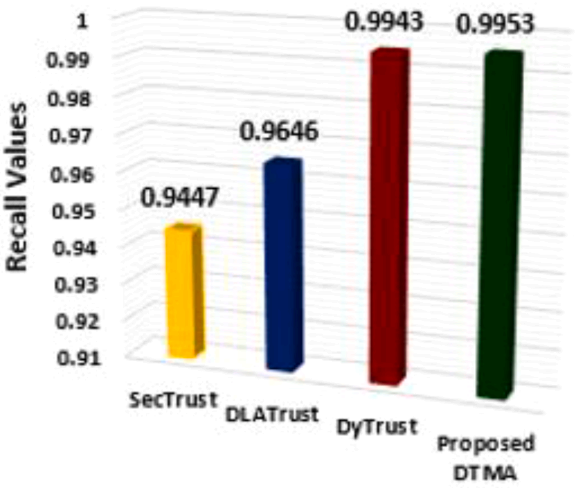


Fig. 10. Recall metric.

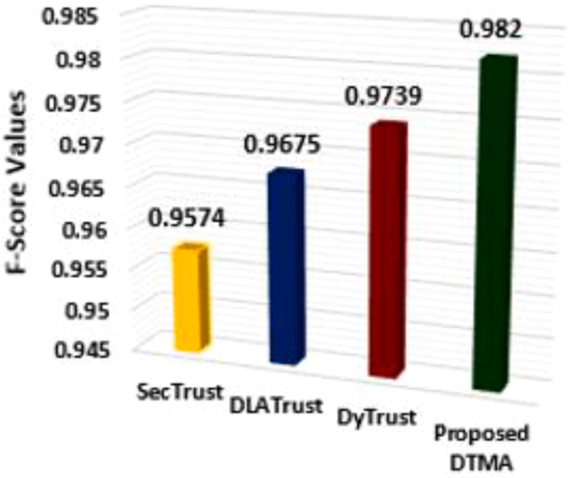


Fig. 11. F-score metric.

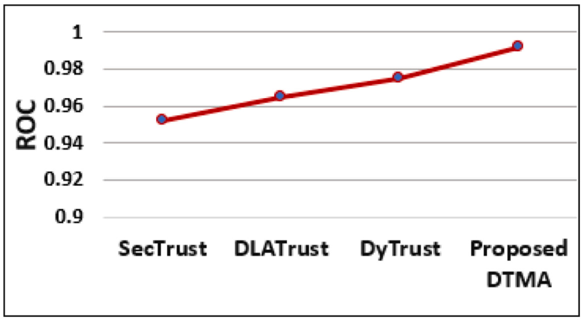


Fig. 12. ROC.

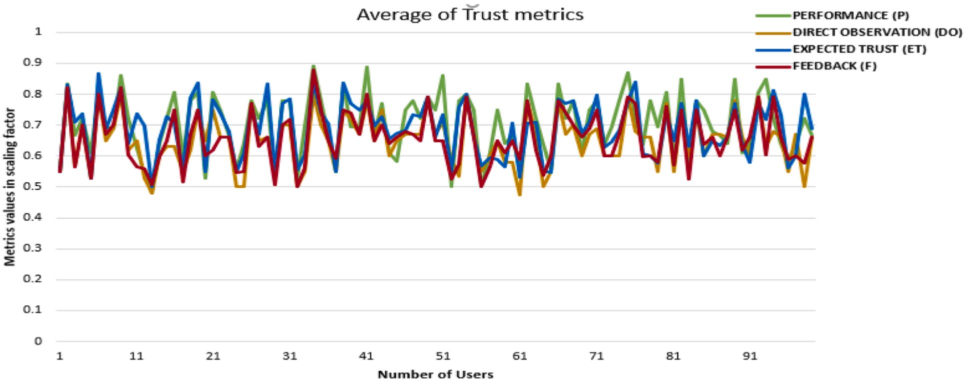


Fig. 13. Average of Trust Value (TV) input parameters.

9. Conclusion

The proposed DTMA calculates the TV of a person and included as one of the attributes in the employee dataset. In order to find the performance of DTMA, the employee dataset of 450 training set and 150 testing set has been run on the WEKA tool with the J48 decision tree. The performance metrics such as MAE, precision, recall, F-Score, and ROC of DTMA are found and compared with other dynamic trust algorithms, namely SecTrust, DLATrust and DYTrust. Hence, the proposed DTMA achieved the MAE of 0.0297, the precision of 98.3%, recall of 99.5%, F-Score of 98.2% and ROC of 99.2%. It shows that the DTMA is 98% accuracy than the other models which is more than all other dynamic trust models. Finally, it is concluded that the DTMA produces better performance metrics to obtain TV. Also, the TV is added as one of the attributes in Attribute Based Access Control (ABAC) to grant or deny the access of resources in companies in a secure manner. Moreover, the proposed DTMA can be applied in any ubiquitous computing area such as IoT, Cloud computing and Big data. Furthermore, providing access privileges to an employee in ABAC for any real scenario is considered as future work.

Declaration of Competing Interest

We don’t have any conflict of interest.

Acknowledgement

We would like to thank SRMIST for providing good resource facilities and a platform to do our research work successfully.

References

[1] Wang J, Liu J. The comparison of distributed p2p trust models based on quantitative parameters in the file downloading scenarios. J Electr Comput Eng 2016: 2016.
[2] Josang A. Trust and reputation systems. Foundation of security analysis and design IV. Berlin: Springer; 2007. p. 209–45.
[3] Das A, Islam MM. SecuredTrust: a dynamic trust computation model for secured communication in multiagent systems. IEEE Trans Depend Secure Comput 2011;2:261–74.
[4] Zhong Y, Bhargava B, Lu Yi, Angin P. A computational dynamic trust model for user authorization. IEEE Trans Dependable Secure Comput 2014;12(1):1–15.
[5] Hoogendoorn M, Jaffry SW, Maanen P-PV, Treur J. Design and validation of a relative trust model. Knowl Based Syst 2014;57:81–94.
[6] Ghavipour M, Meybodi MR. A dynamic algorithm for stochastic trust propagation in online social networks: Learning automata approach. Comput Commun 2018;123:11–23.
[7] Gupta M, Judge P, Ammar M. A reputation system for peer-to-peer networks. In: Proceedings of the 13th international workshop on Network and operating systems support for digital audio and video; 2003.
[8] Nagarajan R, Thirunavukarasu R, Shanmugam S. A fuzzy-based intelligent cloud broker with MapReduce framework to evaluate the trust level of cloud services using customer feedback. Int J Fuzzy Syst 2018;20(1):339–47.
[9] Shashi S, Chatterjee K. A Dynamic Trust Model Based on User Interaction for E-commerce. Recent findings in intelligent computing techniques. Singapore: Springer; 2019. p. 433–43.
[10] Braga DDS, Niemann M, Hellingrath B, Neto FBDL. Survey on computational trust and reputation models. ACM Comput Surv (CSUR) 2018;51(5):1–40.
[11] Srivastava R, Daniel AK. Efficient model of cloud trustworthiness for selecting services using fuzzy logic. Emerging technologies in data mining and information security. Singapore: Springer; 2019. p. 249–60.
[12] Kesarwani A, Khilar PM. Development of trust based access control models using fuzzy logic in cloud computing. J King Saud Univ Comput Inf Sci 2019.
[13] Shunan Ma, He J, Shuai X. An access control method based on scenario trust. Int J Comput Intell Syst 2012;5(5):942–52.
[14] Jayasinghe U, Lee GM, Um T-W, Shi Qi. Machine learning based trust computational model for IoT services. IEEE Trans Sust Comput 2018;4(1):39–52.
[15] Servos D, Osborn SL. Current research and open problems in attribute-based access control. ACM Comput Surveys (CSUR) 2017;49(4):1–45.
[16] Boukerche A, Ren Y. A trust-based security system for ubiquitous and pervasive computing environments. Comput Commun 2008;31(18):4343–51.
[17] Abirami G, Venkataraman R. Attribute based access control policies with trust (ABAC-T) mechanism in pervasive computing. J Adv Res Dyn Control Syst May 2019.
[18] Abirami G, Venkataraman R. Attribute based access control with trust calculation (ABAC-T) for decision policies of health care in pervasive environment. Int J Innovat Technol Explor Eng (IJITEE) 2019;8(7):2278–3075. ISSN.

[19] Jain S. A trust model in cloud computing based on fuzzy logic. In: Proceedings of the IEEE international conference on recent trends in electronics, information & communication technology (RTEICT). IEEE; 2016.
[20] Hu VC, Ferraiolo D, Kuhn R, Friedman AR, Lang AJ, Cogdell MM, Schnitzer A, Sandlin K, Miller R, Scarfone K. Guide to attribute based access control (ABAC) definition and considerations (draft). NIST Spec Publ 2013;800(162).
[21] Chen H, Ye Z, Liu W, Wang C. Fuzzy inference trust in p2p network environment. In: Proceedings of the international workshop on intelligent systems and applications. IEEE; 2009. p. 1–4.
[22] Smari WW, Clemente P, Lalande J-F. An extended attribute based access control model with trust and privacy: application to a collaborative crisis management system. Future Generat Comput Syst 2014;31:147–68.
[23] Smari WW, Zhu J, Clemente P. Trust and privacy in attribute based access control for collaboration environments. In: Proceedings of the 11th international conference on information integration and web-based applications & services; 2009.
[24] Zhao B, Xiao C, Zhang Yu, Zhai P, Wang Z. Assessment of recommendation trust for access control in open networks. Cluster Comput 2019;22(1):565–71.
[25] Singh A, Chatterjee K. Trust based access control model for securing electronic healthcare system. J Amb Intell Human Comput 2019;10(11):4547–65.

G Abirami received a B.E. degree in Computer Science and Engineering from Bharathidasan University, India in 2003., M.E degree from Annamalai University, India in 2008. She is currently pursuing a Ph.D. degree in Computer Science and Engineering at SRM IST, India. Her research interests include cybersecurity, Information security, and Access control of Mobile data.

Revathi Venkataraman currently a Professor and Chairperson in the School of Computing SRMIST, India. She received her Ph.D. degree from SRM University (Currently SRMIST). Her research interests include Trust Computing, Cyber Security, Security enhancements and privacy considerations for IoT. She has received funding from Defence Research and Development Organization. She has also patented few of her innovative ideas in wireless networking.