Insider Attacker Detection Based On Body Language and Technical Behavior Using Light Gradient Boosting Machine (LightGBM)

Mohammed A. Mohammed*, Suhad M. Kadhem, Maisa'a A. Ali
Department of Computer Science, University of Technology, Baghdad, Iraq
* mabdallazez4@gmail.com

Keywords: **Insider Attacker, Insider Threat, LightGBM, Body Language, Detection.**

ABSTRACT

One of most important challenges in cyber security is detecting the insider attacker, where organizations security suffers from the insider attacker, which is an employee (person) with an authorized access to resources and data of an organization then used the access to harm the organization. The insiders are categorizing as active insiders (masquerade and cause physical damage) or passive insider (provide only information). The previous security systems focus on the technical anomaly of an employee to discover the active insider attacker and cannot discover it, if there is not technical anomaly (passive attacker). This paper propose approach to obtain early indicator to passive insider attacker before doing the crime, where body language-based approach used to give earlier alarm of insider attacker. By using three of negative body language gestures (Cross Arms, Clasped Hands, Covering the Mouth) which referred to feeling of insecure, ready for an attack, doubt and a lack of self-confidence, these feelings are the closest to the feelings of the internal attacker. These gestures obtained by use skeleton features from video stream provided by Orbbec Astra Pro camera after passed to rule based classifier to recognize each one of the three body language gestures. Then determined the degree of trust based on the duration of the gesture and the number of occurrences of the same gesture or different gestures and depending on the degree of trust, the organization is alerted to the questionable employees. The test performs on ten of employees, four insider attackers were planted among them, and the results show 70% accuracy of detects the insiders, this approach will detect insider attacker before started his malicious work. Also this paper solves the active attacker, where in reality, the number of malicious events is very small in relation to the number of normal events of the employee, so it was necessary to use a method that accurately characterized this number of harmful behaviors. Several previous studies used complex methods such as deep learning to solve this problem. In this thesis, we used a simpler and faster solution that gave accurate results, where an intelligent approach for detecting insider attacker using Light Gradient Boosting Machine (LightGBM) applied, the cert r4.2 data set used to build and evaluate the model. The results showed the model's ability to distinguish malicious events from data set in its original unbalanced state with accuracy 99.47%.

1.0. INTRODUCTION

The information system integrates not only a digital component (PC, Router, Server, etc.), but also a human component like the user that use the digital component. The threat is no longer so much outside of organizations, whose, firewalls are effective and it no longer targets computers and digital artifacts, which have become more secure; the threat is human which is internal. The human component of the information system constitutes an insider threat to the system's security. A threat that is found inside the organization itself, masters its processes, its firewall and its security policy, whether they are intentional or accidental, malicious or not [1]. The resource of insider threat is insider attacker “Personnel with an authorized access to resources and data of an organization” [2]. Will mention some examples of real-life Insider attackers to prove the problem. Edward Lin (U.S. NAVY 2017) found guilty of wrongly transporting classified material, failing to report foreign contact, mishandling classified information, and disclosing secret information to a foreign citizen. Also shared technical or political classified information pertaining to the Navy’s Special Projects Squadron Two mission with a foreign government. Ivan Lopez (U.S Army 2014) Risk Indicators Depression, anxiety, sleep disturbances, Recent death of mother and grandfather, $14,000 debt, the crimes are death of three soldiers, who left behind wives and children, fourteen other soldiers wounded, Lopez was a husband and father and Lessons learned from this case and several others has shaped DoD response to Insider Risk. Many other case studies of an insider attacker have been publishing in an official website of the Defense Counterintelligence and Security Agency [3]. The insider attacker may be active or passive, the active attacker performs physical operations that cause damage to the organization, while the passive attacker provides information Through what he sees, what he hears, and what he perceives to the opponents or enemies. However, insiders tend to remain hidden and use deceit for activities. One of most important challenges in cybersecurity is detect the inside attacker but how detect the insider attacker, this is the more challenge because in today’s technological era the boundary between friend and rival is growing fuzzier. One of most important challenges in cybersecurity is detect the Insider attacker but how detect the insider attacker, this is the more important question because in today’s technological era the boundary between friend and rival is growing fuzzier [2]. protected approaches should not only monitor host network activities by analyzing technical indicators, but also should identify elements of human behavior, motivation, and intent that may characterize malicious insider threats of employees [3]. Our motivations to deal with the insider attacker is a great threat that the insider causes to organizations, companies, banks and governments, as it leads to huge losses of money and lives in the cases of security organization. The problem we are trying to solve here is how detect the passive insider attacker to avoid losses. In this paper, we propose a model for monitoring employee’s to detect if it is insider attacker or not. Where employee’s body gestures read and technical behavior checked. Body gestures used as pointers to employee’s malicious intentions based on human rule based classifier. While employee’s technical behavior checked by Light Gradient Boosting Machine (LightGBM) classifier. The idea of this paper is to detect the insider attacker, whether he active or passive attacker. The body language act as earlier warning before done the damage, thus even if there is no technical anomaly, it will be possible to identify the attacker before spying or stealing information. We will notice that all previous works use complex methods like deep learning to solve the active attacker without any interest to solve the passive attacker. The reminder of this paper is organize as follows. Previous works will have discussed in sect. 2. Section 3 illustrates the body language. Section 4 data description. The evaluation metric introduces in sect. 5, followed by proposed model in Sect. 6. In Sect 7results and discussion and finally conclusion in Sect 8.

2.0. PREVIOUS WORK

Fang Fang Yuan in [4], presented an insider threat detection method with Deep Neural Network (DNN) based on user behavior. Specifically, the LSTM-CNN framework to find user’s anomalous behavior. The LSTM with CNN gets best result AUC = 0.9449. Qiujian Lv et al [5], proposed a method for the detection of malicious insiders based on the analysis of both user and role behaviors. First, extract several temporal features for every user corresponding to different types of user behaviors. Then, the multiple features reflecting the deviation between the behavior of a user and that of the user group sharing the similar job role with him/her are then calculated. Those significant features, which influence the detection of insider threat significantly, are select by implementing a PCA method. Finally, an efficient detection model is design by leveraging the Isolation Forest Algorithm. They obtain 0.85% accuracy. Adam James Hall, and other in [6] uses the CERT dataset r4.2 along with a series of machine learning classifiers to predict the occurrence of a particular malicious insider threat scenario - the uploading sensitive information to wiki leaks before leaving the organization. These algorithms are aggregate into a meta-classifier, which has a stronger predictive performance than its constituent models. This meta-classifier has an accuracy of 96.2%. Andreas Nicolaou, and other in [7] they attempt to mitigate insider threat problem by developing a machine-learning model based on Bio-inspired computing. The model was developing by using an existing unsupervised learning algorithm for anomaly detection. Where they collected 50,000 samples for experimentation and divided them at rates 66% for training and 34% for testing, and the best result obtained after using optimization algorithms was TP = 91.4%. Minhae JANG and other in [8], they propose an anomaly-based insider threat detection with local features and global statistics over the assumption that a user shows different patterns from regular behaviors during harmful actions. For each user, they built and trained a seq2seq autoencoder model. The training data is the first 60 days of user behavior logs under the assumption that users act normally during this period. The best result obtained was AUC value of 0.9855. Xiaoyun Ye and other in [9], they used the CERT dataset r4.2 along with double-layer HMM structure to model user behavior. They use 50 insiders and obtain 99% accuracy, and they detect drawback in the system when they face the malicious behavior of users without any data accumulation, they can do nothing about the attack. Shuhan Yuan and Xintao Wu in [10], They mentioned deep learning and its relationship with insider attacker processing and a set of challenges and trends. Mehul S. Raval and other in [2], they mentioned Machine Learning (ML) for an insider threat detection, and some case studies on insider threat defense mechanism based on machine learning. There was not study that dealt with LightGBM to solve insider attacker problem as we presented.

3.0. BODY LANGUAGE

The positive points that promotes the use of body language to detect an insider attacker as mentioned in[11]

- Trying to adjust your body language without changing something inside is counterproductive. The nonverbal signals you send out are not controllable: Your body will always want to tell the truth about what you are feeling.
- Body language never lies. “What Is Happening on the Inside Is What You See on the Outside. Everybody speaks a body language.

Body language is visual signals used in people's social intercourse, which include movements, postures, and facial expressions that communicate emotions, attitudes and auxiliary information. Body language is postures and movements, which can communicate emotions and intentions. Verbal language is mainly use to communicate information while body language is mainly use to communicate interpersonal attitudes [12]. A good knowledge of body language helps you to be more aware of what someone else is really feeling. Your body always wants to tell the truth about what you are feeling. Body language is a kind of stethoscope It helps you to examine the possible causes of certain types of behaviour from the outside. Our body instinctively shows on the outside what is happening on the inside, expressions and gestures tend to tell the truth before we can consciously adjust our behaviour. This conscious adjustment is ten thousand times slower than the uncontrollable body language gesture.

What people are experiencing internally will therefore be visible externally. Body language always compensates for the things that are said with words. The body language interpretations are accurate in 60 to 80 percent of situations, if they occur singly or in isolation. If you see, a certain movement occur repeatedly, the likelihood is greater that the interpretation is correct. If within a short period you see a combination of three to five movements that all give a similar signal, you can draw your conclusion with a high degree of certainty [11]. Some negative body language will be used as indicator to insider attacker in this paper because it’s referred to feeling of insecure, ready for an attack, doubt and a lack of self-confidence.

3.1. Crossed Arms

In certain circumstances, crossed arms can indicate a negative or protective attitude or defensive position. You often see this in situations where someone does not feel comfortable or safe. crossed arms as a standard position to indicate you are feeling threatened or insecure is something you see all around the world, the crossed arms shown in Figure 1. [11], [12].

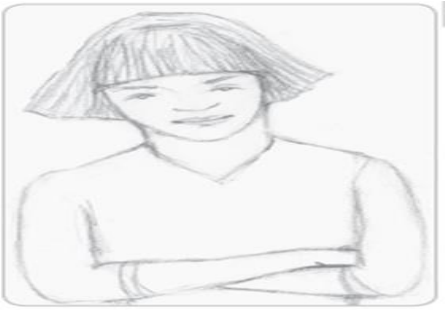

Fig. 1: Crossed arms gesture [11].

3.2. Clasped Hands

In some situations, it is not possible or appropriate to cross your arms. When this happens, some people resort to secondary gestures and positions that carry the same meaning. One of these gestures is with the hands held low (or placed on a table) in a clasped position. This indicates a degree of nervousness, insecurity, and a need for protection. it is similar to the gestures frequently used by liars, clasped hands shown in Figure 2.

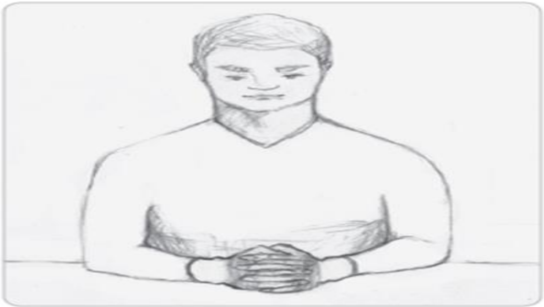

Fig. 2: Clasped hands gesture [11].

3.3. Covering the Mouth

As shown in Fig. 3. Where some people hold a hand close to or in front of their mouths while they are speaking. Sometimes they add a little false cough, as a kind of justification for this movement. In extreme forms, people even push their lips together tightly. This is a protective gesture, designed to conceal doubt and a lack of self- trust from others. Paradoxically, by doing this they actually create a negative impression. What's more, they require their conversation partner to work harder to listen, because in this position they speak less clearly and less distinctly. of course, this makes it much more difficult for them to get their message across. A sudden movement of the hand toward the lips is often a first signal that someone is about to stop speaking. It is possible that the person is momentarily confused or has had a stress-induced blackout, so that she no longer knows what to say. Covering the lips with the hand in this way can also mean that the person has said something she did not intend to say [11].

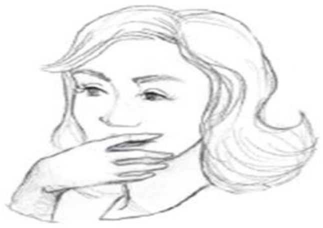

Fig. 3: Covering mouth gesture [11].

4.0. DATASET DESCRIPTION

This section provides an overview of the data used for passive and active attacker. For passive attacker the data obtained from an orbbec astra pro camera, which is a video stream, while CERT r4.2 dataset[13] used for our proposed method to detection of malicious users. Which contains relatively a lot of abnormal events compared to other revisions. A thousand of users generated about 32 million computer usage events during 17 month. The total number of threat events is 7,323. There are seven primary groups of files, which are generated from 1000 simulated users. A description on the contents of each file provided in Table 1; further details can be obtained from the CERT website. In terms of insider threats, version r4.2 of the dataset consists of three primary scenarios described as follows:

1) User who did not previously use removable drives or work after hours begins logging in after hours, using a removable drive, and uploading data to wikileaks.org and leaves the organization shortly thereafter.
2) User begins surfing job websites and soliciting employment from a competitor. Before leaving the company, they use a thumb drive (at markedly higher rates than their previous activity) to steal data.
3) System administrator becomes disgruntled, and downloads a key logger and uses a thumb drive to transfer it to his supervisor's machine. The next day, used the collected key logs to log in as his supervisor and send out an alarming mass email, causing panic in the organization. Leaves the organization immediately [15].

Table 1: Dataset details

Filename	Description
device.csv	Connection and disconnection of Removable devices (e.g., USB hard drive) is describe in this file.
email.csv	Contains logs of user emails.
file.csv	File access activity is provide in this file.
http.csv	This file record the url visited by each user.
logon.csv	Relates to user activity based on logging on and logging off on computing devices.
psychometric.csv	Provides personality and job satisfaction variables for each of the 1000 simulated users.
LDAP	This folder contains a set of LDAP files, which describe the ontology of each simulated user (their role, email, department, supervisor, etc.).

Our focus is on extrapolation of data from the files email.csv, device.csv, file.csv, http.csv and logon.csv. We have chosen to focus on the CERT 4.2 dataset as our data extrapolation methodology is derived from the fact that CERT r4.2 dataset contains a high number of insider threats (Compared with previous and later versions).

5.0. EVALUATION METRIC

To evaluate the performance, we used several typical measures extracted from confusion matrix, including accuracy, Recall, Precision and F1-score as shown in Table 2.

According to the confusion matrix as mentioned in [16] [17], several measurements could be used for examining the performance of the model, the accuracy is usually determined by using the confusion matrix. The recall was use for determining the accuracy of every class known. Precision was also inaccurately classify using the equation below. This helped in calculating the F1 scores.

Table 2: Evaluation metric equations

Metric name	Equation
Accuracy	TP+TN/TP+TN+FP+FN
Recall	TP/TP+FN
Precision	TP/TP+FP
F1-score	2×(Recall× Precision)/(Recall+ Precision)

6.0. PROPOSED MODEL

The aims of the proposed model are detecting inside attacker; the model distinguishes two types of inside attacker (passive and active). The Proposed model consist of two parts, the first one deal with passive attacker, it consists of multi stages to use the body language as early warning of inside tracker. The body language recognized based on the skeleton data provided by Orbbec Astra pro camera. These data contain features used to build rules, which are recognize the body language gestures. The second part deal with active attacker, in this part will use R4.2 dataset to evaluate the Lightgbm, as shown in figure 4.

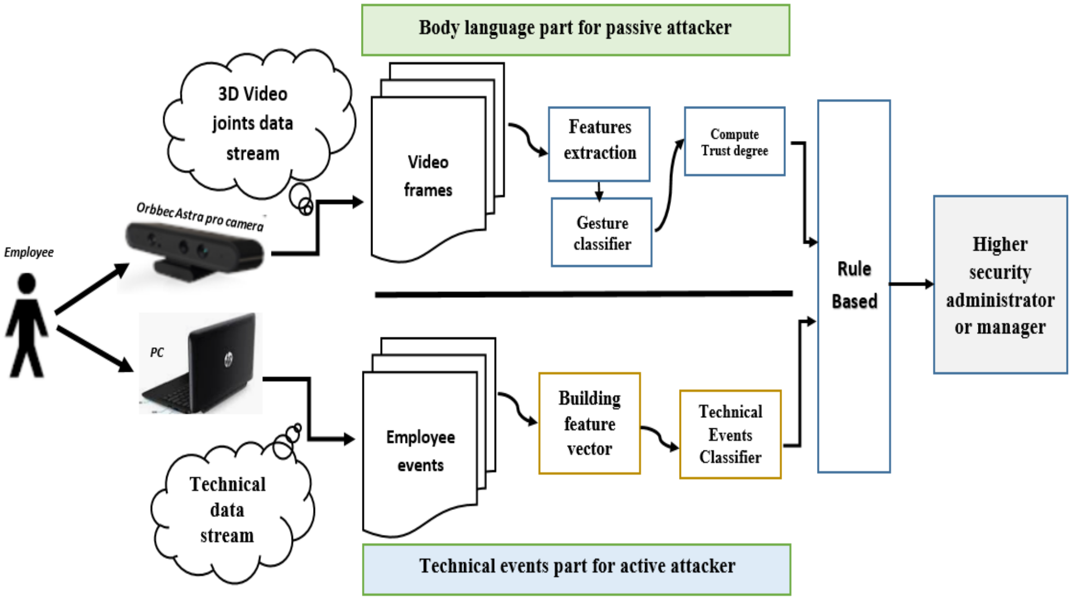
Body language part for passive attacker
3D Video joints data stream
Orbbec Astra pro camera
Employee
PC
Video frames
Features extraction
Gesture classifier
Compute Trust degree
Rule Based
Higher security administrator or manager
Employee events
Building feature vector
Technical Events Classifier
Technical data stream
Technical events part for active attacker

Fig 4. Proposed model.

6.1. Body Language Part of Passive Attacker

The first part is concerned with passive attacker where it reads the employee's body language to get earlier warning if employee is attacker.

6.1.1. Astra pro camera.

The Astra pro camera used to read the video stream, which contain the skeleton data, where it provides 19 joints shown in Fig 5. For each skeleton in each frame. Each joint has position and orientation. The skeleton data is joints coordinates which are x, y and z, where x the dimension on x-axis, y the dimension on y-axis and z the dimension on z-axis which is represent the depth dimension.

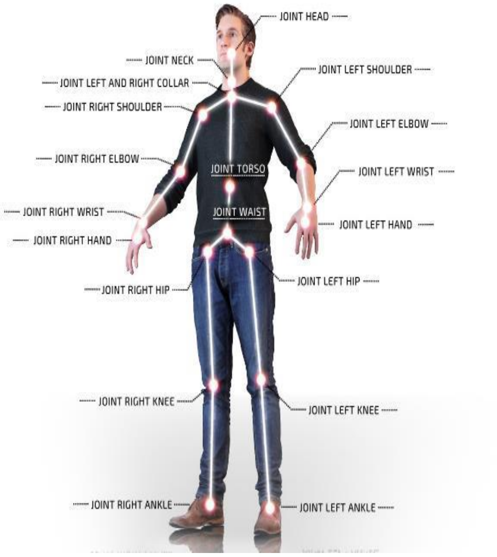

Fig. 5: Astra pro camera skeleton data.

6.1.2. Video frames.

Each frame is read, where features are extracted from it, and it is checked if it contains any body language gesture or not by the classifier that will be explained later.

6.1.3. Feature extraction.

The feature that used to recognized body language gesture extracted from the joints data; where the feature that are related to the body and barely change with time are extracted as static features.

One of the static features is the distance. Distance between two points in three dimensions is given by equation (1) [14], When given two joints coordinates like a, b; where a=(x1, y1, z1), b=(x0, y0, z0); the distance between the joint a and joint b is d, where;

$$d=\sqrt{(x1-x0)^2+(y1-y0)^2+(z1-z0)^2}$$

The equation above will be used to extract the distance feature for each body language gesture then build the rules to recognize the selected body language gestures. For each body language gesture, extract distance features, these features will be used by rule-based classifier to recognize the negative body language gestures, the distance features of each gesture with shorter abbreviated describes in Table 3.

Table 3: Abbreviated description

The gesture	The feature	The abbreviated
Cross Hand	Distance between joint left elbow and joint left shoulder.	dle_ls
	Distance between joint right elbow and joint right shoulder	dre_rs
	Distance between joint right hand and joint left shoulder.	drh_ls
	Distance between joint left hand and joint right shoulder.	dlh_rs
Clasped Hands	Distance between joint right hand and joint left hand	drh_lh
	Distance between joint left hand and joint left shoulder	dlh_ls
	Distance between joint right hand and joint right shoulder.	drh_rs
Covering Mouth	Distance between joint right hand and joint head for right hand	drh_h
	Distance between joint left hand and joint head for left hand	dlh_h

6.1.4. Gestures classifier.

The rule-based classifier is call by the compute employee's trust degree algorithm where it takes a set of values of the distance feature then it checks if there is a gesture or not as shown in Figure 6.

```
Algorithm1: Rule Based Classifier
Input: set of distances features.
Output: body language gesture.
begin
Rule1: if ( | dle_ls - drh_ls |<= thrashold1)&& (|dre_rs -    dlh_rs |<=
         thrashold1) then
                              return "the gesture is cross hand";
Rule2: else if (drh_lh<= thrashold2)&&( dlh_ls> dle_ls) &&( drh_rs> dre_rs)
                 then
                              return "the gesture is clasp hand";
Rule3: else if (drh_h<thrashold3)||( dlh_h<thrashold3) then  return "the
          gesture is covering month";
       end if;
End Algorithm1.
```

Fig. 6: Rule based classifier algorithm.

The distance features passed to the rule-based classifier to discover each frame if it contains one of the body language gesture.

6.1.5. Rules description.

After extracted features the rules built for each body language gesture in algorithm1, describe as shown in Table 4.

Table 4: Rule description

Gesture	Rule	description
Cross Hand	Rule1	The difference between dle_ls and drh_ls and the difference between dre_rs and dlh_rs must be smaller than the predefined threshold selected depend on trial on ten of employees , where measured these two distances of ten employees with deferent sizes and clothes then taken the range.
Clasped Hands	Rule2	The rule of clasped hand is the drh_lh must be less than specific threshold and dlh_ls and drh_rs larger than another threshold, the thresholds selected depend on trial on ten of employees.
Covering Month	Rule3	drh_h must less than threshold for covering mouth with right hand and dlh_h must less than same threshold for covering mouth with left hand, the threshold selected in the same way of above thresholds.

6.1.6. Compute employee's trust degree.

The first algorithm in Fig.6. is used to compute employee's trust degree, the second algorithm in Fig.7. is rule based classifier algorithm, it’s called by the first algorithm to check each frame if contain any of three body language gestures (Cross Arms, Clasped Hands, Covering the Mouth) or not. In algorithm1 when read video, stream the parameters as shown in Table 5. After initialize the parameters , the algorithm implement while loop with condition employee is active and the threshold not reached, the next stage capture the frame from video stream then increment the total number of frames Tf by one, after that pass the frame to algorithm 2 to check if the frame contain body language gesture or not. If the frame contains one of the three body language gestures, the Bf parameter will be incremented by one and compute the new trust degree by equation (2)

$$TD= TD- ((Bf/Tf)*100) \qquad (2)$$

Table 5: Algorithm1 Parameters.

Algorithm	Parameter	Description	Initial
Compute employee's trust degree	Tf	total number of frames	0
	Bf	total number of frames which contain body language gesture	0
	TD	trust degree	100

Where decrement the amount ((Bf/Tf)*100) from trust degree, when employee consume more time in gesture or directly do another gesture the trust degree continue with decrement until TD=threshold will be launch the alarm. If the frame does not contain any body language gesture, the trust degree will be incremented by equation (3) until reach the initial value.

$$TD= TD+ ((Bf/Tf)*100) \qquad (3)$$

```
Algorithm1: Compute employee's trust degree
Input: video stream
Output: Trust degree (TD)
Process:
initialization variables
TD=100, Tf=0, Bf=0
While (TD> threshold and employee is on) do  // employee is on means he/she is sitting in his particular place
        Capture the frame (F)
        Tf =Tf+1
        Call algorithm2  // that take (F) and return (Check of Body Language Gesture)
        if Check is true then      // this mean it is a body language gesture
             Bf =Bf+1
             TD= TD-((Bf/Tf)*100) // decrease the TD when employee body language gesture give a negative signal
        Else if (Bf>0 and TD<100) then
             Bf=Bf-1
             TD= TD+((Bf/Tf)*100) // increase TD when employee body language gesture don't give a negative signal
End While                    //end the while loop
If (TD<= threshold) then
   Launch the alarm    // launch alarm when TD less or equal the threshold
Step3: return (TD)
End Algorithm1.
```

Fig.7: Compute employee's trust degree algorithm.

6.2. Technical Events Part For Active Attacker

The goal is to analyze the employee's body language, to detect the gestures mention above, and then use the frequency of these gestures to obtain earlier warning about insider attacker, as shown in Figure 8. In this section, explain how the model trained and tested based on LightGBM framework and what data preprocessing give the best results.

The model consists of three main parts extract and splitting, LightGBM training with cross validation and independent test as explain in following sections:

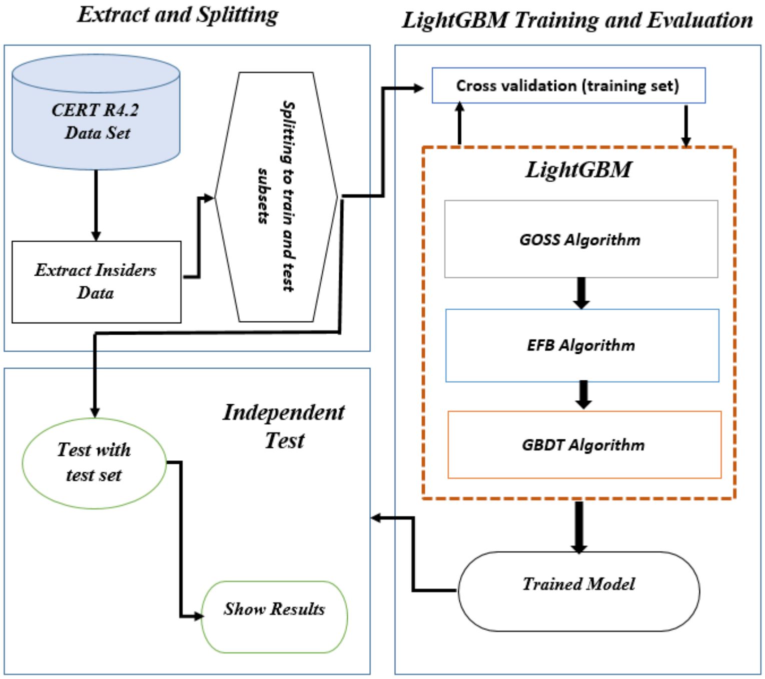

Fig. 8: The general proposed model.

6.2.1. Extract and splitting

The dataset contains a thousand users (whose activities mentioned in the dataset part). Where their activities were record over a period of 17 months, which is an unbalanced dataset. Only 70 of 1000 users represent the insiders, the data of seventy insiders will be extract from the following files (device.csv, email.csv, file.csv, logo.csv and http.csv). Two types for split the dataset will be applied (percentage based and user based), percentage based used 80% for training and 20% for testing, user based used in total 70 users’ where 50 users' for training and 20 users for testing.

As mentioned in previous study have been split dataset by using percentage value, this split-let user's behavior occurred in training and testing set. This is our justification for taking another type of division (user based) in this paper. Where, users in the training set have not the same users in the test set. This would be a realistic indication of the model's ability to distinguish as well give the model reliability and generalization to distinguish new users.

6.2.2. LightGBM training and evaluation

LightGBM algorithm used to training and testing a model to make it capable of distinguish malicious events as shown in the Figure 8 Cross validation used to increase the efficiency of the model and achieve the greatest possible accuracy, where it was use 5-Fold cross validation.

Gradient boosting decision tree (GBDT) is a useful algorithm that can be used for both classification and regression problems. Recently, Ke et al [15] proposed a novel gradient boosting decision tree (GBDT) algorithm named LightGBM , which utilize two novel techniques: Gradient-based One-Side Sampling (GOSS) along with Exclusive Feature Bundling (EFB) to deal with the huge number of data samples along with massive amount of features respectively as illustrated in Figure 8.

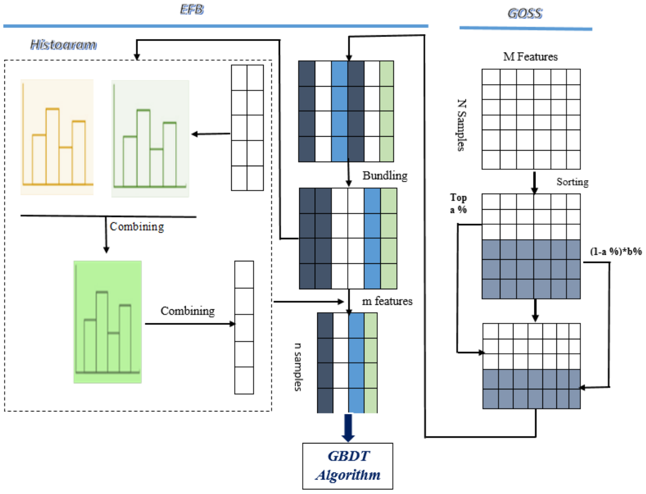

Fig. 8: Gradient-based One-Side Sampling (GOSS) along with Exclusive Feature Bundling (EFB).

GOSS keeps all the examples with large gradients and conducts random sampling on the examples with small gradients. EFB algorithm can bundle many exclusive characteristics to the much fewer dense characteristics, which can dramatically avoid unnecessary calculation for zero feature values. And so on these two algorithms deal with the huge number of data samples along with massive number of features. The LightGBM algorithm can quickly process large amounts of data. It was developed as an open source project by Microsoft. The Light Gradient Boosting algorithm is explained in Figure 9.

```
The LightGBM algorithm
Input: Training data D = {(χ1, y1), (χ2, y2), ..., (χN, yN)}, χi_χ, χ ⊆ R, yi_{−1,+1}; loss function: L(y, _(χ));
iterations: M; sampling ratio of large gradient data: a; sampling ratio of small gradient data: b;
Merge mutually exclusive features(i.e. features never take nonzero values simultaneously) of χi, i = {1, ...,N} by exclusive feature bundling(EFB) method;
Initialize _0(χ) = arg minc_Ni L(yi, c);
for m = 1 toMdo
    Compute absolute values of gradients:
    $ri = \left|\frac{\partial L(y_i, \theta(x_i))}{\partial \theta(x_i)}\right|_{\partial\theta(x)=\theta_{m-1(x)}}, i = \{1, \dots\dots, N\}$
    Resampled dataset by Gradient-based One-Side Sampling (GOSS) method:
    topN = a × len(D); randN = b × len(D);
    sorted = GetSortedIndices(abs(r));
    A = sorted [ 1 : topN]; B = RandomPick(sorted[ topN : len(D)] , randN); D_ = A + B;
    Compute information gains:
    $Vj(d) = \frac{1}{n}\left(\frac{\left(\sum_{x_i \in A_l} r_i + \frac{1-a}{b}\sum_{x_i \in B_l} r_i\right)^2}{N_l^j(d)} + \frac{\left(\sum_{x_i \in A_r} r_i + \frac{1-a}{b}\sum_{x_i \in B_r} r_i\right)^2}{N_r^j(d)}\right)^{nt}$
    Get a new decision tree $\theta_m(X)'$ on set $D'$
    Update $\theta_m(X) = \theta_{m-1}(X) + \theta_m(X)'$
end for
return $\theta'_M(X) = \theta_M(X)$
```

Fig. 9: The LightGBM algorithm.

The LightGBM algorithm includes several parameters, termed hyper parameters. The hyper parameters have a significant impact on the performance of LightGBM algorithm. They are typically set manually and then tuned in a continuous trial and error process.

6.2.3. Independent test

It is an independent test on data that the model has not previously seen, this test to measure the ability of the model to distinguish malicious activities to the user and to give greater reliability to the model.

7.0. RESULTS AND DISSECTION

All data processing tasks in this paper are perform using a PC with Intel Core. i5 2467M @ 1.60GHz CPU and 8.0 GB Dual-Channel DDR, the C# programming language used to paper Implementation.

7.1. Body language part results

Experiments were conduct on a ten of employees, where have planted four of them (insider attackers), results demonstrated competence in distinguishing the body language gestures that shown in Figure 10.

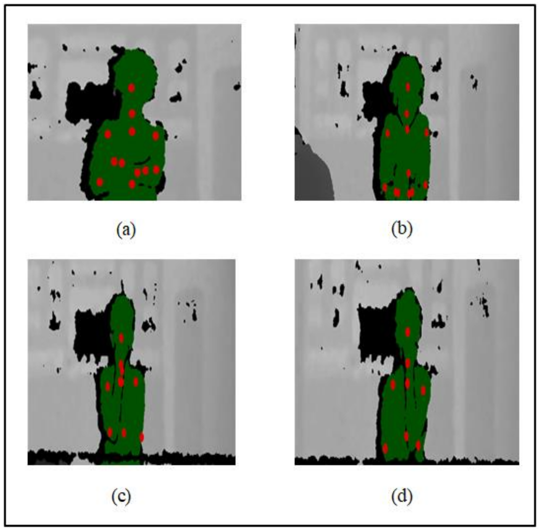

Fig. 10: The body language gestures: (a) cross arms gesture, (b) clasped hands gesture, (c) covering mouth by right hand and (d) covering mouth by left hand.

Also the degree of trust had calculated to the employees according to the time consumed with gesture and the number of gestures, as shown in Table 6.

Table 6: The number of gestures

Employees	Cross Hand	Clasped Hands	Covering Mouth	Total time	gestures	Trust degree	Actual class	Launch alarm
1	×	×	×	0s		100%	Not insider	no
2	×	×	✓	10s		94%	Not insider	no
3	×	✓	×	23s		90%	Not insider	no
4	×	×	×	0s		100%	insider	no
5	✓	✓	×	130s		66%	insider	no
6	✓	✓	×	67s		85%	Not insider	no
7	✓	✓	✓	192s		46%	insider	yes
8	✓	×	✓	90s		78%	Not insider	no
9	✓	✓	✓	200s		43%	Not insider	yes
10	✓	✓	×	230s		39%	insider	yes

In this paper, the interpretation of the body language was adopted based on two important factors: the duration of gesture, the frequency of the gesture itself, or other gesture having the same meaning. The degree of trust affected by these two factors, where same gesture sometimes has a great impact on the degree of trust and sometimes slight. As example when one employee do the gesture for half minute the trust degree of this employee will greater than other do it for ten second, also the same when do more than one gesture with same meaning. The results in table1 affected by the duration the employee consume in gesture and how many gestures the employee do, this two factor demonstrate why employee five and employee six with deferent trust degree although the two employee do the same gestures Clasped Hand and Cross Hand. The confused matrix of the results is shown in Table 7.

Table 7: confused matrix.

Total instances=10	**Actual insider**	**Actual not insider**
Predicate insider	2	1
Predicate not insider	2	5

True Positives (TP)=2, True Negatives (TN)=5, False Positives (FP)=1, False Negatives (FN)=2.

The Accuracy = ((TP+TN)/total)*100 = ((2+5)/10)*100=(7/10)*100= 70%.

Misclassification Rate= ((FP+FN)/total)*100 = ((1+2)/10)*100=(3/10)*100= 30%.

These results show that body language reveals an insider attacker, when he intends to do a job that harms the organization, In the status that the insider attacker does not appear on it the specific gestures, it cannot be detected as in the employee 4. While the fifth employee had two posts and his degree of trust was 66, which did not exceed the limit, So the system did not launch an alarm.

Employees 1, 2, 3,6,8 and 9 were trustworthy so they did not launch the alarm. The Employees 10, 7 were not so trustworthy for them so the alarm was triggered.

The threshold used in the results above is 50%, where if the degree of confidence is less than 50, the alarm will be triggered Otherwise not. It is possible to control the acceptable threshold of confidence according to the vision of the top management of the organization and according to the sensitivity of the information that the employees deal with, their health and psychological conditions and the environmental factors in the workplace. Therefore, these results can change according to the threshold.

7.2. Technical events results

7.2.1. Splitting Data and Class Distribution

Total events of the seventy insiders is 207440 events with five features (id, date, user, pc, activity) the class (1=200117 event, 0= 7323 event). Where, 0 is malicious event and 1is non malicious event.

Percentage based splitting is 80% for training and 20% for testing as shown in Table 8.

Table 8: Percentage based Splitting

class	**1**	**0**
training set	160100	5852
testing set	40017	1471

User Based Splitting, 50 users selected Randomly, their data extracted for training, and remainder 20 users extracted their data for testing as shown in Table 9.

Table 9: User Based Splitting

class	**1**	**0**
training set	116079	3670
testing set	34156	1498

7.2.2. Implement LlightGBM with percentage based splitting

The results of training the model with training set and testing it with test set is shown in Table 10 and Table 11, respectively.

Table 10: Confused matrix of training lightgbm with training set (Percentage based).

	Predicted		
actual	**0**	**1**	**recall**
0	5.608	244	**0.9583**
1	80	160.020	**0.9995**
precision	**0.9859**	**0.9985**	

The confused matrix in Table 10 represent the results of the best model among five models of cross validation models. While, the average accuracy of the five models was 99.3% and average F1Score was 97.19%.

Table 11: Confused matrix of test lightgbm with test set (Percentage based).

	Predicted		
actual	**0**	**1**	**recall**
0	1.311	160	**0.8912**
1	60	39.957	**0.9985**
precision	**0.9562**	**0.9960**	

The confused matrix in Table 11 represent the results of the test the model with test data. While, the accuracy on test data was 99.47 %, the Auc was Auc 99.79 % and F1Score was 92.26 %.

7.2.3. Implement LightGBM with user based splitting

The dataset is splitting here on the basis of the user. Where the test set contains users who are not in the training set. The results of training the model with training set and testing it with test set is shown in Table 12 and Table 13, respectively.

Table 12: Confused matrix of training lightgbm with training set (user based splitting).

	Predicted		
actual	**0**	**1**	**recall**
0	3.635	35	**0.9905**
1	11	116.068	**0.9999**
precision	**0.9970**	**0.9997**	

The confused matrix in Table 12 represent the results of the best model among five models of cross validation models, That trained on data splitted based on the user. While, the average accuracy of the five models was 99.8% and average F1Score was 96.7%.

Table 13: Confused matrix of test lightgbm with test set (user based splitting).

	Predicted		
actual	**0**	**1**	**recall**
0	838	660	**0.5594**
1	44	34.112	**0.9987**
precision	**0.9501**	**0.9810**	

The confused matrix in Table 13 represent the results of the test the model on test set of 20 users the model hasn't seen before. While, the accuracy on test data was 98.03%, the Auc was Auc 97.43% and F1Score was 70.42%.

7.2.4. Comparison between percentage based and user based

The comparison was made on the results of the test group for each of the two divisions as shown in the Table14.

Table 14: Comparison between percentage based and user based

matric	**Percentage based**	**User based**
accuracy	99.47 %	98.03 %
Auc	99.79 %	97.43 %
F1 score	92.26 %.	70.42 %.

As it is clear from the Table 14 that the percentage based splitting is more accurate than user based splitting, the reason for this is that the behavior that was distinguished in the test set belongs to the same users in the training set. The accuracy in the case of user based splitting is more realistic because the users in the test set have not seen the model before and this corresponds to the situation of the new employee, which we want to find out if he is an insider attacker or not.

7.2.5. Comparison with previous studies

All previous studies have focused on the use of complex methods such as deep learning, and have dealt with data in a manner that does not suit the important nature of the internal attacker. The Table 15 shows the method of splitting the data in each work with some measurements for comparison.

Table 15: Comparison with previous studies

paper	**splitting**	**accuracy**	**AUC**	**F1-score**	**TP**
This work	80%-20% randomly	**99.47 %**	**99.79 %**	**92.26 %.**	**-**
	50 user training-20 users testing	**98.03 %**	**97.43 %**	**70.42 %.**	**-**
[4]	~70%-~30%	-	94.49%	-	-
[5]	Basd on user's time	85%	-	-	-
[6]	Use 7260 instances only	96.2%	-	-	-
[7]	66%-34%	-	-	-	91.4%

paper	splitting	accuracy	AUC	F1-score	TP
[8]	Basd on user's days	-	98.55%	-	-
[9]	Basd on user's time	99 %	-	-	-

We note that this work is distinguished by the fact that it adopted two divisions, one of which was tested on 20 users that the model had not seen before, and this did not happen in any of the previous works, in addition to using LightGBM algorithm as it was not used in any of the previous works.

When the behavior belongs to the same user in both the training and testing sets, the identification of the malicious events becomes more clear in this case the model give accuracy 99.47 %. While, when we test the behavior of new users that the model has not seen during the training, the result becomes more realistic, reliable and generlization in this case the model give accuracy 98.03 %., and this is because in the real world, the organizations want to discover new employees if they are insiders or not because the new employee we do not have previous data about him. Also, when using the model in a specific institution, it must be able to detect insiders from its employees, even if it is not trained on data belonging to them.

8.0. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed the insider threat detection model by use Light Gradient Boosting Machine (lightgbm). Because insider threat manifest in various forms, it is not practical explicitly model it. We frame insider threat detection as classification task based on events performed by employee.

the security of many organizations, banks and governaments suffer from the insider attacker, which is an employee with an authorized access to information of an organization then used the access to damage the organization. In reality, the malicious events is very little in relation to the normal events of the employee, so it was necessary to use a method that accurately distinguish this harmful behaviors. Several previous studies used complex methods such as deep learning to solve this problem. we used a simpler and faster solution that gave accurate results, where an intelligent approach for detecting insider attacker using (LightGBM) applied, the cert r4.2 data set used to trainining and test the model. Where two types of division were adopted (percentage based splitting and user based splitting) . The results showed the model's ability to distinguish malicious events from data set in its original unbalanced state with accuracy 99.47 % In case and 98.03% in case of user based.

Lightgbm algorithm bypassed the most important problem for the attacker's data was an imbalance, as it gives high accuracy in detect the malicious events and it is less complexity compared with other method.

REFERENCES

[1] Pierre-Emmanuel Arduin,"Insider Threats", Volume 10, © ISTE Ltd 2018.

[2] Mehul S. Raval, Ratnik Gandhi, and Sanjay Chaudhary ,"Insider Threat Detection: Machine Learning Way", © Springer Nature Switzerland AG 2018.

[3] https://www.cdse.edu/resources/case-studies.html

[4] Yanan Cao, "Insider Threat Detection with Deep Neural Network ", International Conference on Computational Science 2018.

[5] Qiujian Lv, Yan Wang, Leiqi Wang and Dan Wang, "Towards A User and Role-Based Behavior Analysis Method for Insider Threat Detection", Proceedings of IC-NIDC ©IEEE, 2018.

[6] Adam James Hall, Nikolaos Pitropakis, William J Buchanan and Naghmeh Moradpoor, "Predicting Malicious Insider Threat Scenarios Using Organizational Data and a Heterogeneous Stack-Classifier" , arXiv:1907.10272v1 [cs.CR] 24 Jul 2019.

[7] Andreas Nicolaou , Stavros Shiaeles and Nick Savage, “Mitigating Insider Threats Using Bio-Inspired Models” , Appl. Sci. 2020, 10, 5046; doi:10.3390/app10155046.

[8] Minhae JANG, Yeonseung RYU and Jik-Soo KIM, “Against Insider Threats with Hybrid Anomaly Detection with Local-Feature Autoencoder and Global Statistics (LAGS)”, Copyright c_ 2020 The Institute of Electronics, Information and Communication Engineers.

[9] Xiaoyun Ye , Sung-Sam Hong , and Myung-Mook Han, “Feature Engineering Method Using Double-Layer Hidden Markov Model for Insider Threat Detection”, International Journal of Fuzzy Logic and Intelligent Systems, Vol. 20, No. 1, March 2020, pp. 17-25http://doi.org/10.5391/IJFIS.2020.20.1.17.

[10] Shuhan Yuan and Xintao Wu, “Deep Learning for Insider Threat Detection: Review, Challenges and Opportunities”, arXiv:2005.12433v1 [cs.CR] 25 May 2020.

[11] Kasia Wezowski and Patryk Wezowski, “Without Saying a Word”, Kasia Wezowski and Patryk Wezowski 2018.

[12] Jianxue Yin, “Body Language Classification and Communicative Context”, International Conference on Education, Language, Art and Intercultural Communication (ICELAIC) 2014.

[13] https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=508099.

[14] Jiande Sun,” View-invariant gait recognition based on kinect skeleton feature”, Springer Science+Business Media, LLC, part of Springer Nature 2018.

[15] Owen Lo, William J Buchanan, Paul Griffiths and Richard J Macfarlane , “Distance Measurement Methods for Improved Insider Threat Detection” , Edinburgh Napier University, w.buchanan@napier.ac.uk, Academic Editor: Gerardo Pelosi Copyright © 2017.

[16] Dhafar Hamed Abd , Ahmed T. Sadiq, and Ayad R. Abbas, “Political Articles Categorization Based on Different Naïve Bayes Models” , © Springer Nature Switzerland AG 2020, M. I. Khalaf et al. (Eds.): ACRIT 2019, CCIS 1174, pp. 286–301, 2020.

[17] Dhafar Hamed Abd , Ahmed T. Sadiq, and Ayad R. Abbas, “Classifying Political Arabic Articles Using Support Vector Machine with Different Feature Extraction” , © Springer Nature Switzerland AG 2020, M. I. Khalaf et al. (Eds.): ACRIT 2019, CCIS 1174, pp. 286–301, 2020. https://doi.org/10.1007/978-3-030-38752-5_23.

[18] J. M. Borky, T. H. Bradley ,” Protecting Information with Cybersecurity”, © Springer International Publishing AG, part of Springer Nature 2019.