Reducing the False Alarm Rate in Intrusion Detection System by Providing Authentication and Improving the Efficiency of Intrusion Detection System by using Filtered Clusterer Algorithm using Weka Tool

Pratik Jain, Ravikant Kholwal, Tavneet Singh Khurana

Abstract**: *An IDS supervises network traffic by searching for skeptical activities and previously determined threats and sends alerts when detected. In the current times, the splendors of Intrusion detection still prevail censorial in cyber safety, but maybe not as a lasting resolution. To study a plant, one must start with roots, so Cambridge dictionary defines an intrusion as "an occasion when someone goes into an area or situation where they're not wanted or expected to be". For understanding the article, we will characterize interruption as any network movement or unapproved framework identified with one or more PCs or networks. This is an interpretation of permissible use of a system attempting to strengthen his advantages to acquire more noteworthy access to the framework that he is at present endowed, or a similar client attempting to associate with an unapproved far-off port of a server. These are the interruptions which will cause from the surface world, a bothered ex-representative who was terminated recently, or from your reliable staff. In this proviso, the fair information is found as an attack when the case is a false positive. Here they are zeroing in on this issue with a representation and offering one answer for a similar issue. The KDD CUP 1999 informational index is utilized. Here we dropped the number of counts and considered the OTP authentication system. In the result of this test, it may be very well seen that on the off chance that a class has a higher number of checks, at that point this class is believed to be an anomaly class. In any case, it will be considered an oddity if the genuine individual is passing the edge esteem is considered an intruder. One arrangement is proposed to distinguish the genuine individual and to eliminate false positives.***

Keywords**: *Anomaly Detection System (ADS), Bogus positive, Clustering, Data mining, Detection rate, Ensemble, False alert rate, K-Means.***

Manuscript received on April 06, 2021.
Revised Manuscript received on April 15, 2021.
Manuscript published on April 30, 2021.
***** Correspondence Author

Pratik Jain*, Computer Science, IPS Academy, Institute of Engineering and Science, Indore, India. Email: pratikjain@ipsacademy.org

Ravikant Kholwal Computer Science, Indian Institute of Information Technology, Design and Manufacturing, Jabalpur, India. Email: rkant4112@gmail.com

Tavneet Singh Khurana, Computer Science, IPS Academy, Institute of Engineering and Science, Indore, India. Email: tavneetsingh2000@gmail.com

I. INTRODUCTION

Within the final two decades, with the development of computer innovation, the security of the arranged framework has ended up a significant issue, as computer innovation has been abused by numerous individuals all over the world in a few regions, this leads to network invasion day by day over the past some years. It is extremely important to locate a predominant method to watch the information as it contains exceptionally susceptive data. Today, there is extremely interminable security, for example, information encryption, VPN, and fire divider. They were good within them. Still, they have worth to utilize but they are missing to distinguish the assaults by a crack. Notwithstanding, interruption distinguishing proof may be a moveable one that can grant energetic affirmation to the organized security in invigilating ambushes and slug/antithetical attacks. Network Intrusion Detection Systems (NIDS) typically stick to one of the three plan models. These ordinary IDS plan classifications are signature-based, anomaly-based, and protocol modeling. Each plan model has its qualities and failure, and numerous gadgets are a combination of the three models. This is the nonexclusive plan: generally, all NIDS gadgets have a firm reliance on mark-based location to some degree. This innovation explanation bundles for select examples identified with familiar assaults. Signature-based discovery is similarly helpful to unfasten, see, and update, and furthermore, it is reasonable at emphatically distinguishing known assaults. Despite this, it has one drawback that they may not find out unknown or modified invasions.Intrusions are attainable by using different approaches and techniques and are grouped broadly in the categories below:

Signature Based Detection Systems

A signature-based intrusion detection system is based on matching the incoming request's authenticity by comparing it with the predefined signature. These sensing issues on the consistent up to dating signature as it is hostile known intrusions. Moreover, it is incapable to distinguish new interruptions and novel assaults, as its common imperfection. The single accommodation is that it incorporates a more prominent acknowledgment rate than the peculiarity intrusion detection.

A. Anomaly Based Detection System

Anomaly-based interruption recognition framework had pulled numerous specialists because of its capacities of recognizing novel/ fiction attacks. There is some type of unrecognized assault that the machine plot is not cognizant of during exercitation. For this, the Fiction attack location system of working is proposed, ABIDS has two prime advantages over SBIDS, and the absolute initial is the capacity to distinguish extraneous and "zero-day" intrusion i.e. When the flaw is detected without any prior knowledge to the system security council. This is done by contrasting the inconspicuous activity and that of deviation from them. The subsequent one is the regular activity profiles are revamped for system, network and in the future structure it much solidified for an aggressor to know with sureness what practices it can eliminate without getting found. The competency of the system depends upon how enjoyably it is instrumented and taken a stab at all conventions. The overall downside of inconsistency detection is delimiting its standard set.

B. Protocol Modeling

Protocol modeling is executed by assessing network inconvenience for exceptional protocol clamoring and exasperating activity with totally deputed conventions or conventions that are obscure to the scheme. Protocol modeling depends on different numerous information sources to portray what ordinary convention action is. Nonexclusive sources for this information can incorporate convention detail RFCs, conceivable applications that work out that convention, and a whole investigation of typical network activity.

II. LITERATURE TRACERY

In this paper, they utilized a Hybrid detection system that relies upon information mining scientific categorization and clustering methods [1]. In his exploration, says it should worry to zero in on the utilization of information mining methods along with Embattle trees and face direct machines for anonymity discovery. The aftereffect of tests gives the idea that the computation C4.5 has a more unmistakable limit than SVM in perceiving network abnormality and wrong alarm rate by using 1999 KDD cup data [2]. In this, the Algorithm exploits a feature reduction algorithm called symbolic dynamic filtering (SDF) [3]. In SDF, time-series information is isolated for producing symbolic arrangements that at that point create probabilistic finite-state automata (PFSA) to go to as highlights for design scientific classification [4]. In this paper, at the point when commotion improves it is initially considering the conduct of the inclining technique since it could change the capacity of separating exact standards. Effectiveness is figured out considering 3 metrics: Max rule confidence, Precision, and Recall [5]. In this, they, utilized Cluster Investigation for Anomaly Detection. Here it has been utilized as a basic K-mean clustering method. K-mean clustering could be a straightforward, outrageous calculation. It is less PC significant than numerous different calculations, and appropriately it may be a predominant choice when the dataset is colossal [6].

K-mean preferable alternative when the dataset is enormous since It is less system escalated than numerous different algorithms [6]. In this, for Lofty spatial collective data, definitive numbers of bunches given by clients are terrible for estimation, since it prompts non-suitable data overseeing or its prompts distinctive irregularity [7]. Arrange interruption location frameworks utilize stamp-dependent methodologies or information scooping-based strategies which for the foremost portion depend on stamped information. Quirk arranges intrusion recognizable proof strategy based on Foremost Component Investigation for information diminish and Fluffy Versatile Reverberation Hypothesis for the classifier is presented [8] They used, Modern in cross interruption recognition structure utilizing shrewd unique herd based harsh set for highlight election and enlightened herd advancement for interruption information order [9]. They opt for the technique is concentrated over reenactment and connected to a mechanical relevant examination. The result propounds attainable utilize for energetic underway organization. Its praxis Calculation for the structure of an astir arrangement foundational on work request information [10]. Their approach was to apply in cross views for interruption scooping frameworks rooted in information scooping. The significant technique is bundling examination with the objective of corrected unmasking percentage and decrease in bogus warning percentage [11]. They used, Irregularity activity discovery framework positioned on the Entropy of net lineaments, and SVM is set side by side. Amalgam strategy that's the addition of both entropy of organizing highlights and plan of action vector machines are set side by side with individual methods [12]. He, chips away at the roomy relative investigation of several anomaly detection programs for diagnosing various network intrusion [13]. They advertised another columned on matrix and density gathering calculation that's accommodating for unsupervised peculiarity recognizable proof [14]. In this paper, it shows the description of all the algorithm used in weka tool [15].

III. PROBLEM RECOGNIZANCE

The term Intrusion detection can be featured by the Intrusion detection system in which the terms Intrusion are unwanted access and all. In the detection system, it detects irregular activity automatically and it secures the network and guards it. The methods for the discovery of the atypical action are systematized into two bunches: -

A. Predefined Intrusion Behavior

First, it buffers the impression of intrusion or the malevolent behavior, and then it judges the intrusion according to the acquired pattern. It can discover predetermined examples of intrusions and furthermore, it has larger acknowledgment precision and having a moo fake alert rate.

B. Predefined Normal Behavior

It studies the traditional conduct by storing the impression of an individual's traditional conduct into an information cluster and in the event that if the disparity is adequately serious, an abnormal act is declared.IDS desires sublimate chastity, disclosure percentage and inferior bogus alert rate. All in all, the presentation of IDS is assessed in term of accuracy, disclosure rate and bogus alert rate as in the accompanying equations:

(1) Accuracy = (True Positive + True Negative) / (True Positive + True Negative + False Positive + False Negative)

(2) Disclosure Rate = (True Positive) / (True Positive + False Positive)

(3) Bogus Alert Rate = (False Positive) / (False Positive + True Negative)

Table- I: Common Behavior of Intrusion Detection Information

Absolute	**Anticipated Normal**	**Anticipated Strike**
Regular	True Negative	False Positive
Intrusions	False Negative	True Positive

1. True positive implies intrusion input recognized as a strike.

2. True negative implies regular data recognized as regular.

3. False positive implies regular input recognized as a strike.

4. False negative implies intrusion input recognized as regular.

Let's consider false positives as the subject. Here the regular data is tagged as suspicious activity. First, it needs to see how the information is believed as normal or an anomaly. It takes the input from the referenced dataset. The 1998 Defense Advanced Research Projects Agency Invasion Detection Assessment Project was arranged and overseen by Lincoln Labs-MIT. The point is to review and assess studies in intrusion detection. A standard arrangement of information to be inspected, which incorporates a vast assortment of invasions reproduced in a defense force organization climate, was given. The KDD intrusion detection challenge utilizes a rendition of this data file. So, in the wake of backtracking the information and by contrasting the ordinary classes and anomaly classes it presumes that it considers forty-one attributes to categories them into two classes, regular class, and abnormality(anomaly) class.

The attributes are: Attribute 1:duration, Attribute 2:protocol_type, Attribute 3:service, Attribute 4:flag, Attribute 5:src_bytes, Attribute 6:dst_bytes, Attribute 7:land, Attribute 8:wrong_fragment, Attribute 9:urgent, Attribute 10: hot, Attribute 11: num_failed_logins, Attribute 12: logged_in, Attribute 13: num_compromised, Attribute 14: root_shell, Attribute 15: su_attempted, Attribute 16: num_root, Attribute 17: num_file_creations, Attribute 18: num_shells, Attribute 19: num_access_files, Attribute 20: num_outbound_cmds, Attribute 21: is_host_login, Attribute 22: is_guest_login, Attribute 23: count, Attribute 24: srv_count, Attribute 25: serror_rate, Attribute 26: srv_serror_rate, Attribute 27: rerror_rate, Attribute 28: srv_rerror_rate, Attribute29: same_srv_rate, Attribute 30: diff_srv_rate, Attribute 31: srv_diff_host_rate, Attribute 32: dst_host_count, Attribute 33: dst_host_srv_count, Attribute 34: dst_host_same_srv_rate, Attribute 35: dst_host_diff_srv_ rate, Attribute 36: dst_host_same_src_ port_rate, Attribute 37: dst _ host _ srv _ diff_host_rate, Attribute 38: dst _host _serror _rate, Attribute 39: dst _ host _ srv _serror _rate, Attribute 40: dst_host_rerror_rate, Attribute 41: dst_host_srv_rerror_rate

Class: Class 1: normal, Class 2: anomaly

Now, by these 41 attributes, it will be concluded whether the input is normal or an anomaly. For instance: Let us take four value sets from the referenced dataset.

Table- II: Dataset Values

Attributes	**Set 1**	**Set 2**	**Set 3**	**Set 4**
Attribute 1	0	0	0	0
Attribute 2	udp	tcp	tcp	tcp
Attribute 3	other	http	finger	Private
Attribute 4	SF	SF	S0	S0
Attribute 5	146	232	0	0
Attribute 6	0	8153	0	0
Attribute 7	0	0	0	0
Attribute 8	0	0	0	0
Attribute 9	0	0	0	0
Attribute 10	0	0	0	0

Attribute 11	0	0	0	0
Attribute 12	0	1	0	0
Attribute 13	0	0	0	0
Attribute 14	0	0	0	0
Attribute 15	0	0	0	0
Attribute 16	0	0	0	0
Attribute 17	0	0	0	0
Attribute 18	0	0	0	0
Attribute 19	0	0	0	0
Attribute 20	0	0	0	0
Attribute 21	0	0	0	0
Attribute 22	0	0	0	0
Attribute 23	13	5	0	48
Attribute 24	1	5	24	16
Attribute 25	0.00	0.20	12	1.00
Attribute 26	0.00	0.20	1.00	1.00
Attribute 27	0.00	0.00	1.00	0.00
Attribute 28	0.00	0.00	0.00	0.00
Attribute 29	0.08	1.00	0.00	0.14
Attribute 30	0.15	0.00	0.50	0.06
Attribute 31	0.00	0.00	0.00	0.00
Attribute 32	255	255	255	255
Attribute 33	1	30	59	15
Attribute 34	0.00	1.00	0.23	0.06
Attribute 35	0.60	0.00	0.04	0.07
Attribute 36	0.88	0.03	0.00	0.00
Attribute 37	0.00	0.04	0.00	0.00
Attribute 38	0.00	0.03	1.00	1.00

Attribute 39	0.00	0.01	1.00	1.00
Attribute 40	0.00	0.00	0.00	0.00
Attribute 41	0.00	0.01	0.00	0.00
Class	Normal	Normal	Anomaly	Anomaly

The Intrusion detection system is working on 41 attributes to add up the anomalous behavior. There are several values of each attribute & if any of the entries is deviating from the mean value then it is considered an anomaly. There is a problem with false positives in the intrusion detection system. It should be discarding the count attribute to solve the false positive in IDS. The problem is that there is a need to find out the attack before time. The main concern is attributed as there are 41 attributes and it takes time to find the anomalous behavior. There is a need to improve efficiency by decreasing the number of attributes. The major components should be kept under menstruation while reshuffling attributes. So, instead of revolutionizing algorithms, we need to focus on attributes.Filtered Clusterer algorithm, is a class for running a self-assertive group on information that has been gone through a subjective channel. Filtering is the way toward eliminating extraordinary characters and accentuation that are not needed for giving the outcome.

IV. EXPERIMENTS AND RESULTS

There is a rise in the pace of false positives because of count attributes. To evaluate the system the interest is in two general signs of performance: the disclosure percentage and the bogus positive percentage. The false-positive percentage is characterized as the absolute count of the normal instance that was (mistakenly) named interruptions separated by the complete counts of regular objects. The detection rate is demonstrated as the count of invasion paradigm encountered by the mechanism split up by the overall count of invasion paradigms existent in the inspection set. These are pleasant pointers of exaction since they scale what rate of interruptions the framework can identify and how numerous erroneous collections it makes is the method. By calculating these values over the label data to measure performance.So, the problem is authentication. It can improve the authenticity by providing an OTP or one-time password to the user's Email address or contact number. Since the count attribute is of no use it can remove the count attributes by this. OTP is the best way, hence by using this the problem can be solved easily.

Algorithm 1: Signing Up

1. Start
2. Complete all the specified required inputs within the sign-up form, together with user_name, email_id & pswd.
3. If the user tries to submit the unfinished registration form Prompt "error message" in window
4. Else
   Registration success.
5. Exit.

Algorithm 2: Check-In

1. Start
2. Fill user_name & password and fill the CAPTCHA/I am not Robot
3. If user_name & password are matched in the database, then successfully logged in.
4. Else (for i=1 to i= 10)
   // (i is the no. of available try)
   Redo 1 to 2.
5. Produce one-time passkey (OTP) & send via email id or the provided contact number of the individual.
6. If One Time Passkey is matched, Redo steps one to four
7. Else
   Show "Wrong OTP".
8. Exit.

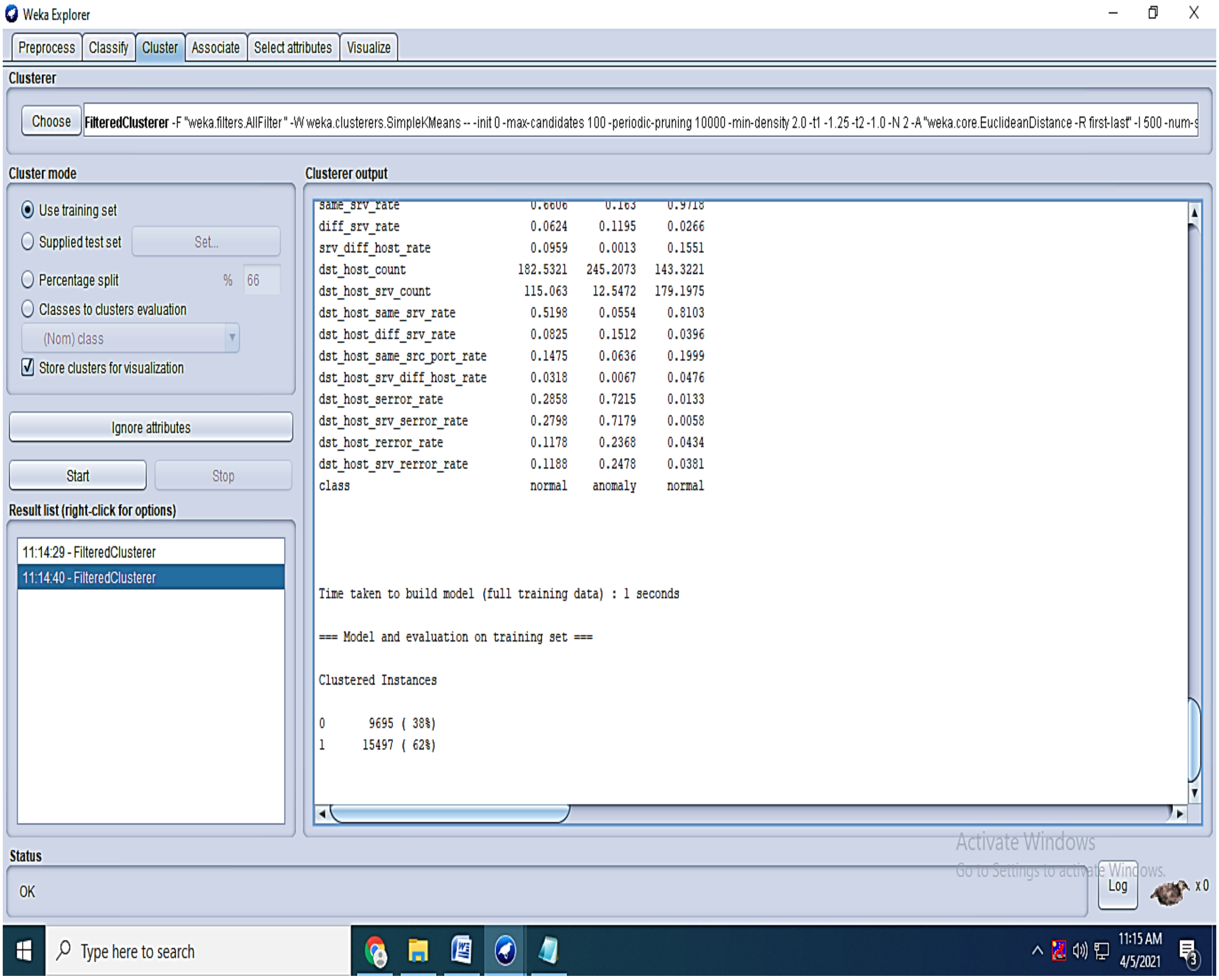
A filter is a peculiar subgroup of a partially ordered set. Let X be a topological space and x a point of X. A channel base B on X is said to cluster at x if and as it were in the event that each component of B has a nonempty crossing point with each neighborhood of x. In case a channel base B clusters at x and is better than a channel base C, at that point C clusters at x too.
Calculation
i. I. Each constraint of a channel base is likewise a bunching point of the base.
ii. A channel base B that has x as a group point may not join to x. In any case, there is a better channel base that does.
iii. For a channel base B, the set ∩{cl(B0): B0∈B} is the arrangement of all group points of B.
iv. The cutoff sub-par of B is the infimum of the arrangement of all bunch points of B.
v. The cutoff unrivaled of B is the supremum of the arrangement of all group points of B.
vi. B is a concurrent channel base if and just if its breaking point substandard and limit predominant concur; for this situation, the worth on which they concur is the restriction of the channel base.

Table- III: Shows the details of Final Cluster Centeroids with count attribute

Attribute	**Full Data (25192.0)**	Cluster #0 (9695.0)	1 (15497.0)
Attribute 1	305.0541	533.1584	162.3509
Attribute 2	Tcp	tcp	Tcp
Attribute 3	http	private	http
Attribute 4	SF	S0	SF
Attribute 5	24330.6282	39374.1009	14919.3572
Attribute 6	3491.8472	115.1045	5604.3541
Attribute 7	0	0	0
Attribute 8	0.0237	0.0175	0.0276
Attribute 8	0	0	0.0001
Attribute 10	0.198	0.0018	0.3208
Attribute 11	0.0012	0.0002	0.0018
Attribute 12	0	0	1
Attribute 13	0.2279	0	0.3704
Attribute 14	0.0015	0.0001	0.0025
Attribute 15	0.0013	0.0002	0.0021
Attribute 16	0.2498	0.0005	0.4058
Attribute 17	0.0147	0.001	0.0233
Attribute 18	0.0004	0	0.0006
Attribute 19	0.0043	0	0.007
Attribute 20	0	0	0
Attribute 21	0	0	0
Attribute 22	0	0	0
Attribute 23	84.5912	166.3895	33.4178
Attribute 24	27.6988	9.9234	38.8191
Attribute 25	0.2863	0.7253	0.0117
Attribute 26	0.2838	0.7212	0.0101
Attribute 27	0.1186	0.2467	0.0385
Attribute 28	0.1203	0.2486	0.04
Attribute 29	0.6606	0.163	0.9718
Attribute 30	0.0624	0.1195	0.0266
Attribute 31	0.0959	0.0013	0.1551
Attribute 32	182.5321	245.2073	143.3221
Attribute 33	115.063	12.5472	179.1975
Attribute 34	0.5198	0.0554	0.8103
Attribute 35	0.0825	0.1512	0.0396
Attribute 36	0.1475	0.0636	0.1999
Attribute 37	0.0318	0.0067	0.0476
Attribute 38	0.2858	0.7215	0.0133
Attribute 39	0.2798	0.7179	0.0058
Attribute 40	0.1178	0.2368	0.0434
Attribute 41	0.1188	0.2478	0.0381
Class	Normal	Anomaly	Normal
Computation Time: 1 seconds			

```
Clustered Instances
0      9695 (38%)
1     15497 (62%)
```

Figure 4.1 portrays the aftereffect framework utilizing a filtered clusterer algorithm with a count attribute. It can be seen that it takes 1 second to finish the clustering.



FilteredClusterer -F "weka.filters.AllFilter " -W weka.clusterers.SimpleKMeans -- -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 2 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-s

Clusterer output

```
same_srv_rate                  0.6606     0.163    0.9718
diff_srv_rate                  0.0624    0.1195    0.0266
srv_diff_host_rate             0.0959    0.0013    0.1551
dst_host_count               182.5321  245.2073  143.3221
dst_host_srv_count            115.063   12.5472  179.1975
dst_host_same_srv_rate         0.5198    0.0554    0.8103
dst_host_diff_srv_rate         0.0825    0.1512    0.0396
dst_host_same_src_port_rate    0.1475    0.0636    0.1999
dst_host_srv_diff_host_rate    0.0318    0.0067    0.0476
dst_host_serror_rate           0.2858    0.7215    0.0133
dst_host_srv_serror_rate       0.2798    0.7179    0.0058
dst_host_rerror_rate           0.1178    0.2368    0.0434
dst_host_srv_rerror_rate       0.1188    0.2478    0.0381
class                          normal   anomaly    normal



Time taken to build model (full training data) : 1 seconds

=== Model and evaluation on training set ===

Clustered Instances

0      9695 ( 38%)
1     15497 ( 62%)
```

Result list: 11:14:29 - FilteredClusterer; 11:14:40 - FilteredClusterer

Status: OK

Fig. 4.1 Experiment result using Filtered clusterer algorithm

Table- IV: Shows the details of Final Cluster Centeroids without count attribute.

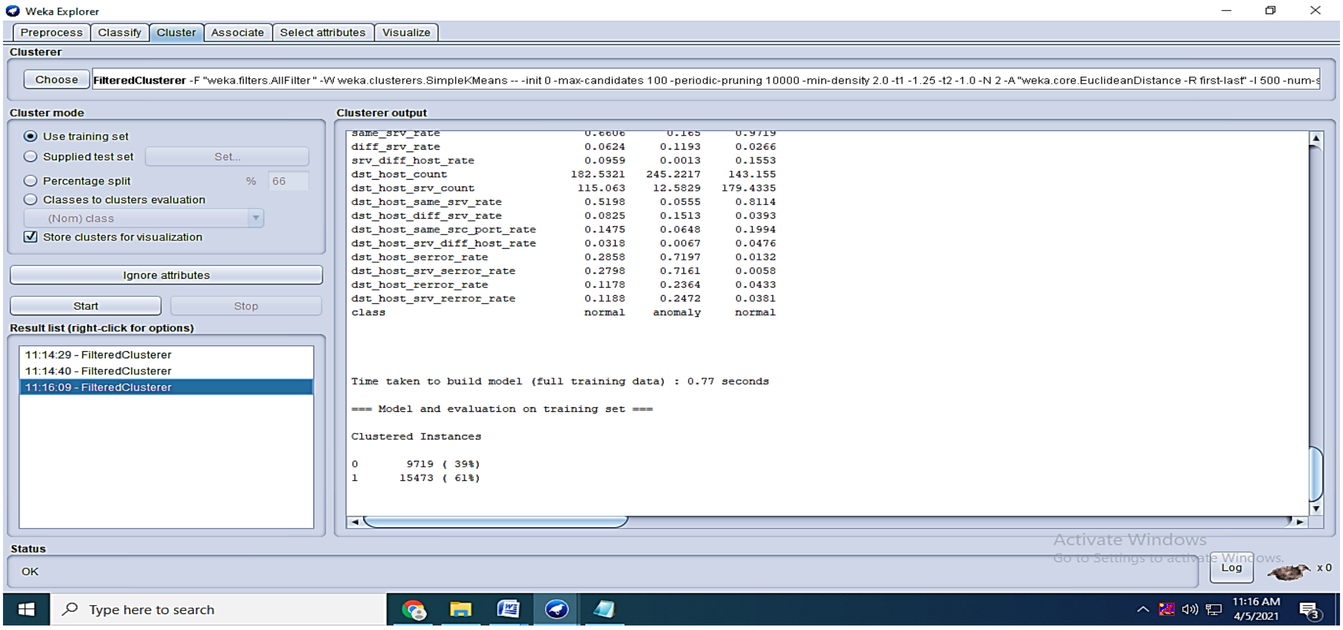
Attribute	**Full Data (25192.0)**	Cluster #0 (9719.0)	1 (15473.0)
Attribute 1	305.0541	533.1584	162.3509
Attribute 2	Tcp	tcp	tcp
Attribute 3	http	private	http
Attribute 4	SF	S0	SF
Attribute 5	24330.6282	39277.0561	14942.3821
Attribute 6	3491.8472	114.8202	5613.047
Attribute 7	0	0	0
Attribute 8	0.0237	0.021	0.0255
Attribute 8	0	0	0.0001
Attribute 10	0.198	0.0017	0.3213
Attribute 11	0.0012	0.0002	0.0018

Retrieval Number: 100.1/ijeat.D24130410421
DOI:10.35940/ijeat.D2413.0410421
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Attribute 12	0	0	1
Attribute 13	0.2279	0	0.371
Attribute 14	0.0015	0.0001	0.0025
Attribute 15	0.0013	0.0002	0.0021
Attribute 16	0.2498	0.0005	0.4064
Attribute 17	0.0147	0.001	0.0233
Attribute 18	0.0004	0	0.0006
Attribute 19	0.0043	0	0.007
Attribute 20	0	0	0
Attribute 21	0	0	0
Attribute 22	0	0	0
Attribute 24	27.6988	9.9546	38.8443
Attribute 25	0.2863	0.7236	0.0117
Attribute 26	0.2838	0.7194	0.0101
Attribute 27	0.1186	0.2461	0.0386
Attribute 28	0.1203	0.248	0.04
Attribute 29	0.6606	0.165	0.9719
Attribute 30	0.0624	0.1193	0.0266
Attribute 31	0.0959	0.0013	0.1553
Attribute 32	182.5321	245.2217	143.155
Attribute 33	115.063	12.5829	179.4335
Attribute 34	0.5198	0.0555	0.8114
Attribute 35	0.0825	0.1513	0.0393
Attribute 36	0.1475	0.0648	0.1994
Attribute 37	0.0318	0.0067	0.0476
Attribute 38	0.2858	0.7197	0.0132
Attribute 39	0.2798	0.7161	0.0058
Attribute 40	0.1178	0.2364	0.0433
Attribute 41	0.1188	0.2472	0.0381
Class	Normal	Anomaly	Normal
Computation Time: 0.77 seconds			
Clustered Instances 0 9719 (39%) 1 15473 (61%)			

Figure 4.2 portrays the aftereffect of utilizing the filtered clusterer algorithm without the count attribute. It can be seen that it takes 0.77 seconds to finish the clustering.



FilteredClusterer -F "weka.filters.AllFilter " -W weka.clusterers.SimpleKMeans -- -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 2 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-s

● Use training set
○ Supplied test set
○ Percentage split % 66
○ Classes to clusters evaluation
(Nom) class
☑ Store clusters for visualization

11:14:29 - FilteredClusterer
11:14:40 - FilteredClusterer
11:16:09 - FilteredClusterer

```
same_srv_rate                  0.6606     0.165     0.9719
diff_srv_rate                  0.0624     0.1193    0.0266
srv_diff_host_rate             0.0959     0.0013    0.1553
dst_host_count               182.5321   245.2217   143.155
dst_host_srv_count           115.063     12.5829   179.4335
dst_host_same_srv_rate         0.5198     0.0555    0.8114
dst_host_diff_srv_rate         0.0825     0.1513    0.0393
dst_host_same_src_port_rate    0.1475     0.0648    0.1994
dst_host_srv_diff_host_rate    0.0318     0.0067    0.0476
dst_host_serror_rate           0.2858     0.7197    0.0132
dst_host_srv_serror_rate       0.2798     0.7161    0.0058
dst_host_rerror_rate           0.1178     0.2364    0.0433
dst_host_srv_rerror_rate       0.1188     0.2472    0.0381
class                          normal    anomaly    normal




Time taken to build model (full training data) : 0.77 seconds

=== Model and evaluation on training set ===

Clustered Instances

0       9719 ( 39%)
1      15473 ( 61%)
```

OK

Fig. 4.2 Experiment result using filtered clusterer algorithm

Table- V: Portrays the comparison of results of Filtered Clusterer Algorithm with and without count attribute.

Algorithm	Time taken with count attribute	Time taken without count attribute
Filtered Clusterer	1 second	0.77 seconds

V. CONCLUSION

In the current situation, such countless individuals have experienced a ton of these when they need to open an account on the web or in web banking, and furthermore, in light of having more accounts, it is hard to oversee such countless passwords in the buffer. Within the occasion of experiencing three off-base endeavors, they are blocked by that bank's location for another 24 hours. In this paper, the course of action is given for the particular issue. So, if this arrangement is trailed by framework the issue of false-positive can be decreased. By eliminating the count attribute, it can see that the presentation of algorithms is improving in a decent way. While looking at the columns of table 4.1 it can obviously think that the exhibition of algorithms is being improved and the OTP authentication system prevails.

REFERENCES

1. V. Chandola, A. Banerjee, V. Kumar, “Anomaly detection as a survey” ACM Comput. Surv.41(3) (2009)15:1–15:58.
2. Francesco Mercaldo, “Identification of anomalies in processes of database alteration” IEEE 2013.
3. Dorothy E. Denning. “An Intrusion- Detection Model” 1986 IEEE Computer Society Symposium on Research in Security and Privacy, pp 118-31.
4. S. K. Chaturvedi1, Prof. Vineet R., Prof. Nirupama T. “Anomaly Detection in Network using Data mining Techniques” International Journal ISSN 2250-2459 Volume 2, Issue 5, May 2012.
5. UgoFiore, Francesco, Aniello “Network anomaly detection with the restricted Boltzmann machine” Neurocomputing 122 (2013) 13–23.
6. T. Bhavani et al., “Data Mining for Security Applications,” Proceedings of the 2008 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing Volume 02, IEEE Computer Society, 2008.
7. Bhavani Thuraisingham, “Data Mining for Malicious Code Detection and Security Applications” 2009 IEEE/WIC/ACM 2009.
8. Shih-Wei Lina, Kuo-Ching Yingb, Chou-Yuan Leec, Zne-Jung Leed “An intelligent algorithm with feature selection and decision rules applied to anomaly detection” Elsevier 2011.
9. Bharat singh, Nidhi Kushwaha, and OP Vyas “Exploiting Anomaly Detections for high Dimensional data using Descriptive Approach of Data mining” IEEE(ICCT) 2013.
10. Shu Wu, Member, and Shengrui Wang “Information-Theoretic Outlier Detection for Large-Scale Categorical Data” VOL. 25, NO. 3, MARCH 2013.
11. Kapil Wankhade, Mrudula Gudadhe, Prakash Prasad, “A New Data Mining Based Network Intrusion Detection Model”, In Proceedings of ICCCT 2010, IEEE, 2010, pp.731-735.
12. M. Xue, C. Zhu, "Applied Research on Data Mining Algorithm in Network Intrusion Detection," jcai, pp.275-277, 2009 International Joint Conference Artificial Intelligence, 2009.
13. Abdul Samad bin Haji Ismail “A Novel Method for Unsupervised Anomaly Detection using Unlabeled Data” IEEE 2008.
14. Jonathan J, Davis, Andrew J. Clark “Data preprocessing for anomaly-based network intrusion detection: A review” Elsevier 2011.
15. S. Gnanapriya, R. Adline Freeda, M. Sowmiya “Evaluation of Clustering Capability Using Weka Tool” International Journal of Innovations in Engineering and Technology (IJIET) 2017.

AUTHORS PROFILE

Pratik Jain, Completed his Master of Engineering from IPS Academy Institute of Engineering and science. Publications are as follows:

1) Design and Implementation of Binary Neural Network Classification Learning Algorithm in IOSR, 2012.

2) IDENTIFYING THE PROBLEM & SOLUTION OF FALSE POSITIVE in Proceedings of Eleventh IRF International Conference, 2014
3) "Eliminating the Attribute Count from Intrusion Detection System to Reduce the Problem of False Positive in the Network" in IRJET, 2015.
4) Improving the Performance of Intrusion Detection System by Removing the Count Attribute from KDD Cup 1999 Data in IJSRD, 2017
5) Improving the Efficiency of KDD Cup 1999 Data for Intrusion Detection Using K-Means Algorithm by Removing the Count Attribute IJSRD, 2020.
6) Comparing the Result of KDD Cup 1999 Data by using K-mean Algorithm and Make Density based Cluster in Intrusion Detection System by Removing the Count Attribute in IJCA, 2020
7) Improving the efficiency of KDD cup 1999 data by using Make Density Based Clusterer algorithm in Intrusion Detection system by removing the count attribute in JETIR, 2020.

Membership information is as follows:

1) International Economics Development Research Center (IEDRC), Membership ID: **90081028**
2) **International Computer Science and Engineering Society (ICSES)**

Ravikant Kholwal, is a final year undergraduate student pursuing Computer Science and Engineering at the Indian Institute of Information Technology, Design, and Manufacturing, Jabalpur. He has developed several projects in the field of Android Development One of his projects is Smart Parking. In this app, Users register their accounts and can book parking slots before reaching their respective destination. They can also pay for their Parking slot using UPI. The frontend part is developed using XML and Java, and in the backend, Firebase is implemented. When the car comes to the parking slot, the sensor sends information to the NodeMCU which in turn sends information to the Firebase database.

Tavneet Singh Khurana, is a third-year Computer Science and Engineering student pursuing his undergraduate degree from IPS Academy, Institute of Engineering and Science (Indore). His area of interest includes web development, full-stack development, and machine learning. He has done an online certification course in frontend UI frameworks and has created some mobile-friendly static websites. He is currently working on terrorism detection project, which aims to detect terrorism related activities, that will report and detect terrorism related posts on a social media platform. Tavneet plans to enhance in his skills in web technologies and explore more into the vast spectrum of machine learning.