VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Title of the Project: INTRUSION DETECTION USING DEEP LEARNING(LSTM)

PROF:SUMAIYA THASEEN

NAME:RAHUL ANAND

REG NO:18BIT0436

❖ Technique/algorithm used and why it was chosen (motivation)

PAPER:1.

•We utilize the LSTM organization to hold long haul conditions among separated highlights and evade the angle disappearing issue.

•We propose to utilize the drop-interface regularization method on the covered up to-concealed weight grids inside the LSTM to keep away from the overfitting issue.

•We enhance the model's hyper-boundaries dependent on experimentation. The proposed half and half model is assessed utilizing the open UNSW-NB15 dataset for interruption location and afterward contrasted and cutting edge models utilizing the UNSW-NB15 dataset and an extra dataset of interruption recognition, ISCX2012.

PAPER:2.

•We proposed an assault discovery strategy utilizing IDS, which depends on Spark ML and the Conv-LSTM network. It is a novel mixture approach, which consolidates both profound and shallow learning ways to deal with abuse their qualities and conquer diagnostic overheads.

•We assessed our IDS on the ISCX-UNB dataset and examined the parcel catch record (pcap) with Spark; prior scientists didn't consider or assess crude bundle datasets.

•We contrast our half and half IDS and best in class IDS frameworks dependent on customary ML.

•The recreation results show that our IDS can distinguish network abuses precisely in 97.29% of cases and outflanks cutting edge approaches during 10-overlap cross-approval tests.

•Our proposed IDS outflanks existing methodologies as well as accomplish mass adaptability while seriously lessening the preparation time, generally speaking giving a further extent of exactness with a low likelihood of bogus alerts.

PAPER:3.

Interruption Detection System (IDS) is one of the significant issues in network security. IDSs are worked to distinguish both known and obscure pernicious assaults. A few AI calculations are utilized broadly in IDS, for example, neural organization, SVM, KNN and so on. In any case, these calculations have still a few restrictions, for example, high bogus positive and bogus caution rate. In this paper, our commitment is to manufacture a classifier of IDS following profound learning approach. We locate the most appropriate enhancer among six streamlines for Long Short-Term Memory Recurrent Neural Network (LSTM RNN) model are utilized to IDS. Through our analyses, we found that LSTM RNN model with Nadam analyzer beats to past works. We exhibit our methodology is truly effectiveness to interruption identification with precision is 97.54%, recognition rate is 98.95%, and the bogus caution rate is sensible with 9.98%.

PAPER:4.

Our principle commitments for planning interruption recognition frameworks as portrayed in this paper have two sections: the presentation of a framework call language displaying approach and another gathering method.To the best of the creators' information, our strategy is the first to present the idea of a language model, particularly utilizing LSTM, to irregularity based IDS. The framework call language model can catch the semantic significance of each call and its connection to other framework calls. Also, we proposed an inventive and basic group technique that can more readily fit to IDS plan by zeroing in on bringing down bogus alert rates. We indicated its exceptional exhibition by contrasting it and existing condition of-theart strategies and showed its power and over-simplification by investigates different benchmarks.

❖ Architecture/ model/pseudocode developed

PAPER:1.

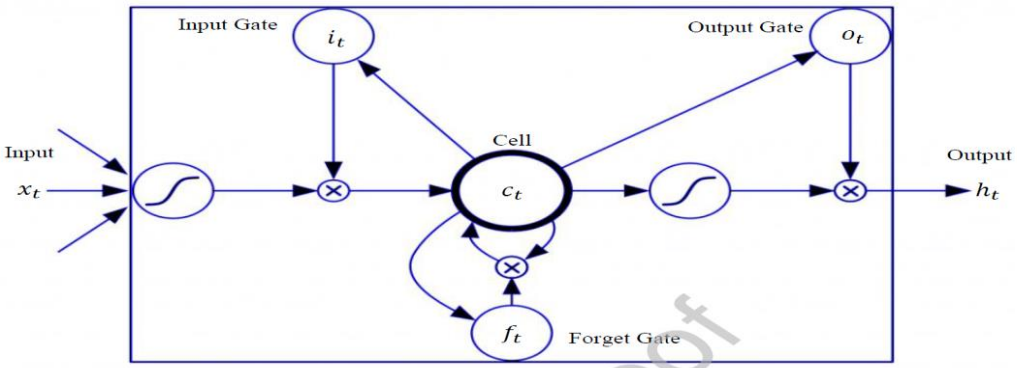
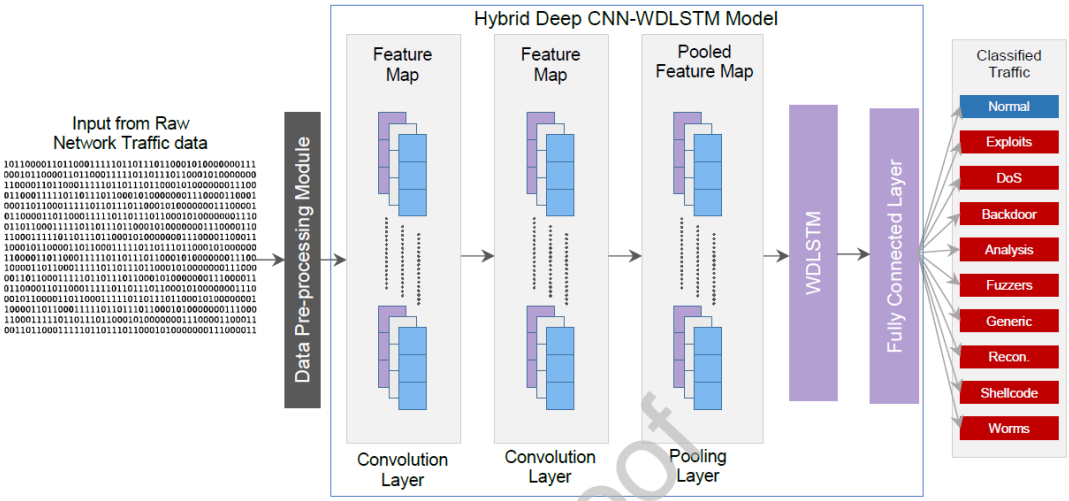


Fig. 2. A common structure of LSTM.



PAPER:2.

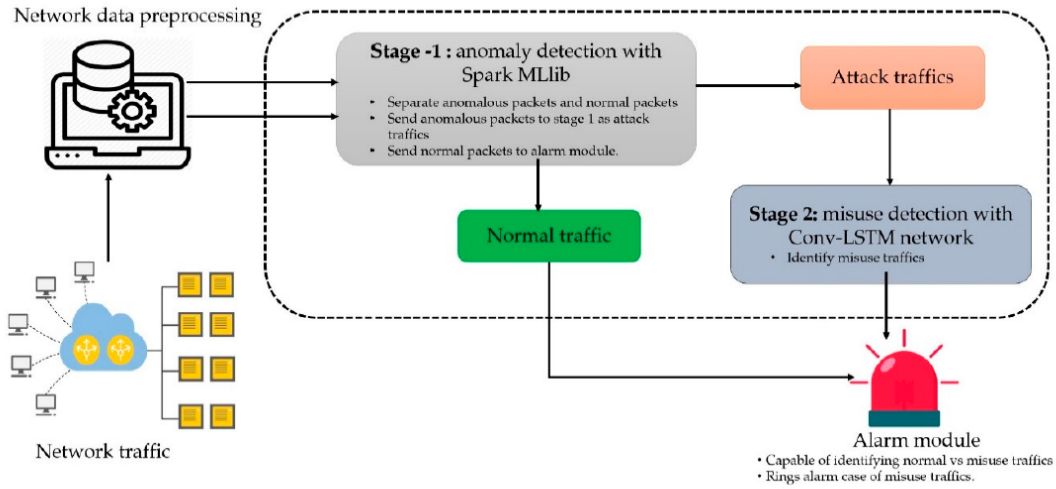
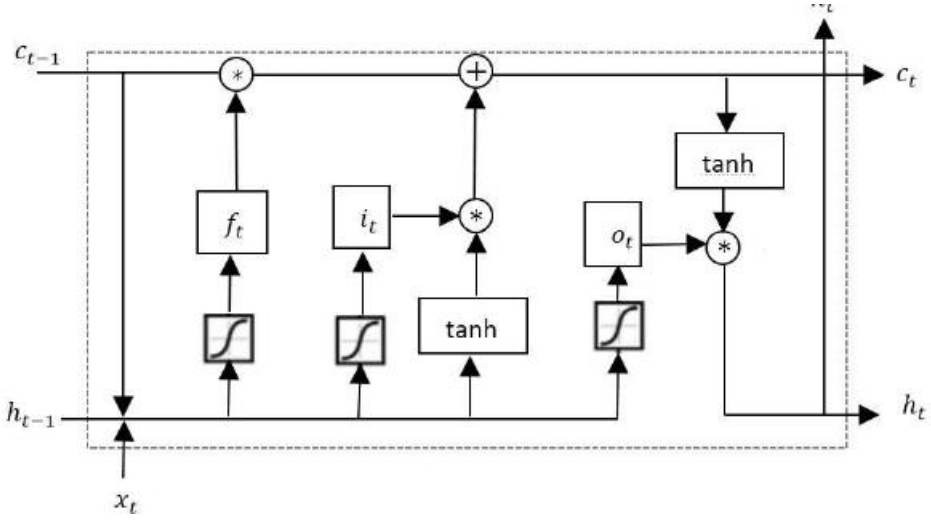


Figure 1. An overview of the proposed ID model.

PAPER:3.

Intermittent Neural Network (RNN) is a famous model in Deep Learning field. This model has applied effectively to perceive creating picture and text; machine decipher with superior. Notwithstanding, RNN has the inconvenience. It can't catch long haul reliance. It causes disappearing angle plunge. Consequently, Hochrieter and Schiemidbuher proposed Long Momentary Memory (LSTM), which can learn long haul conditions. LSTM is intended to defeat disappearing inclination drop since it keeps away from longterm reliance issue. To recall data forlong timeframes, we supplant every regular shrouded hub by LSTM cell. LSTM cell is shown in Fig.



PAPER:4.

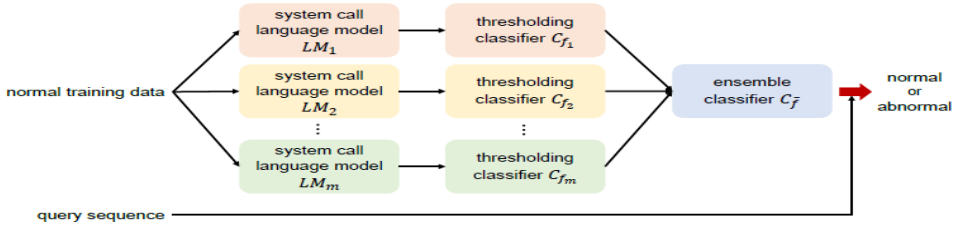
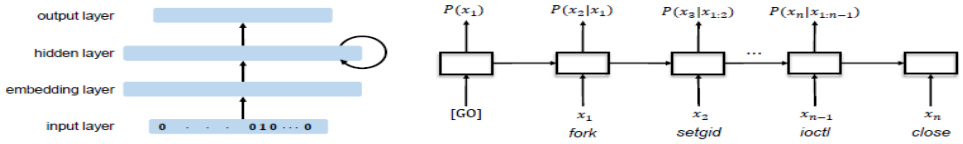


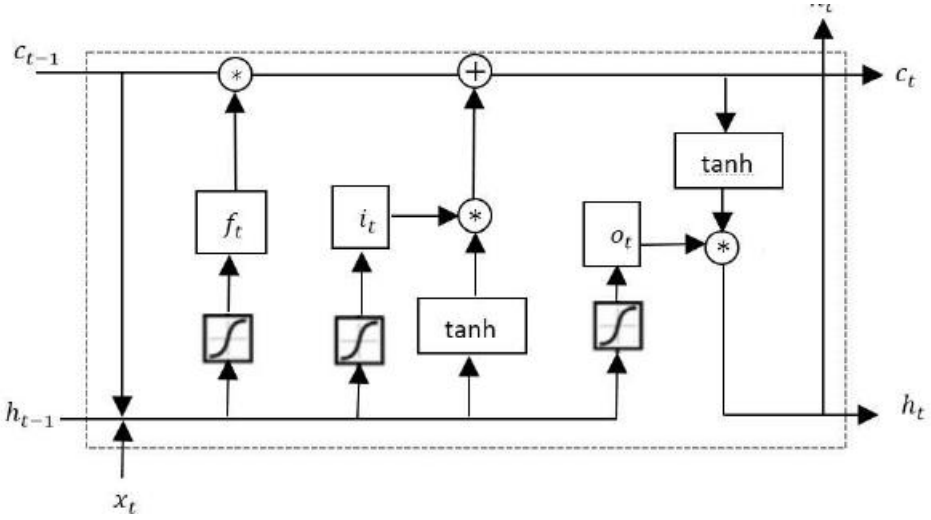
Figure 1: Overview of the proposed method.



(a) language model architecture (b) estimation of sequence probability

Figure 2: System-call language model.

PAPER:5.



❖ Datasets analyzed in the paper with the performance results

PAPER:1.

Class Name	#Instances	Distribution Percentage
Normal	1,958,467	87.94%
Exploits	33,422	1.50%
DoS	11,716	0.53%
Backdoor	1,959	0.09%
Analysis	2,069	0.09%
Fuzzers	19,578	0.88%
Generic	187,598	8.42%
Reconnaissance	10,871	0.49%
Shellcode	1,187	0.05%
Worms	134	0.01%
Total	2,227,001	100%

Table 2. Distribution of training and testing sets

CLASS NAME	Training set (70%)	Testing set (30%)
Normal	1370880	587587
Exploits	23275	10147
DoS	8281	3435
Backdoor	1371	588
Analysis	1440	629
Fuzzers	13712	5866
Generic	131435	56163
Reconnaissance	7608	3263
Shellcode	811	376
Worms	87	47
Total	1558900	668101

Table 3. Confusion Matrix of Normal and Abnormal Classification

	Normal	Abnormal	Total
Normal	**583,129**	4,458	587,587
Abnormal	14,307	**66,207**	80,514
Total	597,436	70,665	668,101

Table 4. Confusion Matrix of All Types of Attacks Classification.

	Normal	Exploits	DoS	Backdoor	Analysis	Fuzzers	Generic	Reconnaissance	Shellcode	Worms	**Total**
Normal	**586,579**	53	8	2	14	928	0	1	2	0	587,587
Exploits	146	**8,074**	1,351	10	17	241	79	197	31	1	10,147
DoS	29	2,270	**965**	7	5	87	40	14	17	1	3,435
Backdoor	0	331	137	**38**	6	74	0	0	2	0	588
Analysis	87	257	132	3	**61**	76	13	0	0	0	629
Fuzzers	1,637	429	146	6	16	**3,625**	3	3	1	0	5,866
Generic	15	466	163	2	3	22	**55,478**	5	8	1	56,163
Reconnaissance	9	535	182	1	0	5	3	**2,527**	0	1	3,263
Shellcode	6	44	5	1	0	4	6	0	**310**	0	376
Worms	0	36	0	0	0	3	1	0	0	**7**	47
Total	588,508	12,495	3,089	70	122	5,065	55,623	2,747	371	11	668,101

Table 5. Performance Metrics of Normal and Abnormal Classification

	Precision	Recall	F1-score	Accuracy
Normal	0.98	0.99	0.98	
Abnormal	0.94	0.82	0.88	**97.17%**
Weighted avg.	0.97	0.97	0.97	

Table 6. Performance Metrics of Normal and Other Types of Attacks Classification

	Precision	Recall	F-score	Accuracy
Normal	1	1	1	
Exploits	0.64	0.8	0.71	
DoS	0.32	0.27	0.29	
Backdoor	0.5	0.07	0.12	
Analysis	0.44	0.09	0.15	
Fuzzers	0.71	0.61	0.66	
Generic	1	0.99	0.99	**98,43%**
Reconnaissance	0.93	0.77	0.84	
Shellcode	0.82	0.79	0.81	
Worms	0.5	0.09	0.15	
Weighted Avg.	0.98	0.98	0.98	

PAPER:2.

Reference	Approach	Accuracy	Dataset
Yin et al. [15]	RNN IDS	90%	NSL-KDD
Reddy et al. [17]	SVM	99.95%	KDD99
B. Inger et al. [31]	ANN	99.67%	KDD99
Tsiropoulou et al. [42]	IMRA game theory	90.0%	Passive RFID
N. Gao et al. [33]	DBN	93.49%	NSL-KDD
Ghanem et al. [43]	Metaheuristic	96.4%	NSL-KDD
Sabhnani et al. [44]	MLP	97.0%	KDD99
Ying Chung et al. [45]	SSO	93.0%	KDD99
Kakavand et al. [25]	Ada boost + DT	97.0%	ISCX 2012
Kumar et al. [26]	PCA	94.05%	ISCX 2012
Yassin et al. [27]	AMGA2-NB	98.8%	ISCX 2012
Tan et al. [29]	MCA + EMD	90.12%	ISCX 2012
Sallay et al. [30]	PLL + NGL	95.30%	ISCX 2012

Table 2. Summary of the ISCX-IDS 2012 dataset (daily traffic).

Days	Date	Description	Size (GB)
Friday	11 June 2010	Normal, hence no malicious activity	16.1
Saturday	12 June 2010	Infiltrating the network from inside and normal activity	4.22
Sunday	13 June 2010	Infiltrating the network from inside and normal activity	3.95
Monday	14 June 2010	HTTP denial of service and normal activity	6.85
Tuesday	15 June 2010	Distributed denial of service using an IRC Botnet	23.04
Wednesday	16 June 2010	Normal, hence no malicious activity	17.6
Thursday	17 June 2010	Brute force SSH and normal activity	12.3

Table 3. Distribution of ISCX-IDS 2012 training and testing dataset.

Dataset/Network Flows	#Feature	Training		Testing	
ISCX-UNB Saturday	8	85,222	1353	45,889	1353
ISCX-UNB Monday	8	108,945	2451	58,664	1320
ISCX-UNB Tuesday	8	347,308	24,295	187,012	13,083
ISCX-UNB Wednesday	8	339,470	0	182,793	0
ISCX-UNB Thursday	8	255,054	3381	137,338	1822

Note: For both set: left—Benign, right—Malicious.

Table 4. Distribution of the data for the second stage classifier.

Input	#Features	Attack Category
Training set	8	HTTP DoS, DDoS, and Botnet
Test set	8	Brute force SSH, HTTP DoS, DDoS, Botnet, and Brute force SSH

PAPER:3.

Attack type	List of relevance features
DoS	back, land, neptune, pod, smurt, teardrop, apache2, back, land, mailbomb, Neptune, podm processtable, smurf
Probe	ipsweep, nmap, portsweep, satan, mscan, saint
U2R	ftpwrite, guespasswd, imap, multihop, phf, spy,warezclient, Warezmaster, httptunnel, named,sendmail, snmpgetattack, xlock, xsnoop
R2L	bufferiverflow, loadmodule, perl, rootkit, ps, snmpguess, sqlattack, worm, xterm

TABLE 4
The average classification performance of each attack

Optimizer	DoS	Probe	R2L	U2R	Normal
RMSprop	0.9672	0.59131	0.51041	0.5	0.89945
Adagrad	0.98272	0.63132	0.56719	0	0.91213
Adadelta	0.98131	0.5819	0.52	0	0.81508
Adam	0.98288	0.60313	0.51401	0	0.90597
Adamax	0.98362	0.68257	0.66236	0.5	0.94464
Nadam	**0.98435**	**0.77034**	**0.66702**	**0.5**	**0.95716**

TABLE 5
The average classification performance

Optimizer	Accuracy	Recall	FAR	Precision	Efficiency
RMSprop	0.9496	0.97709	0.15361	0.95987	7.0753
Adagrad	0.96786	0.99142	0.12277	0.96887	8.5766
Adadelta	0.94401	0.99074	0.23128	0.9422	5.40848
Adam	0.95915	0.98723	0.1287	0.9675	8.20554
Adamax	0.97376	0.98621	0.10642	0.97595	9.30781
Nadam	**0.9754**	**0.9895**	**0.0998**	**0.9769**	**9.9808**

PAPER:4.

Table 1: Summary of datasets used for experiments

	Normal		Attack	
Benchmark	# training	# validation	# type	# attack
ADFA-LD	833	4372	6	746
KDD98	1364	5459	10	41
UNM-lpr	627	3136	1	2002

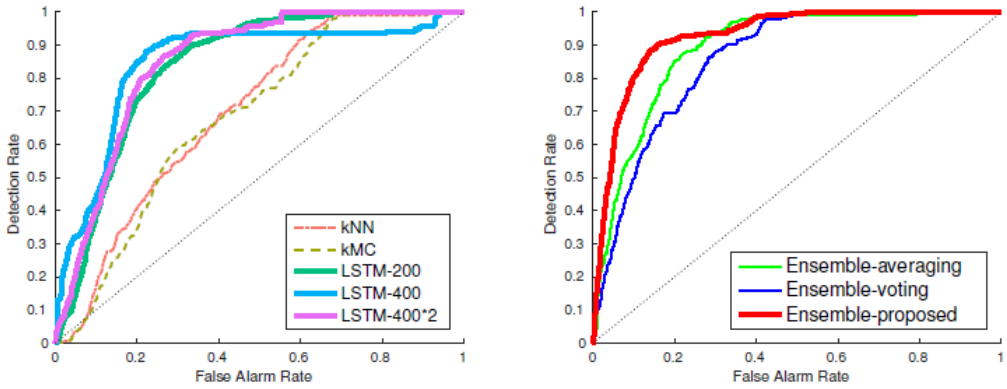


Figure 3: ROC curves from the ADFA-LD. Left shows the result from our three system-call language models with different parameters and two baseline classifiers. Right illustrates the results from different ensemble methods.

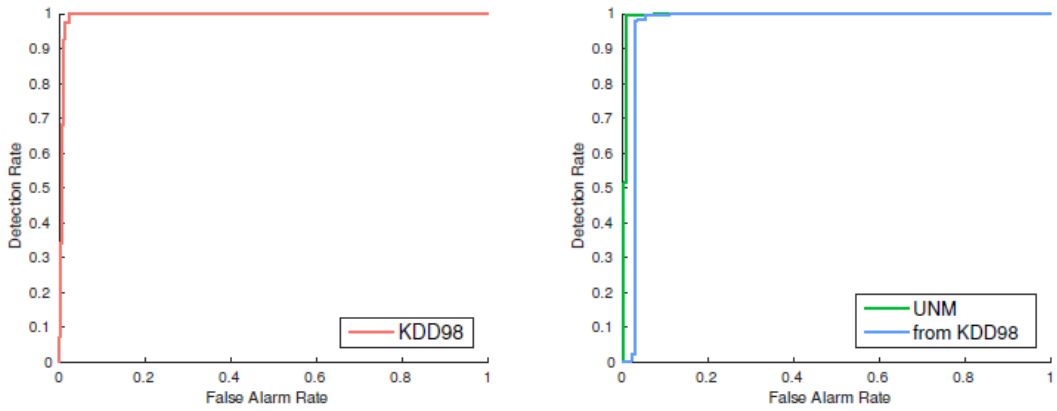


Figure 4: ROC curves from the KDD dataset and UNM dataset. Left is the evaluation about the KDD dataset. Right is the evaluation about UNM dataset using the model trained with the KDD98 dataset and the UNM dataset.

PAPER:5.

KDD Cup 1999 dataset has been used to measure a performance of IDS in many researches. Although the dataset is old, it is good to compare the IDS models. Because there are lots of performance measurement results with the same dataset. That is the main reason why we chooesed KDD Cup
1999 dataset.

TABLE 1
Category of the attacks

Category	Attacks
DoS	back, land, neptune, pod, smurf, teardrop
R2L	ftp-write, guess-passwd, imap, multihop, phf, spy, warezclient, warezmaster
U2R	buffer-overflow, loadmodule, perl, rootkit
Probe	ipsweep, nmap, portsweep, satan

TABLE 2
Efficiency by the learning rate

	0.0001	0.001	0.01	0.1
DR	0.998	0.984	0.877	0.979
FAR	0.817	0.512	0.413	0.479
Efficiency	**1.222**	**1.921**	**2.124**	**2.045**

TABLE 3
Efficiency by the hidden layer size

Size	DR	FAR	Efficiency
10	0.68	0.849	**0.800942285**
20	0.949	0.497	**1.90945674**
30	0.966	0.202	**4.782178218**
40	0.984	0.311	**3.163987138**
50	0.995	0.293	**3.395904437**
60	0.951	0.282	**3.372340426**
70	0.951	0.233	**4.081545064**
80	0.993	0.133	**7.466165414**
90	0.985	0.227	**4.339207048**

TABLE 4
Result summary

	DR	FAR	Efficiency
Best	0.989583	0.07781	**12.692786**
Worst	0.984807	0.129093	**7.663612**
Average	0.9879003	0.1003805	**10.005282**

❖ Any comparison done with the previous techniques to specify that the proposed method is superior

PAPER:1.

Model	Dataset	Accuracy (%)
Basic CNN [33]	ISCX2012	94.26%
Inception CNN [33]	ISCX2012	94.74%
LSTM [33]	ISCX2012	93.72%
GRU [33]	ISCX2012	94.30%
TSDL model [22]	UNSW-NB15	89.134%
Proposed CNN-WDLSTM model	UNSW-NB15	**97.17%**

Table 8. Average execution time of detection for one instance in milliseconds.

Model	Average execution time
TSDL model [22]	0.003372
CNN-WDLSTM Model	**0.002383**

Table 9. Accuracy results of detecting normal and abnormal traffic for TSDL and CNN-WDLSTM models traffic using 10-fold cross validation.

Model	Correctly Classified Instances	Incorrectly Classified Instances	Accuracy (%)
TSDL [22]	212,007	24,316	89.711
CNN-WDLSTM	229,173	7,150	**96.975**

PAPER:2.

Classifier	Precision	Recall	F1-Score	FAR	DR	Stage
SVM	0.6835	0.6515	0.6786	15.27	0.65	1
DT	0.7930	0.8012	0.7965	11.29	0.82	1
GBT	0.8529	0.8632	0.8612	8.13	0.85	1
RF	0.8919	0.8875	0.8845	5.72	0.89	1
Conv-LSTM	0.9725	0.9750	0.9729	0.71	0.97	2

The most significant boost that we experienced is with the Conv-LSTM network, which manages to accurately detect misuse in up to 97% of cases. The superior feature extraction of CNN and
long-term dependencies between non-linear features is the reason behind this significant performance
improvement. Implementation details are given in Supplementary Materials.

PAPER:3.

TABLE 6
Comparison to other IDS classifiers

Classifier	Precision	DR	Accuracy	FAR
FNN [17]	92.47	86.89	97.35	2.65
GNNN [17]	87.08	59.12	93.05	12.46
RBNN [17]	69.56	59.12	93.05	12.46
Jordan ANN [17]	-	62.9	-	37.09
RNN with Hessian-free [7]	-	95.37	-	2.1
LSTM RNN with SGD [8]	-	98.88	96.93	10.04
Our classifier	**97.69**	**98.95**	**97.54**	**9.98**

PAPER:4.

As examined before, the proposed technique additionally has phenomenal transportability. Rather than elective strategies, our proposed technique causes noteworthy littler preparing overhead since it doesn't have to manufacture information bases or word references to keep a possibly exponential measure of examples. Our technique is reduced and light in that the size of the space needed to spare boundaries is little. The general preparing and surmising measures are likewise proficient and quick, as our strategies can be actualized utilizing productive successive network duplications.

PAPER:5.

TABLE 5
Comparison with other algorithms

	DR(%)	FAR(%)	Accuracy(%)
GRNN	59.12	12.46	87.54
PNN	96.33	3.34	96.66
RBNN	69.83	6.95	93.05
KNN	45.74	46.49	90.74
SVM	87.65	6.12	90.4
Bayesian	77.6	17.57	88.46
LSTM-RNN	**98.88**	**10.04**	**96.93**