Machine Learning for Classifying Anomalies and Intrusions in Communication Networks

Ph.D. Defense

Zhida Li

School of Engineering Science
Simon Fraser University, Vancouver
British Columbia, Canada

Roadmap

- Introduction
- Network anomalies and intrusions
- Feature selection and dimension reduction
- Applications of machine learning algorithms
- Variable features broad learning systems
- BGPGuard: BGP anomaly detection tool
- Conclusions and future work
- References

Roadmap

- Introduction:
  - background and motivation
  - summary of research contributions
  - research publications
- Network anomalies and intrusions
- Feature selection and dimension reduction
- Applications of machine learning algorithms
- Variable features broad learning systems
- BGPGuard: BGP anomaly detection tool
- Conclusions and future work
- References

Background and motivation

- The Internet is highly susceptible to failures and attacks
- Various machine learning models have been implemented to enhance cybersecurity
- Using machine learning techniques to detect network intrusions is an important topic in cybersecurity

Machine learning techniques

- A variety of network-based intrusion detection systems (NIDSs) have been designed using:
  - supervised, unsupervised, and semi-supervised learning
- They help detect the malicious intentions of network users
- Detection of attacks:
  - require updating or retraining generated models to capture deviations from regular network activities
- Training time:
  - important for the decision-making process

Summary of research contributions

- Three main contributions:
  - implementation and comparison of various machine learning algorithms
  - development of new machine learning algorithms
  - development of an anomaly detection tool named BGPGuard

Research publications

I have co-authored 2 book chapters, 1 journal paper, and 10 conference publications. Additional publications are in preparation.

Book chapters:

- Q. Ding, Z. Li, S. Haeri, and Lj. Trajković, "Application of machine learning techniques to detecting anomalies in communication networks: Datasets and Feature Selection Algorithms" in *Cyber Threat Intelligence,* M. Conti, A. Dehghantanha, and T. Dargahi, Eds., Berlin: Springer, pp. 47–70, 2018.
- Z. Li, Q. Ding, S. Haeri, and Lj. Trajković, "Application of machine learning techniques to detecting anomalies in communication networks: Classification Algorithms" in *Cyber Threat Intelligence,* M. Conti, A. Dehghantanha, and T. Dargahi, Eds., Berlin: Springer, pp. 71–92, 2018.

Research publications

Journal publications:

- Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting power outage and ransomware using BGP routing records," *IEEE Commun. Mag.*, to be submitted.
- Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Research publications

Conference publications

- Z. Li, A. L. Gonzalez Rios, and Lj. Trajković, “Classifying denial of service attacks using fast machine learning algorithms,” in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Melbourne, Australia, Oct. 2021, pp. 1221-1226 (virtual).
- Z. L, A. L. Gonzalez Rios, and Lj. Trajković, “Detecting Internet worms, ransomware, and blackouts using recurrent neural networks,” in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Toronto, Canada, Oct. 2020, pp. 2165-2172 (virtual).
- A. L. Gonzalez Rios, Z. L, K. Bekshentayeva, and Lj. Trajković, “Detection of denial of service attacks in communication networks,” in *Proc. IEEE Int. Symp. Circuits and Systems*, Seville, Spain, Oct. 2020 (virtual).
- Z. L, A. L. Gonzalez Rios, G. Xu, and Lj. Trajković, “Machine learning techniques for classifying network anomalies and intrusions,” in *Proc. IEEE Int. Symp. Circuits and Systems*, Sapporo, Japan, May 2019 (virtual).
- A. L. Gonzalez Rios, Z. L, G. Xu, A. Dias Alonso, and Lj. Trajković, “Detecting network anomalies and intrusions in communication networks,” in *Proc. 23rd IEEE International Conference on Intelligent Engineering Systems 2019*, Gödöllő, Hungary, Apr. 2019, pp. 29–34.
- Z. L, P. Batta, and Lj. Trajković, “Comparison of machine learning algorithms for detection of network intrusions,” in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Miyazaki, Japan, Oct. 2018, pp. 4248–4253.
- P. Batta, M. Singh, Z. L, Q. Ding, and Lj. Trajković, “Evaluation of support vector machine kernels for detecting network anomalies,” in *Proc. IEEE Int. Symp. Circuits and Systems*, Florence, Italy, May 2018, pp. 1-4.
- Q. Ding, Z. L, P. Batta, and Lj. Trajković, “Detecting BGP anomalies using machine learning techniques,” in *Proc. IEEE International Conference on Systems, Man, and Cybernetics,* Budapest, Hungary, Oct. 2016, pp. 3352–3355.

Research publications

Conference publications (virtual network embedding)

- H. Ben Yedder, Q. Ding, U. Zakia, Z. Li, S. Haeri, and Lj. Trajkovic, "Comparison of virtualization algorithms and topologies for data center networks," in *Proc. The 26th Int. Conf. Compt. Comm. Netw., 2nd Workshop Netw. Secur. Analytics Automat.*, Vancouver, Canada, Aug. 2017.
- S. Haeri, Q. Ding, Z. Li, and Lj. Trajkovic, "Global resource capacity algorithm with path splitting for virtual network embedding," in *Proc. IEEE Int. Symp. Circuits Syst.*, Montreal, Canada, May 2016, pp. 666–669.

Roadmap

- Introduction
- Network anomalies and intrusions
- Feature selection and dimension reduction
- Applications of machine learning algorithms
- Variable features broad learning systems
- BGPGuard: BGP anomaly detection tool
- Conclusions and future work
- References

Network traffic datasets

- Anomalies affect performance of the Internet Border Gateway Protocol (BGP)
- Réseaux IP Européens (RIPE) and Route Views:
  - Slammer（2003）, Nimda (2001), Code Red (2001)
  - Moscow blackout (2005), Pakistan power outage (2021)
  - WannaCrypt (2017), WestRock (2021)
- NSL-KDD (an improvement of the KDD'99 dataset)
- Canadian Institute for Cybersecurity (CIC) collections: CICIDS2017, CSE-CIC-IDS2018, CICDDoS2019
- BCNET

BGP anomalies: Internet worms

- Slammer (2003):
  - infected Microsoft SQL servers through a small piece of code that generated IP addresses at random
- Nimda (2001):
  - exploited vulnerabilities in the Microsoft Internet Information Services (IIS) web servers for Internet Explorer 5
- Code Red (2001):
  - attacked Microsoft IIS web servers by replicating itself through the IIS server weaknesses

BGP anomalies: power outages

- Moscow blackout (2005):
  - caused a complete shutdown of the Chagino substation of the Moscow energy ring
  - caused the failure of the Internet traffic exchange
- Pakistan power outage (2021):
  - caused by a cascading effect after an abrupt frequency drop in the power transmission system of the Guddu power plant
  - decreased network connectivity levels in Pakistan to 62% within the first hour and to 52% after six hours

BGP anomalies: ransomware attacks

- WannaCrypt (2017):
  - malicious attackers encrypted data files
  - ransom was requested
- WestRock (2021):
  - impacted the company's information and operational technology systems for over six days
  - caused delays in shipments and production levels

Network traffic datasets

BGP datasets:

- Anomalous data: days of the attack
- Regular data: two days prior and two days after the attack
- 37 numerical features from BGP update messages

BGP and CIC datasets:

- Training and test datasets are created based on the percentages of anomalous data:
  - training: 80%, 70%, 60%
  - test: 20%, 30%, 40%

BGP datasets: Internet worms

- Slammer, Nimda, Code Red:

Collection site	Dataset	Regular (min)	Anomaly (min)	Regular (training)	Anomaly (training)	Regular (test)	Anomaly (test)	Start	End
RIPE	Slammer	6,331	869	3,210	530	3,121	339	23.01.2003 00:00:00	27.01.2003 23:59:59
	Nimda	7,308	1,301	3,673	827	3,635	474	16.09.2001 00:00:00	21.09.2001 23:59:59
	Code Red	6,880	320	4,000	200	2,880	120	17.07.2001 00:00:00	21.07.2001 23:59:59
Route Views	Slammer	6,319	869	3,198	530	3,121	339	23.01.2003 00:00:00	27.01.2003 23:59:59

Route Views data collection began in 2003.

BGP datasets: power blackouts and outages

- Moscow blackout and Pakistan power outage:

Collection site	Dataset	Regular (min)	Anomaly (min)	Regular (training)	Anomaly (training)	Regular (test)	Anomaly (test)	Start	End
RIPE	Moscow blackout	6,960	240	3,120	180	3,840	60	23.05.2005 00:00:00	27.05.2005 23:59:59
	Pakistan power outage	6,880	320	4,000	200	2,880	120	07.01.2021 00:00:00	11.01.2021 23:59:59
Route Views	Moscow blackout	6,865	130	3,075	85	3,790	45	23.05.2005 00:00:00	27.05.2005 23:59:59
	Pakistan power outage	6,880	320	4,000	200	2,880	120	07.01.2021 00:00:00	11.01.2021 23:59:59

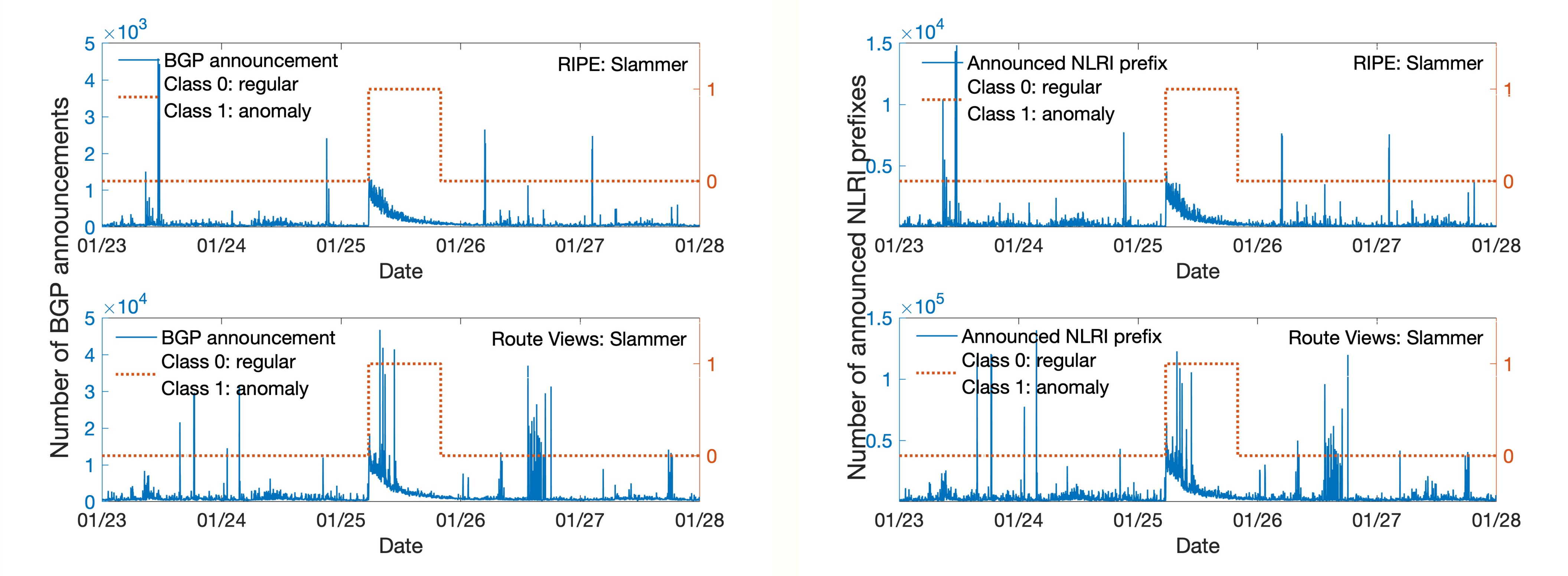
BGP datasets: ransomware attacks

- WannaCrypt and WestRock ransomware attacks:

Collection site	Dataset	Regular (min)	Anomaly (min)	Regular (training)	Anomaly (training)	Regular (test)	Anomaly (test)	Start	End
RIPE/ Route Views	WannaCrypt	5,760	5,760	2,880	3,420	2,880	2,340	10.05.2017 00:00:00	17.05.2017 23:59:59
	WestRock ransomware	5,832	10,008	2,952	6,008	2,880	4,000	21.01.2021 00:00:00	31.01.2021 23:59:59

BGP dataset: Slammer (2003)

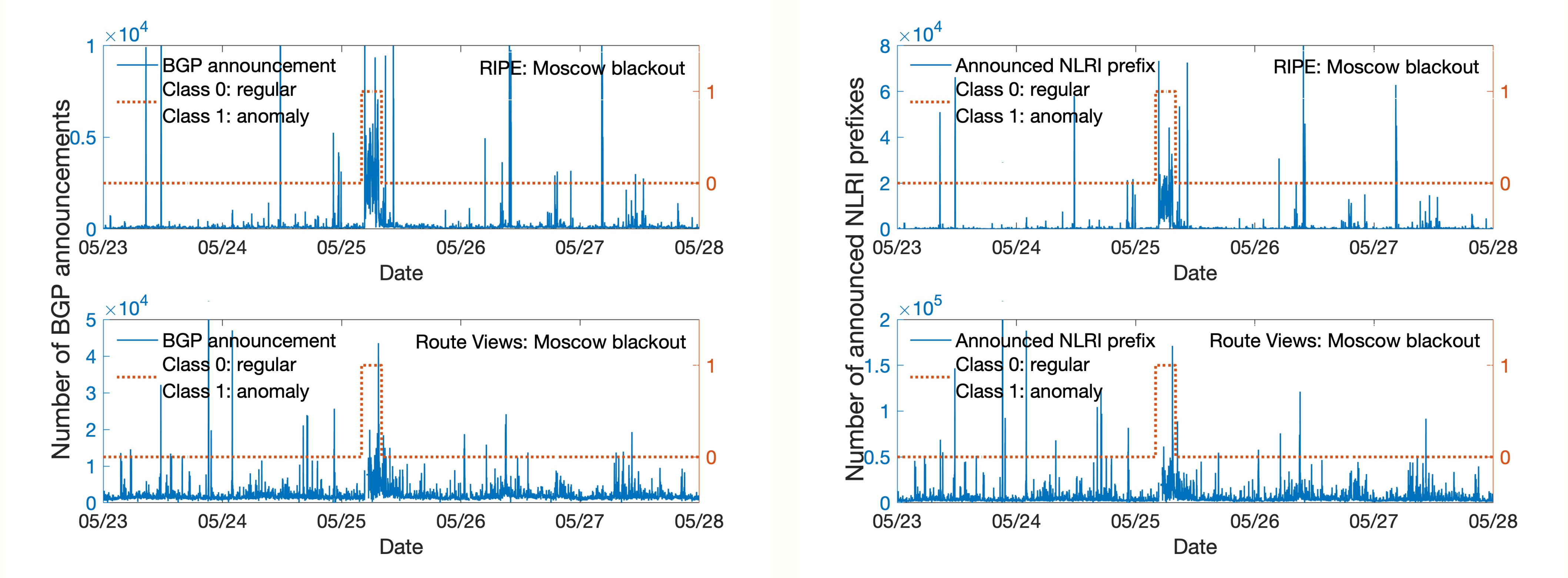
- Number of BGP announcements and announced NLRI prefixes vs. date:



BGP: Border Gateway Protocol
NLRI: Network Layer Reachability Information

BGP dataset: Moscow blackout (2005)

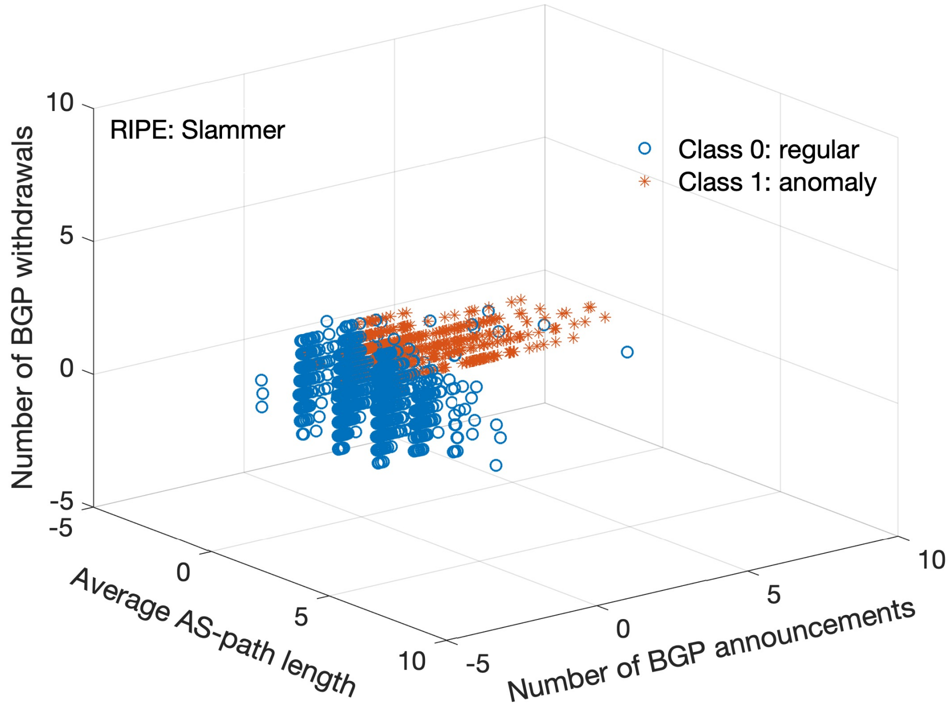
- Number of BGP announcements and announced NLRI prefixes vs. date:

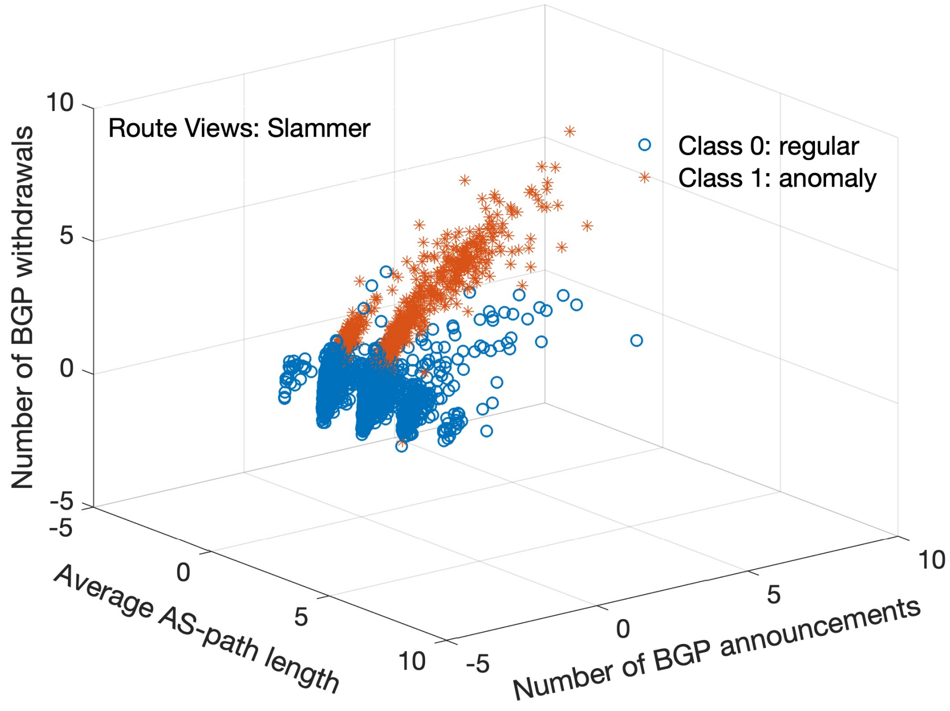


BGP: Border Gateway Protocol
NLRI: Network Layer Reachability Information

BGP dataset: Slammer

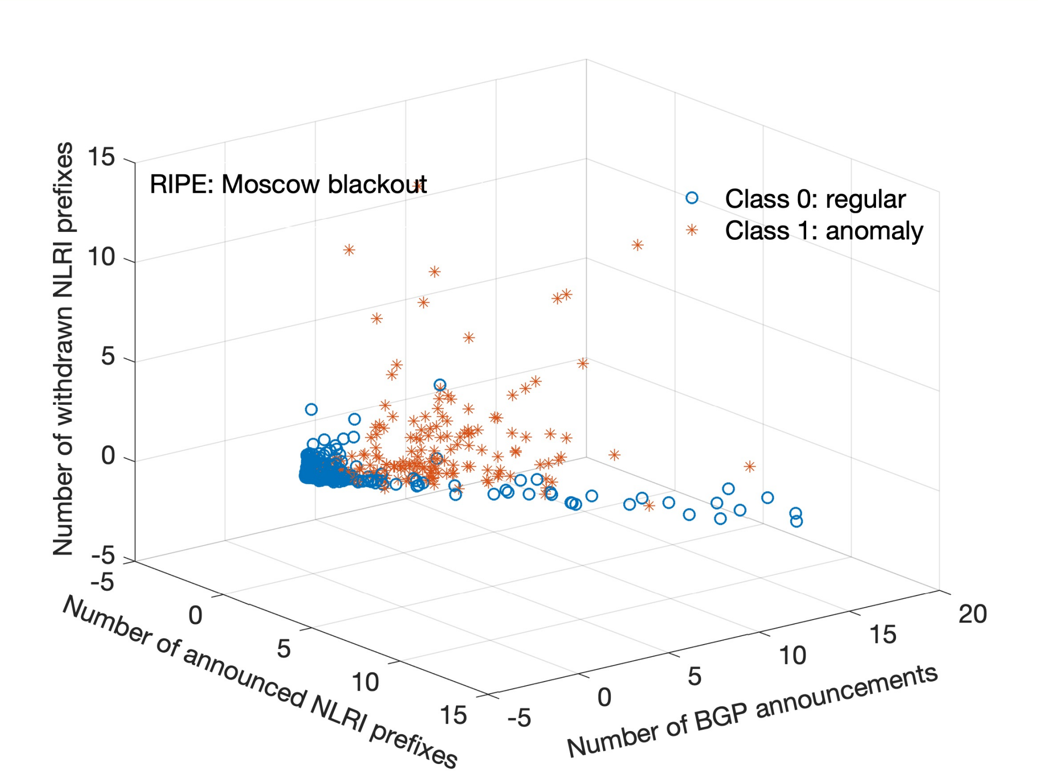
- Average AS-path length vs. number of BGP announcements vs. number of BGP withdrawals:

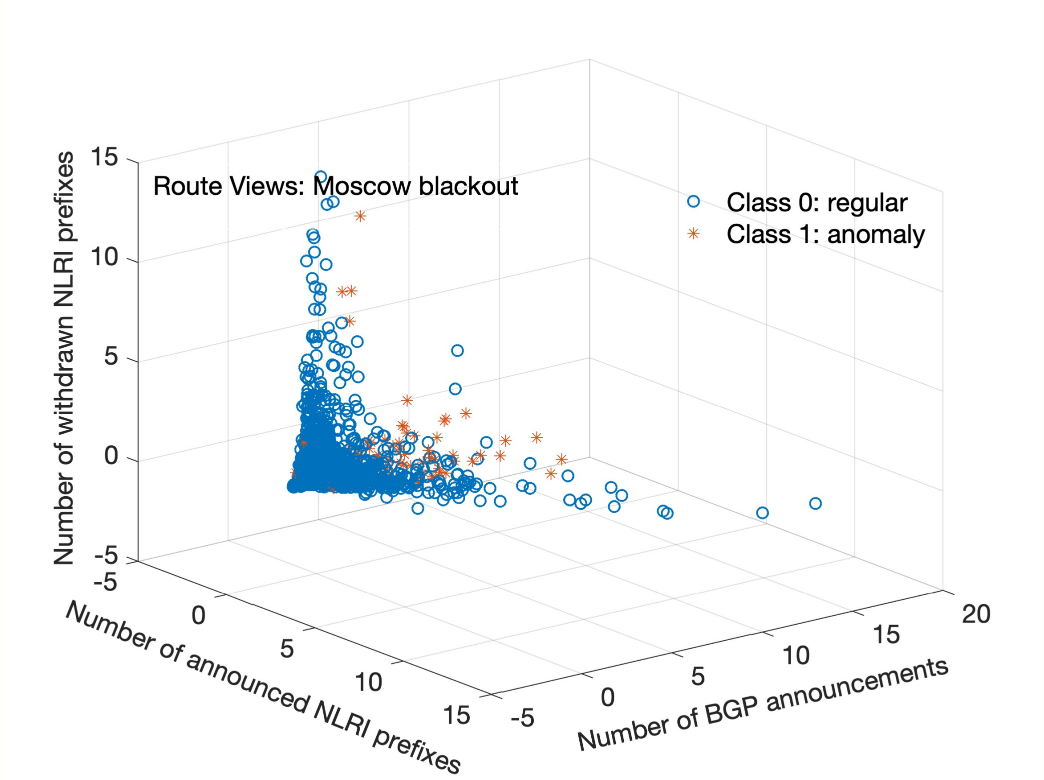




BGP dataset: Moscow blackout

- Number of announced NLRI prefixes vs. number of BGP announcements vs. number of withdrawn NLRI prefixes:





NSL-KDD datasets

- NSL-KDD dataset: an improvement of the KDD’99 dataset that was used in various intrusion detection systems
- NSL-KDD dataset: a benchmark used for evaluating anomaly detection and intrusion techniques

	Regular	DoS	U2R	R2L	Probe	Total
$KDDTrain^{+}$	67,343	45,927	52	995	11,656	125,973
$KDDTest^{+}$	9,711	7,458	200	2,754	2,421	22,544
$KDDTest^{-21}$	2,152	4,342	200	2,754	2,402	11,850

Canadian Institute for Cybersecurity datasets

CICIDS2017, CSE-CIC-IDS2018, and CICDDoS2019:

- Testbed used to create the publicly available dataset that includes multiple types of recent cyber attacks
- Dataset features: extracted from collected TCP and UDP network flows with a network traffic flow analyzer
- Each dataset: over 80 features including destination IP and port, protocol type, flow duration, and maximum/minimum packet size
- Network traffic collected:
    - Monday, 03.07.2017 to Friday, 07.07.2017
    - Wednesday, 14.02.2018 to Friday, 02.03.2018
    - Saturday, 03.11.2018 and Saturday, 01.12.2018

CIC datasets: DoS and DDoS attacks

- Application-layer DoS and TCP/UDP DDoS attacks

Dataset	Attack	Number of Data Points
	GoldenEye	10,293
CCIDS2017	Hulk	230,124
July 05, 2017	SlowHTTPTest	5,499
	Slowloris	5,796
CSE-CIC-IDS2018	GoldenEye	41,508
February 15, 2018	Slowloris	10,990
CICDDoS2019	Domain Name System	5,071,011
December 01, 2018	Lightweight Directory Access Protocol	2,179,930
	Network Time Protocol	1,202,642

Roadmap

- Introduction
- Network anomalies and intrusions
- Feature selection and dimension reduction
- Applications of machine learning algorithms
- Variable features broad learning systems
- BGPGuard: BGP anomaly detection tool
- Conclusions and future work
- References

Feature selection

- Using feature selection algorithms to select the most relevant features in the original dataset often improves classification performance
- Various feature selection algorithms are used to reduce redundancies by ranking and identifying the most relevant features:
  - Fisher
  - minimum redundancy maximum relevance (mRMR)
  - mutual information base (MIBASE)
  - odds ratio (OR)
  - decision trees
  - extremely randomized trees (extra-trees)

Feature selections: extra-trees

- The Gini importance is used to compute feature scores in a given dataset:

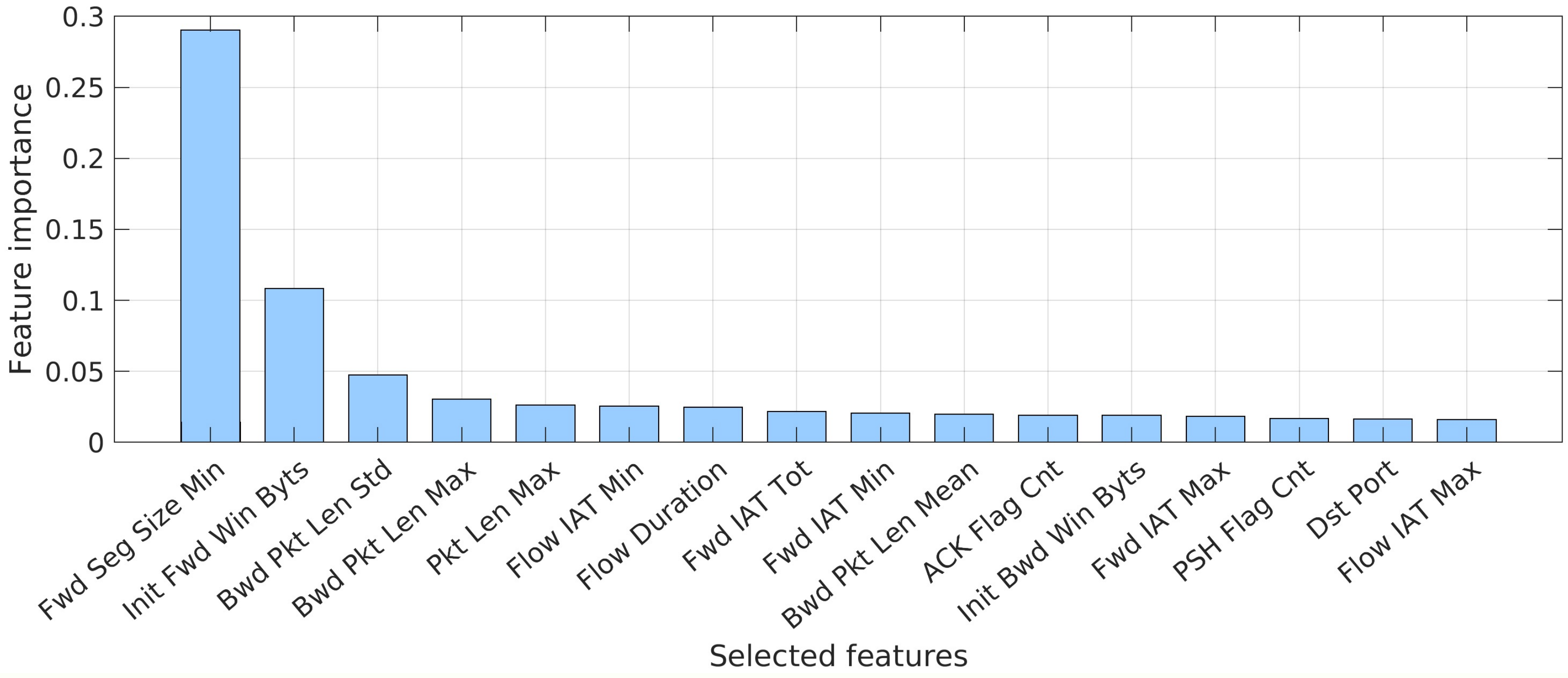
$$Importance(\boldsymbol{X}_c) = \frac{1}{N_T}\sum_{T}\sum_{t\in T:v(s_t)=\boldsymbol{X}_c} p(t)\Delta i(s_t, t)\,,$$

where:

- $\boldsymbol{X}_c$ is the subset of $\boldsymbol{X}$ corresponding to one feature
- N_T is the number of trees
- t is the index of a node in a tree
- s_t is the direction of the split
- $v(s_t)$ is a randomly generated threshold
- $p(t)$ is the weight
- $\Delta i(st, t)$ is the decrease of the node impurity equivalent to its importance

Most relevant features

- CSE-CIC-IDS2018: 16 most relevant features

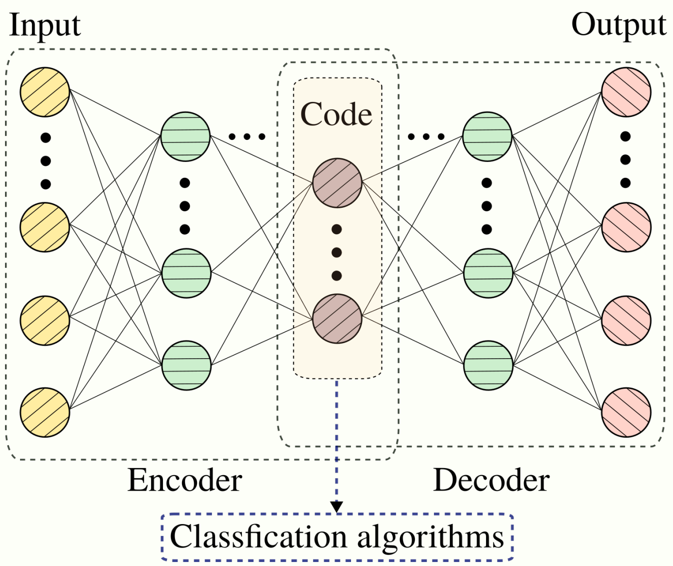


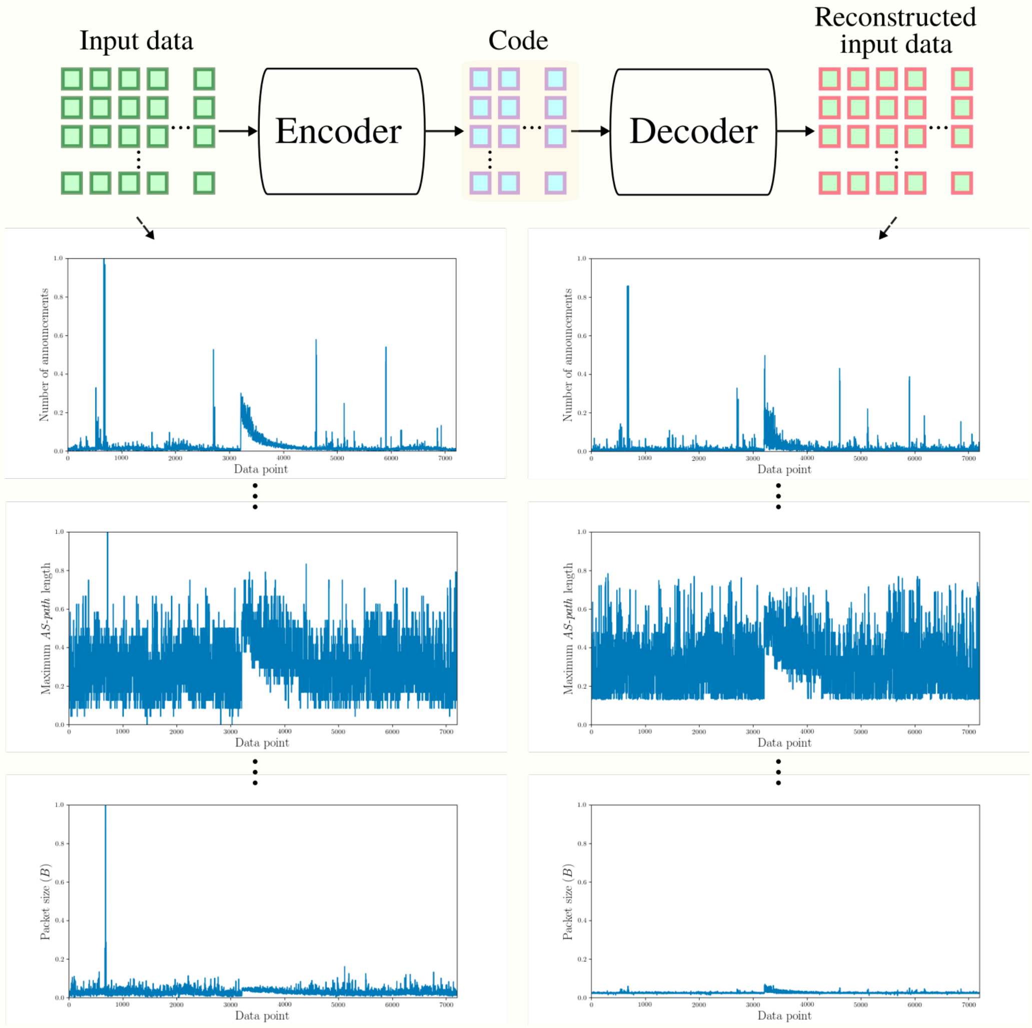
Dimension reduction

- Dimension reduction (unsupervised learning):
  - uses unlabeled input data to train a model
- Its goal is to transform original data into the lower dimensional data that preserve characteristics of the original data:
  - autoencoders: unsupervised neural networks used to learn a representation from a given dataset
- Deep autoencoder with various LSTM/GRU hidden layers was used for dimension reduction

Autoencoders

- High-level training process:





Roadmap

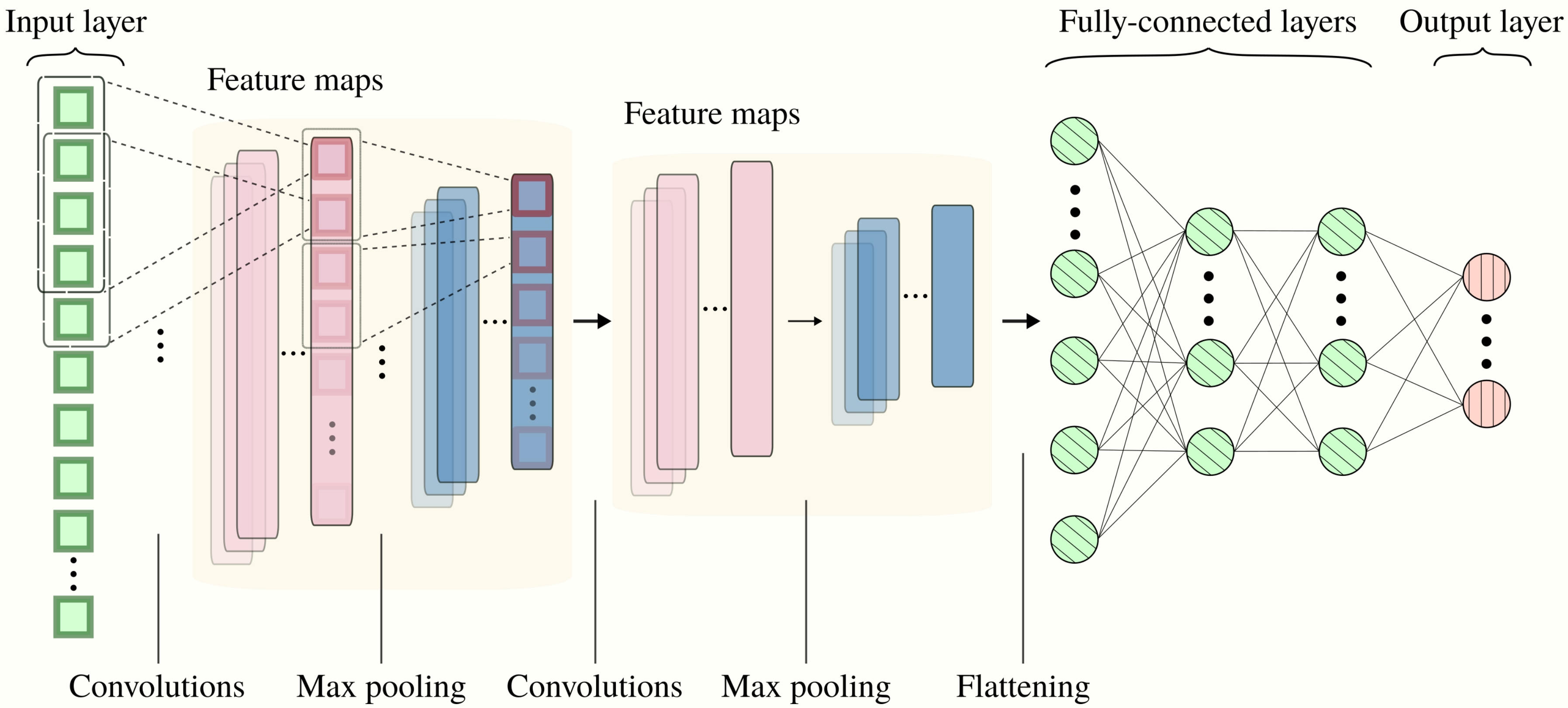
- Introduction
- Network anomalies and intrusions
- Feature selection and dimension reduction
- Applications of machine learning algorithms:
  - traditional machine learning
  - deep learning
  - fast machine learning
- Variable features broad learning systems
- BGPGuard: BGP anomaly detection tool
- Conclusions and future work
- References

Machine learning algorithms

- Network intrusion detection systems employ algorithms:
  - traditional machine learning:
    - support vector machine (SVM), naïve Bayes, decision tree, hidden Markov model (HMM), extreme learning machine (ELM)
  - deep learning:
    - convolutional neural networks (CNNs)
    - recurrent neural networks (RNNs)
    - autoencoders
  - fast machine learning:
    - broad learning system (BLS) and its extensions
    - gradient boosting decision trees (GBDT)

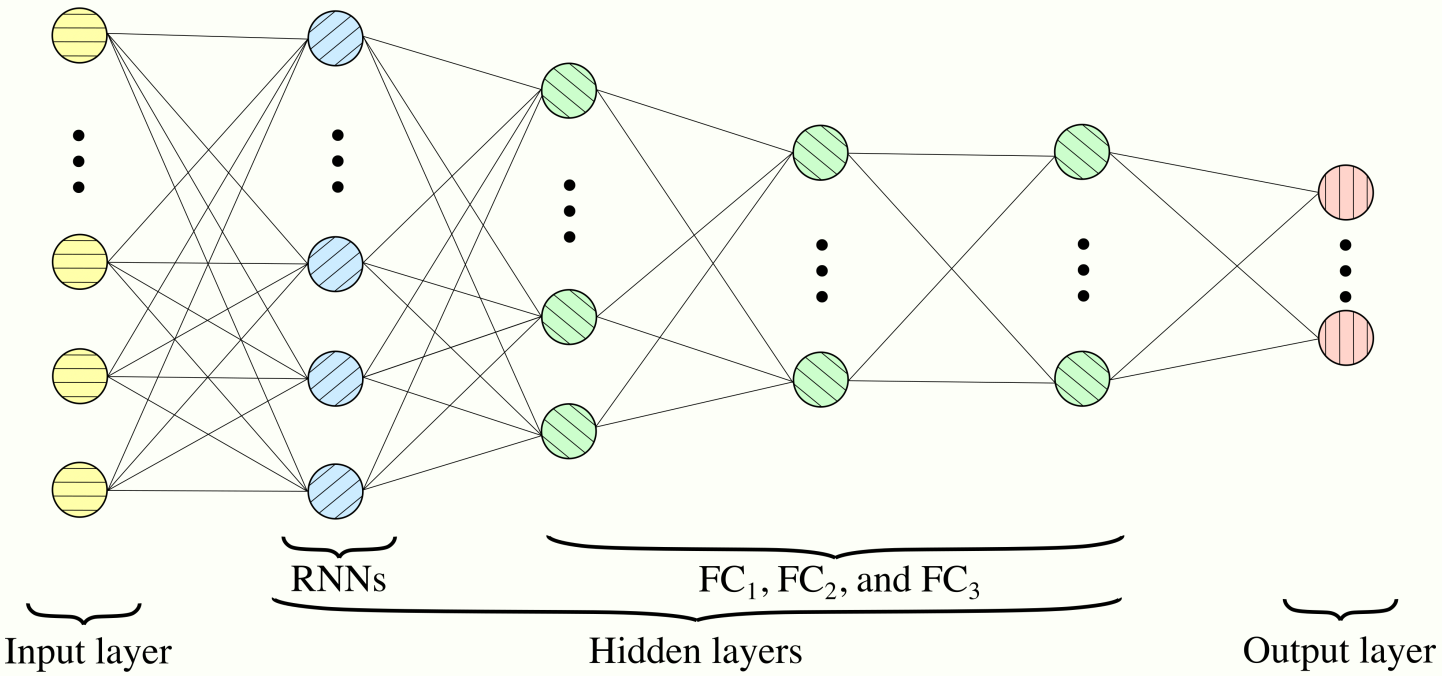
Convolutional neural network

- High-level structure of a CNN using one-dimensional input data:



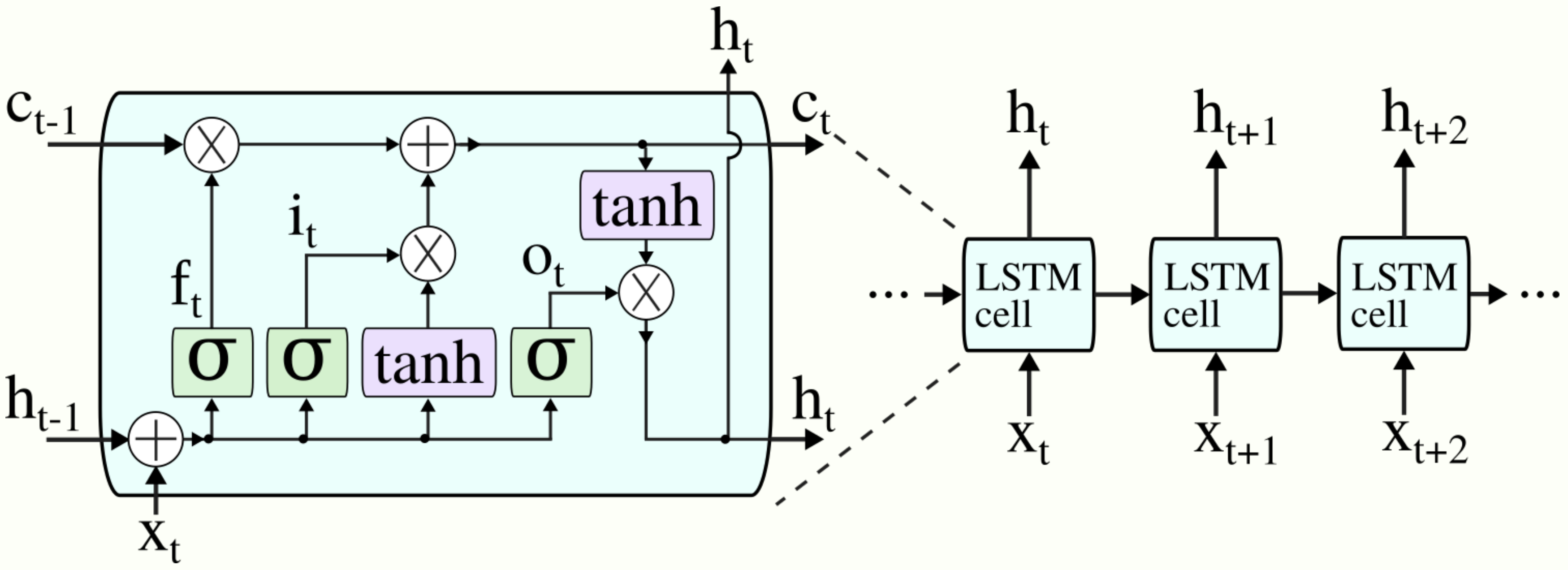
Deep learning neural network

- 37 (BGP)/109 (NSL-KDD) RNNs, 64 FC_1, 32 FC_2, and 16 FC_3 fully connected (FC) hidden nodes:



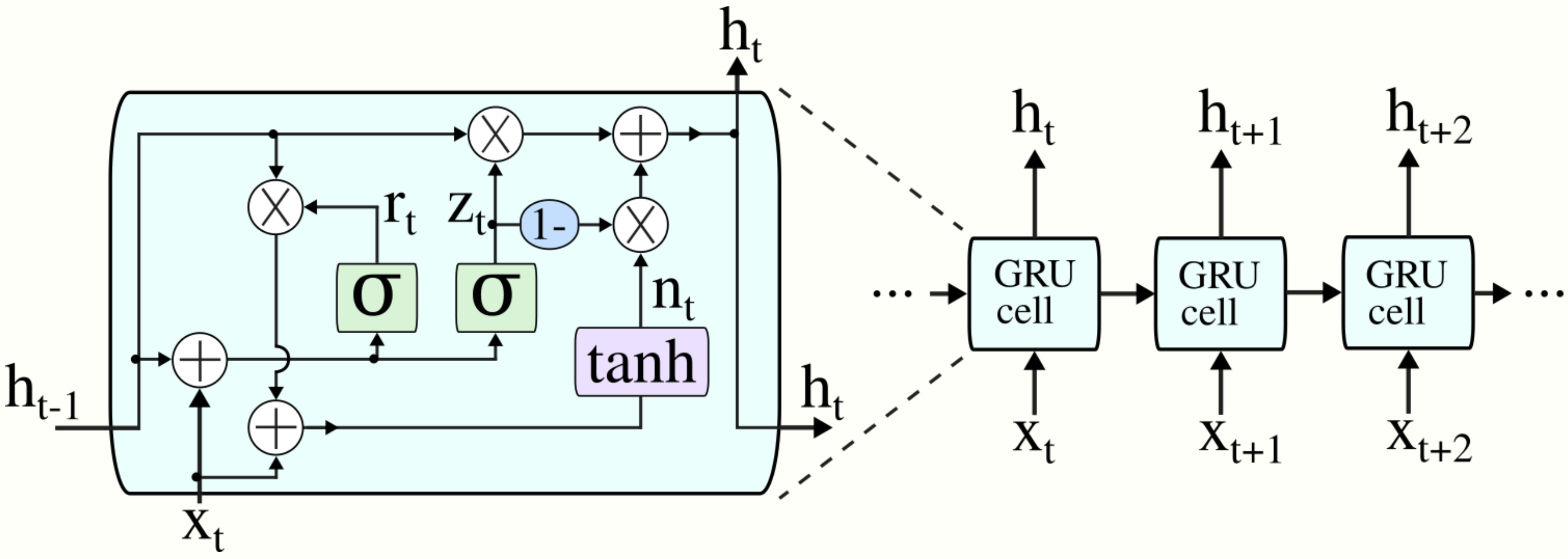
Long short-term memory: LSTM

- Repeating module for the LSTM neural network:



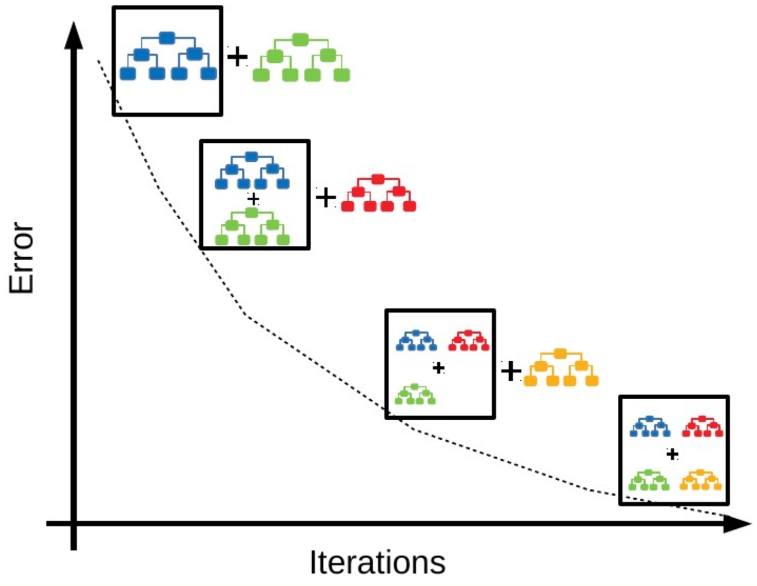
Gated recurrent unit: GRU

- Repeating module for the GRU neural network:



Gradient boosting machines

- Gradient boosting machines (GBMs): boosting algorithms that employ functional gradient descent to minimize the loss function
- GBDT: GBM variant that employs decision trees as estimators
- Generating a gradient boosting model:



https://medium.com/swlh/gradient-boosting-trees-for-classification-a-beginners-guide-596b594a14ea

Gradient boosting decision trees: GBDT

- When training a GBDT model with K estimators using N data points, the predicted output for the i^{th} data point $\boldsymbol{x}_i$ is:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(\boldsymbol{x}_i)\,,$$

where:

- f_k: the k^{th} decision tree
- $\boldsymbol{x}_i$: a row vector of matrix $\boldsymbol{X}$ containing input data and represents one collection sample

Gradient boosting decision trees: GBDT

- In the k^{th} iteration, predicted output is evaluated using the k^{th} estimator and k^{th} decision tree:

$$\hat{y}_i^{(k)} = \hat{y}_i^{(k-1)} + f_k(\boldsymbol{x}_i),$$

where:

- $\hat{y}_i^{(k)}$: predicted output of the i^{th} data point
- $\hat{y}_i^{(k-1)}$: previously predicted output
- f_k: the k^{th} decision tree

Gradient boosting decision trees: GBDT

- Goal of the GBDT models is to minimize the objective function:

$$\mathcal{L}^{(k)} = \sum_{i=1}^{N} l\left(y_i - \hat{y}_i^{(k)}\right) + \Omega(f_k),$$

where:

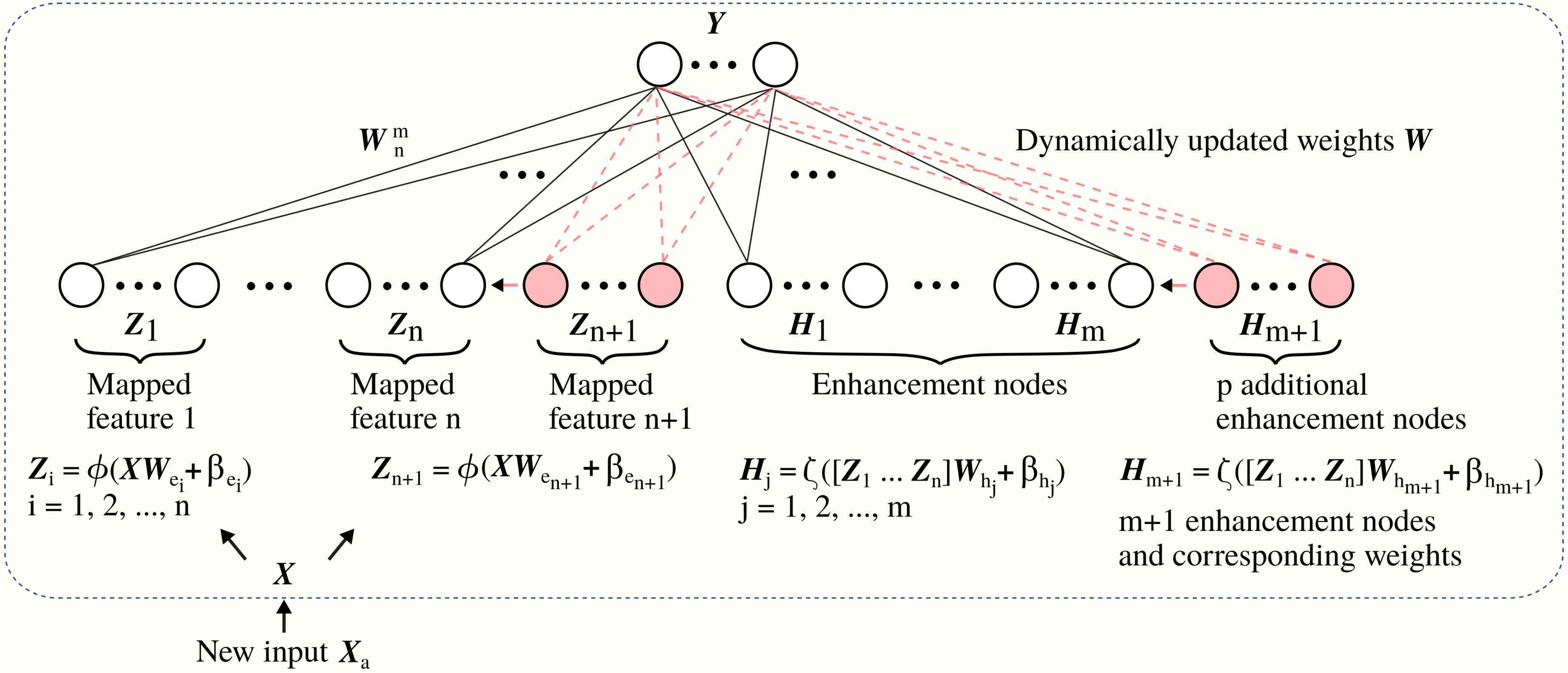
- $l(\cdot)$: loss function
- y_i: true value of the i^{th} data point
- $\hat{y}_i^{(k)}$: predicted output of the i^{th} data point for the k^{th} iteration
- $\Omega(f_k)$: (optional) regularization term

GBDT: XGBoost, LightGBM, CatBoost

- XGBoost:
  - adds an L^2 norm regularization term to avoid over-fitting
  - employs the second-order Taylor series to approximate its objective function
- LightGBM:
  - accelerate the training speed by using gradient-based one-side sampling (GOSS) and exclusive feature bundling (EFB)
- CatBoost:
  - deals with categorical features
  - employs target statistic to convert categorical to numerical features
  - employs ordered boosting

Broad learning system

- Broad Learning System (BLS) algorithm with increments of mapped features, enhancement nodes, and new input data:



Original BLS

- State matrix $\boldsymbol{A}_x$ is constructed from groups of mapped features $\boldsymbol{Z}^n$ and enhancement nodes $\boldsymbol{H}^m$ as:

$$\begin{aligned}\boldsymbol{A}_x &= [\boldsymbol{Z}^n \mid \boldsymbol{H}^m] \\ &= \left[\phi\left(\boldsymbol{X}\boldsymbol{W}_{e_i} + \boldsymbol{\beta}_{e_i}\right) \mid \xi(\boldsymbol{Z}_x^n \boldsymbol{W}_{h_j} + \boldsymbol{\beta}_{h_j})\right], \\ &\qquad i = 1, 2, \ldots, n \text{ and } j = 1, 2, \ldots, m,\end{aligned}$$

where:

- ϕ and ξ: projection mappings
- $\boldsymbol{W}_{e_i}$, $\boldsymbol{W}_{h_j}$: weights
- $\boldsymbol{\beta}_{e_i}$, $\boldsymbol{\beta}_{h_j}$: bias parameters

Original BLS

- Moore-Penrose pseudo inverse of matrix $\boldsymbol{A}_x$ is computed to calculate the weights of the output:

$$\boldsymbol{W}_n^m = [\boldsymbol{A}_n^m]^+ \boldsymbol{Y}$$

- Calculated using ridge regression:

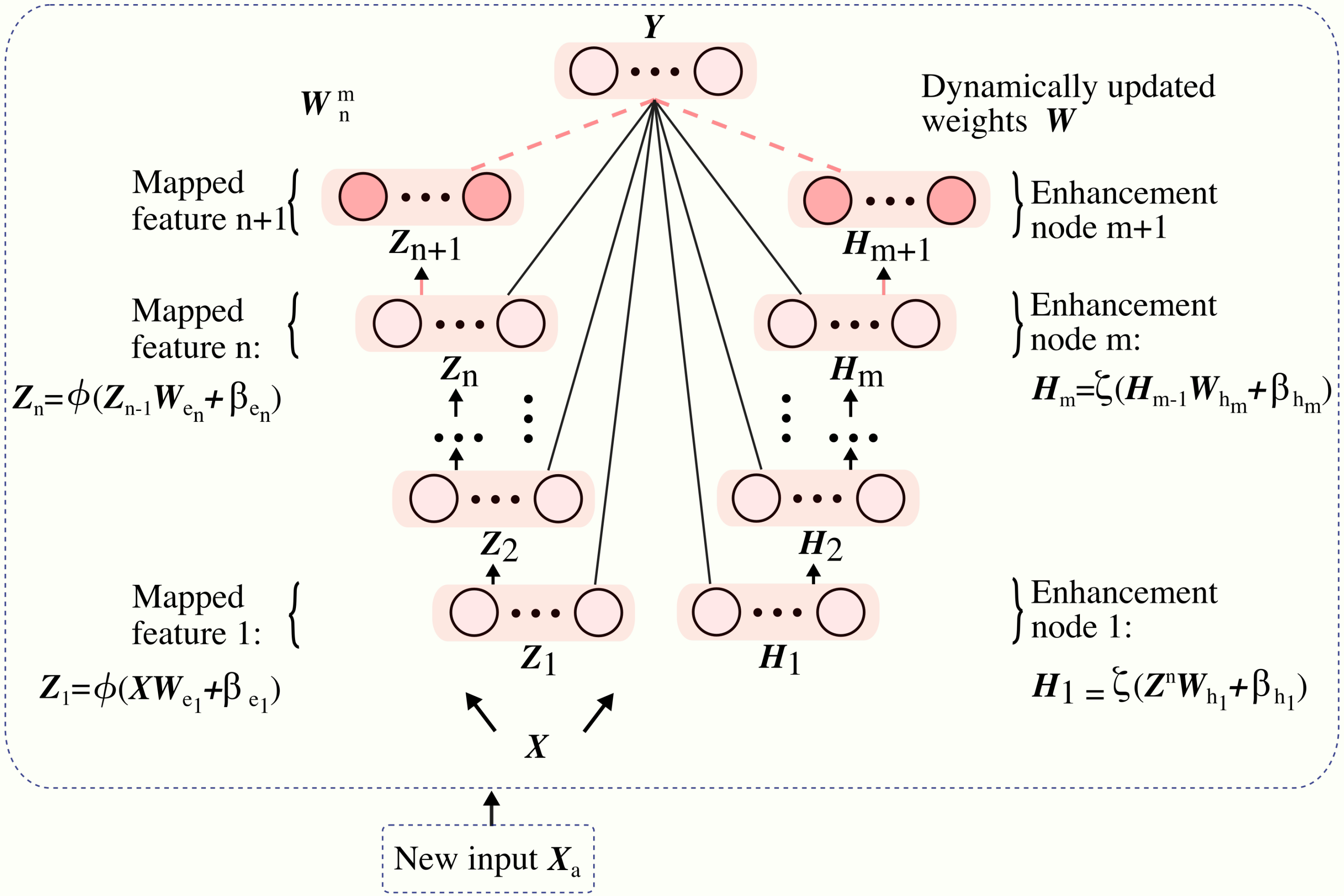
$$\boldsymbol{W}_n^m = [(\boldsymbol{A}_n^m)^T \boldsymbol{A}_n^m + \lambda \boldsymbol{I}]^{-1} (\boldsymbol{A}_n^m)^T \boldsymbol{Y}$$

- During the testing process, data labels are deduced using the calculated weights $\boldsymbol{W}_n^m$, mapped features $\boldsymbol{Z}_n$, and enhancement nodes $\boldsymbol{H}_m$:

$$\begin{aligned} \boldsymbol{Y} &= \boldsymbol{A}_n^m \boldsymbol{W}_n^m \\ &= [\boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_n | \boldsymbol{H}_1, \ldots, \boldsymbol{H}_m] \boldsymbol{W}_n^m \end{aligned}$$

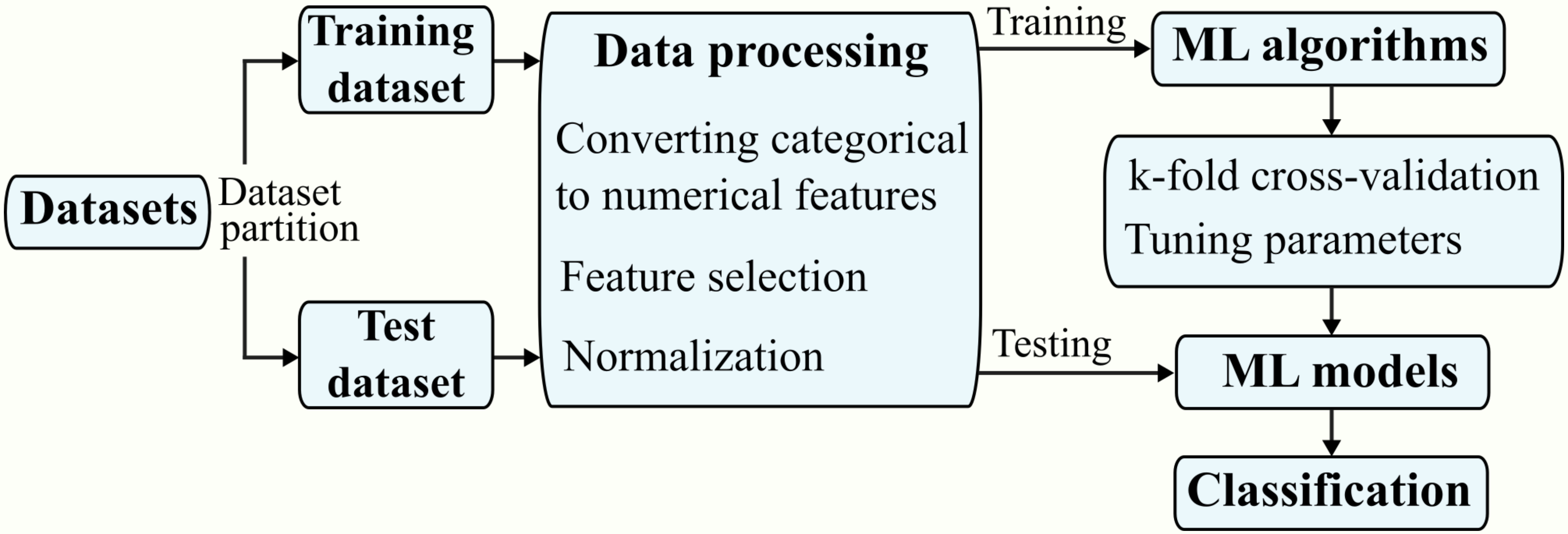
- Modified to include additional mapped features $\boldsymbol{Z}_{n+1}$, enhancement nodes $\boldsymbol{H}_{m+1}$, and/or input nodes $\boldsymbol{X}_a$

Cascades with incremental learning



Experimental procedure

- Architecture:



Performance metrics

- Training time
- Accuracy:
  - (TP + TN) / (TP + TN + FP + FN)
- F-Score signifies harmonic mean between precision and sensitivity (recall):
  - 2 x (precision x sensitivity) / (precision + sensitivity)

where:

  - Precision
    - TP / (TP + FP)
  - Sensitivity:
    - TP / (TP + FN)
- Confusion matrix: TP, FP, TN, FN

Performance comparison: RNN and BLS

	Datasets	$LSTM_2$	$LSTM_3$	$LSTM_4$	GRU_2	GRU_3	GRU_4
		Python (CPU)					
Training time (s)	Slammer	224.52	259.91	819.78	54.12	60.76	759.82
	NSL-KDD	4,481.73	4,614.66	11,478.62	1,108.31	1,161.80	11,581.30

	Datasets	BLS	RBF-BLS	CFBLS	CEBLS	CFEBLS
		Python (CPU)				
Training time (s)	Slammer	21.53	18.68	18.89	32.36	32.13
	NSL-KDD	99.47	98.27	98.13	108.23	108.14

Z. Li, A. L. Gonzalez Rios, G. Xu, and Lj. Trajković, “Machine learning techniques for classifying network anomalies and intrusions,” in *Proc. IEEE Int. Symp. Circuits and Systems*, Sapporo, Japan, May 2019.

Performance comparison: RNN and BLS

- RNN and BLS models: NSL-KDD dataset

	Accuracy (%)		F-Score (%)	
Model	$KDDTest^{+}$	$KDDTest^{-21}$	$KDDTest^{+}$	$KDDTest^{-21}$
$LSTM_4$	82.78	66.74	83.34	76.21
GRU_3	82.87	65.42	83.05	74.06
CFBLS	82.20	67.47	82.23	76.29

Z. Li, A. L. Gonzalez Rios, G. Xu, and Lj. Trajković, "Machine learning techniques for classifying network anomalies and intrusions," in Proc. *IEEE Int. Symp. Circuits and Systems*, Sapporo, Japan, May 2019.

Best performance:
CNN, RNN, and Bi-RNN models

- BGP dataset: WestRock ransomware attack

Model	Collection site	Training time (s)	Accuracy (%)	F-Score (%)	Precision (%)	Sensitivity (%)	TP	FP	TN	FN
CNN	RIPE	18.79	55.33	70.96	57.04	93.85	3,754	2,827	53	246
CNN	Route Views	18.66	57.67	72.96	58.04	98.23	3,929	2,841	39	71
GRU_4	RIPE	13.99	75.23	80.24	74.84	86.48	3,459	1,163	1,717	541
$LSTM_4$	Route Views	18.95	55.42	70.72	57.20	92.60	3,704	2,771	109	296
$Bi\text{-}GRU_4$	RIPE	20.59	78.49	81.92	80.10	83.83	3,353	833	2,047	647
$Bi\text{-}GRU_3$	Route Views	21.89	62.50	69.70	65.73	74.18	2,967	1,547	1,333	1,033

Best performance:
XGBoost, LightGBM, and CatBoost models

- CICIDS2017, CSE-CIC-IDS2018, CICDDoS2019

Model	Dataset	Training time (s)	Accuracy (%)	F-Score (%)	Precision (%)	Sensitivity (%)	TP	FP	TN	FN
XGBoost	CICIDS2017	24.49	98.62	98.72	99.43	98.02	98,684	568	84,359	1,989
	CSE-CIC-IDS2018	14.43	99.90	99.39	99.99	98.79	20,731	1	240,314	254
	CICDDoS2019	62.99	99.99	99.99	99.99	99.99	2,541,767	7	1,151	6
LightGBM	CICIDS2017	3.35	97.93	98.06	99.94	96.25	96,896	60	84,867	3,777
	CSE-CIC-IDS2018	1.73	98.73	91.44	99.99	84.23	17,675	1	240,314	3,310
	CICDDoS2019	8.12	99.99	99.99	99.99	99.99	2,541,767	8	1,150	6
CatBoost	CICIDS2017	20.27	98.01	98.13	99.91	96.41	97,056	83	84,844	3,617
	CSE-CIC-IDS2018	19.03	99.95	99.72	99.97	99.46	20,872	6	240,309	113
	CICDDoS2019	17.38	99.99	99.99	99.99	99.99	2,541,762	19	1,139	11

Z. Li, A. L. Gonzalez Rios, and Lj. Trajković, "Classifying denial of service attacks using fast machine learning algorithms," in Proc. *IEEE Int. Conf. Syst., Man, Cybern.*, Melbourne, Australia, Oct. 2021, pp. 1221-1226.

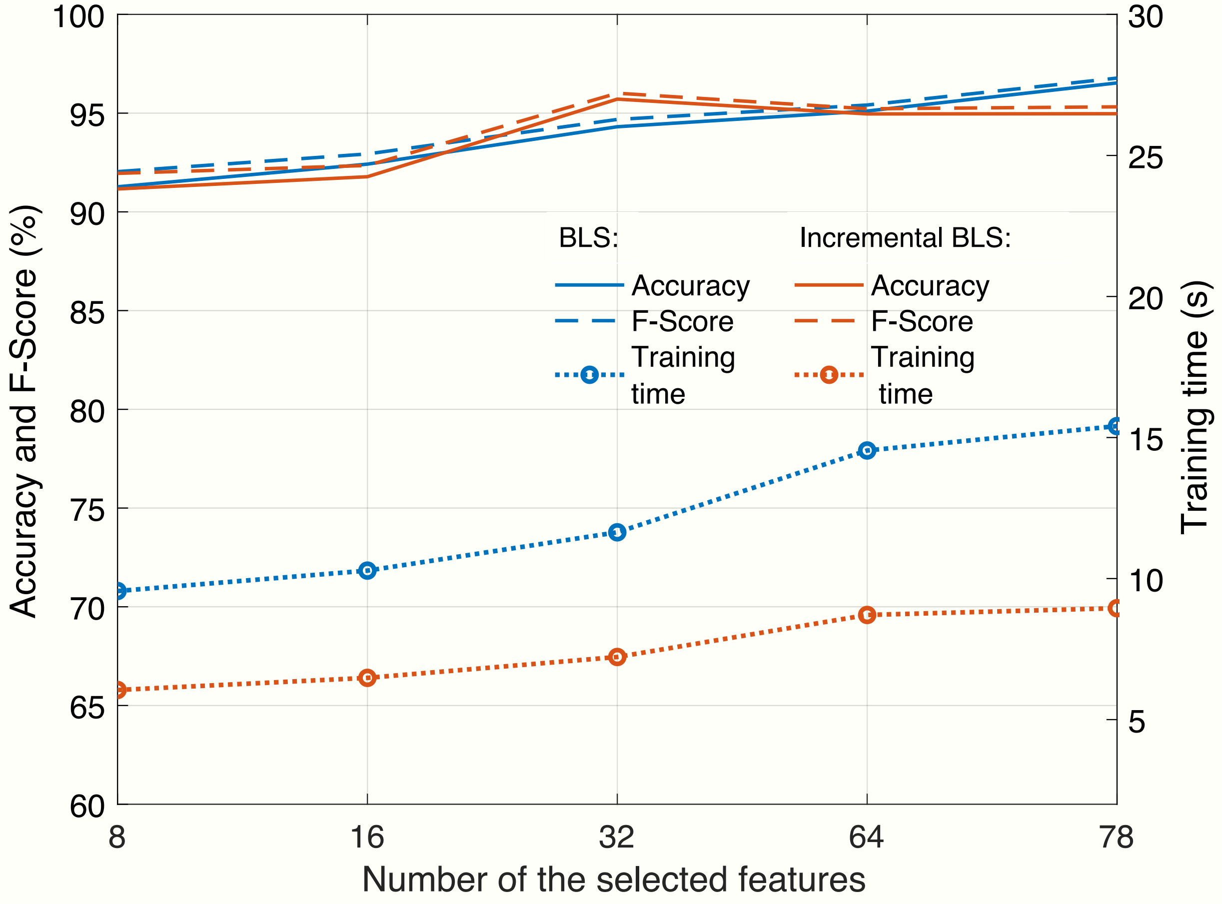
Roadmap

- Introduction
- Network anomalies and intrusions
- Feature selection and dimension reduction
- Applications of machine learning algorithms
- Variable features broad learning systems
- BGPGuard: BGP anomaly detection tool
- Conclusions and future work
- References

Variable features broad learning system

- Motivation:
  - best BLS models were sometimes derived by including all features
  - using a subset of relevant features may enhance performance
  - BLS models that achieved the best performance were trained using a single subset of features extracted from the input data
  - existing BLS-based algorithms include a single set of groups of mapped features (each group has a constant number of mapped features)
  - extra-trees algorithm has been used to select most relevant features

Performance: BLS and Incremental BLS, CIC 2017 Dataset



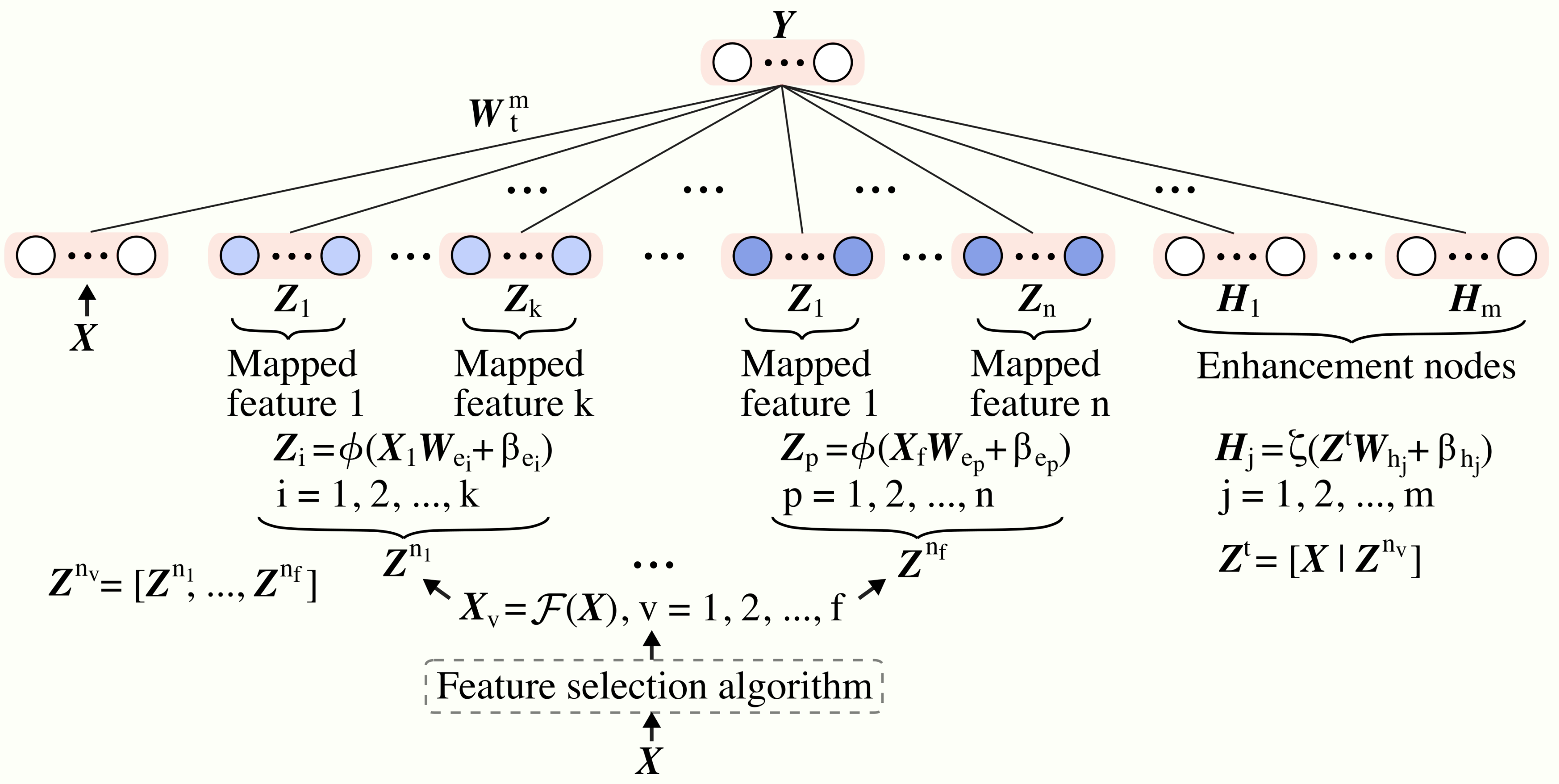
A. L. Gonzalez Rios, Z. Li, K. Bekshentayeva, and Lj. Trajković, "Detection of denial of service attacks in communication networks," in Proc. *IEEE Int. Symp. Circuits and Systems*, Seville, Spain, Oct. 2020.

Variable features broad learning system

- Variable features BLS without (VFBLS) and with cascades (VCFBLS) with and without incremental learning consist of:
  - variable number of mapped features and groups of mapped features
  - a feature selection algorithm to create subsets of input data
- VFBLS and VCFBLS enable:
  - derivation of generalized models
  - integration of selecting features and generating models
  - reduction of the training time by employing a smaller number of features

Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Variable features broad learning system: VFBLS



Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Variable features broad learning system: VFBLS

- Subsets of input data $\boldsymbol{X}$ using a feature selection algorithm $\mathcal{F}$:

$$\boldsymbol{X}_v = \mathcal{F}(\boldsymbol{X}), \qquad v = 1, 2, \ldots, f$$

- Sets of groups of mapped features: $\boldsymbol{Z}^{n_v} = [\boldsymbol{Z}^{n_1}, \ldots, \boldsymbol{Z}^{n_f}]$
- Concatenation of $\boldsymbol{X}$ and $\boldsymbol{Z}^{n_v}$: $\boldsymbol{Z}^t = [\boldsymbol{X} \mid \boldsymbol{Z}^{n_v}]$
- Enhancement nodes:

$$\boldsymbol{H}_j = \xi\left(\boldsymbol{Z}^t \boldsymbol{W}_{h_j} + \boldsymbol{\beta}_{h_j}\right), \qquad j = 1, 2, \ldots, m$$

where:

- f: number of subsets
- n_v: number of sets of mapped features
- ξ: projection mapping

Variable features broad learning system: VFBLS

- State matrix $\boldsymbol{A}_t^m$: concatenation of $\boldsymbol{Z}^t$ and $\boldsymbol{H}^m$
- Ridge regression algorithm is employed to compute the weights $\boldsymbol{W}_t^m$ based on $\boldsymbol{A}_t^m$ and given labels $\boldsymbol{Y}$
- Error function, minimized during the training process:

$$\boldsymbol{E}(\boldsymbol{W}_t^m) = (||\boldsymbol{W}_t^m - \boldsymbol{Y}||_2)^2 + (\lambda||\boldsymbol{W}_t^m||_2)^2$$

- Output weights:

$$\boldsymbol{W}_t^m = (\lambda \boldsymbol{I} + (\boldsymbol{A}_t^m)^T \boldsymbol{A}_t^m)^{-1} (\boldsymbol{A}_t^m)^T \boldsymbol{Y}$$

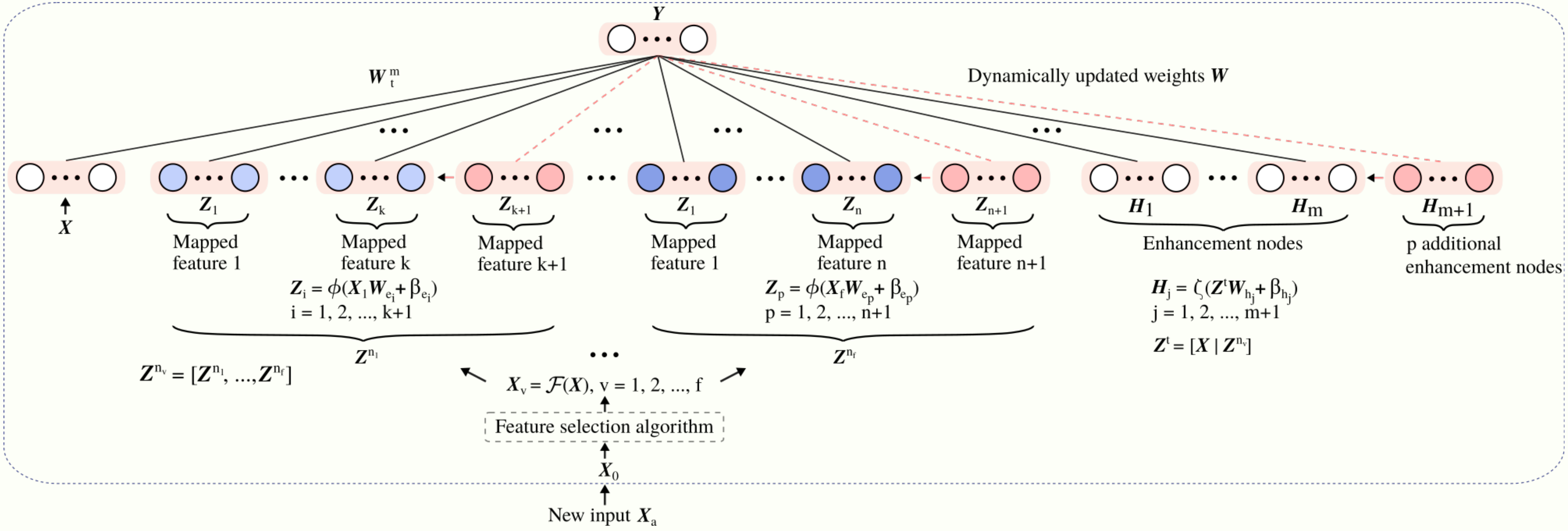
where:

- λ is the sparse regularization coefficient

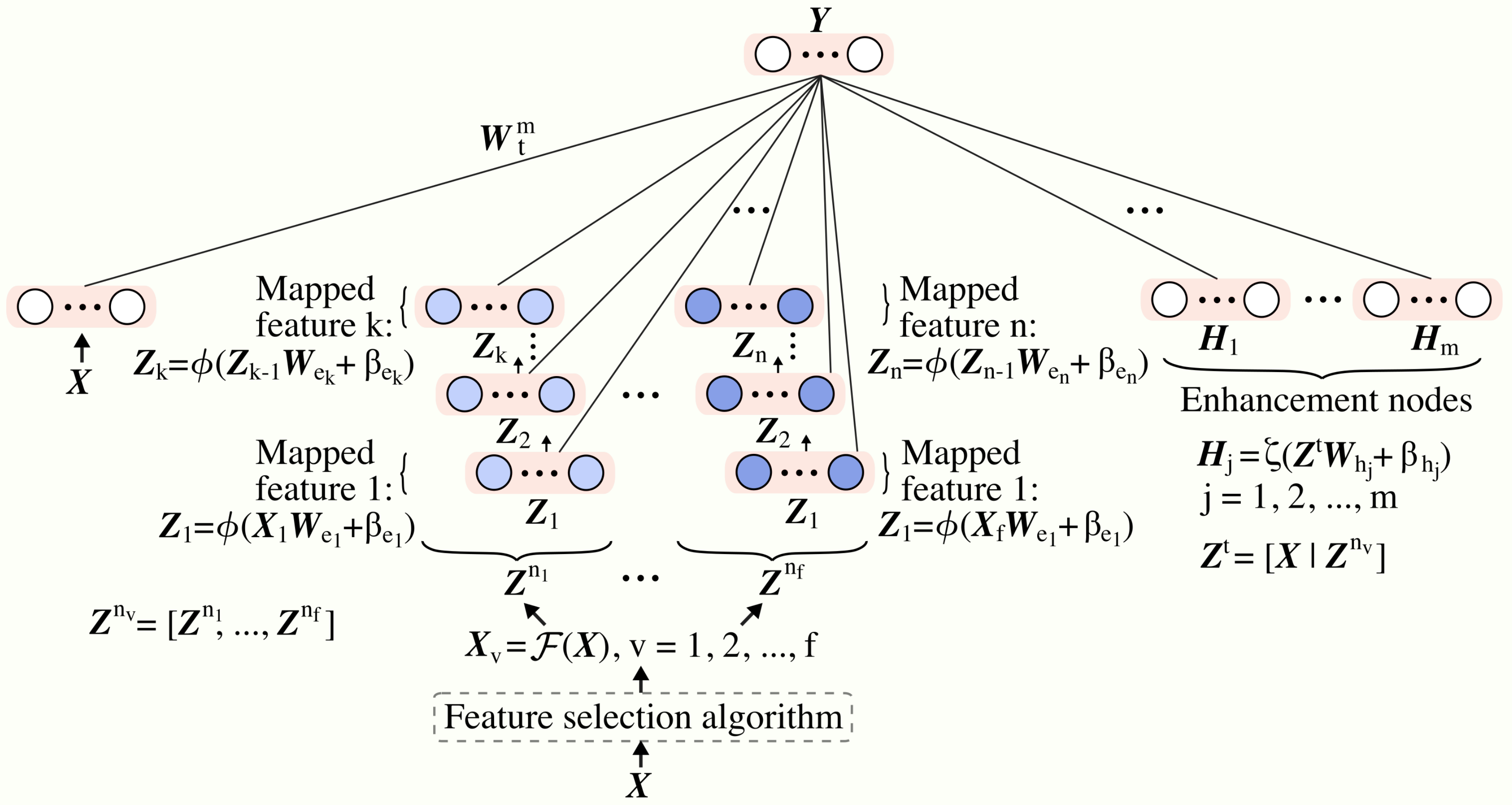
Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Variable features broad learning system: VFBLS

- Incremental learning:



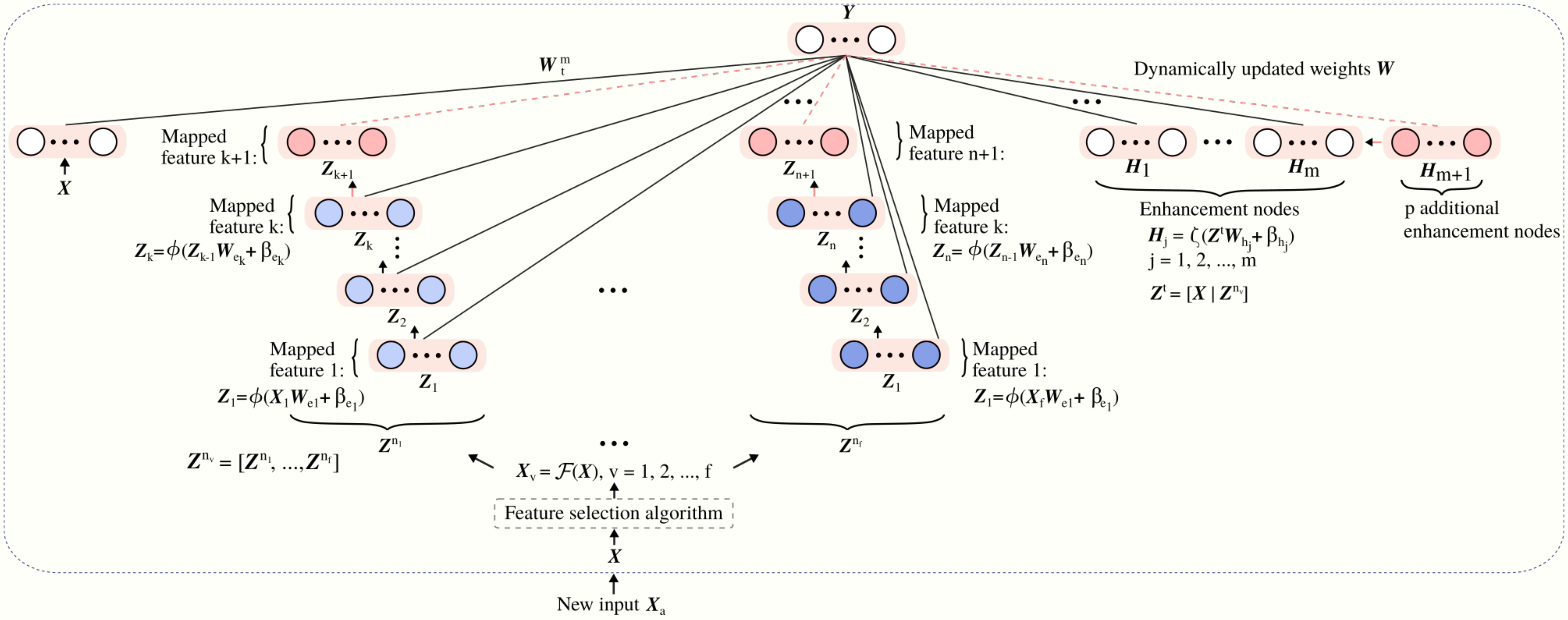
Variable features broad learning system: VCFBLS



Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Variable features broad learning system: VCFBLS

- Incremental learning:



Best parameters: VFBLS

Parameters	Slammer	Nimda	Code Red	NSL-KDD	CIC 2017	CIC 2018
			Number of features			
VFBLS		**8, 16, 37**		**32, 64, 109**	**32, 64, 78**	
Mapped features	100, 30, 40	20, 40, 30	20, 50, 30	20, 40, 30	15, 10, 10	10, 20, 10
Groups of mapped features	30, 20, 10	10, 20, 10	10, 10, 20	20, 20, 20	5, 10, 5	10, 5, 10
Enhancement nodes	100	50	100	40	40	40

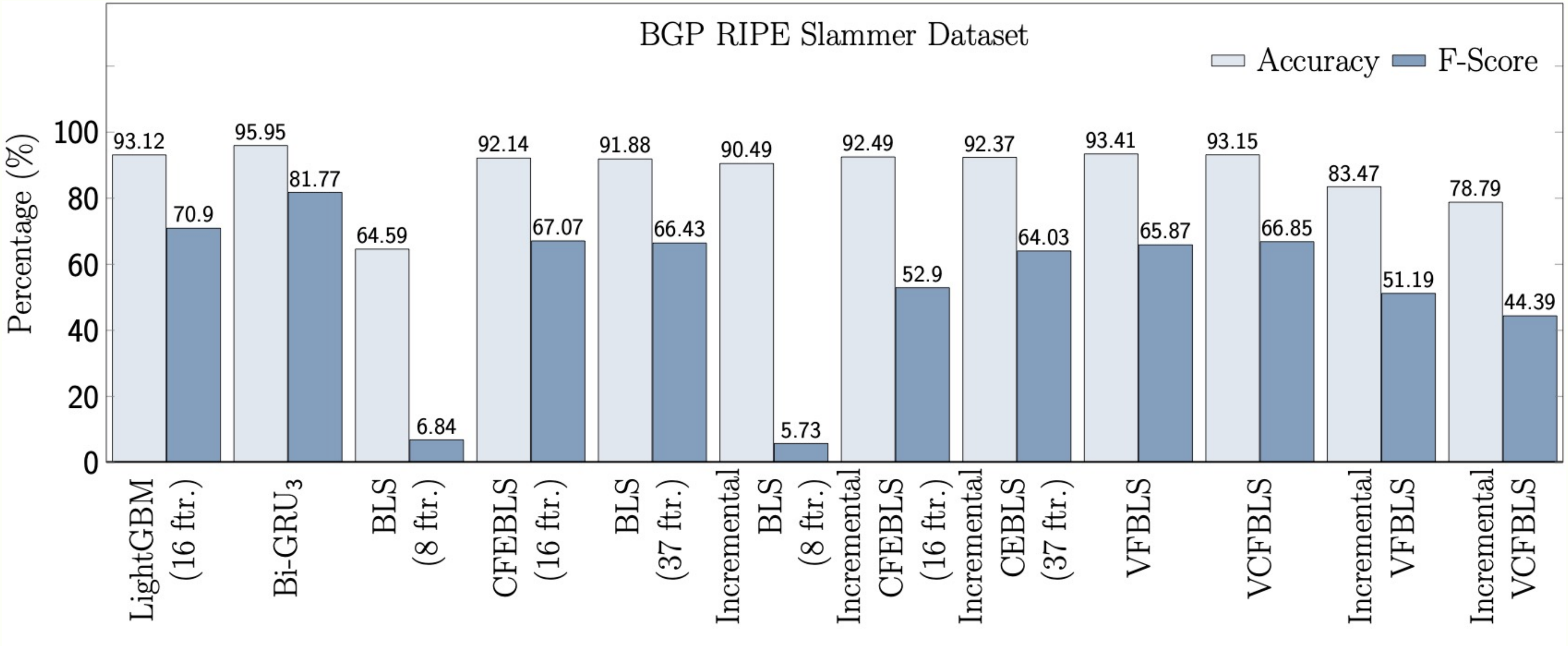
Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Best parameters: VCFBLS

Parameters	Slammer	Nimda	Code Red	NSL-KDD	CIC 2017	CIC 2018
Number of features						
VCFBLS	**8, 16, 37**			**32, 64, 109**	**32, 64, 78**	
Mapped features	200, 30, 30	20, 30, 30	30, 40, 40	20, 40, 30	10, 20, 10	10, 10, 20
Groups of mapped features	20, 10, 20	10, 20, 10	10, 10, 10	10, 20, 30	10, 5, 5	5, 10, 5
Enhancement nodes	100	100	100	60	40	40

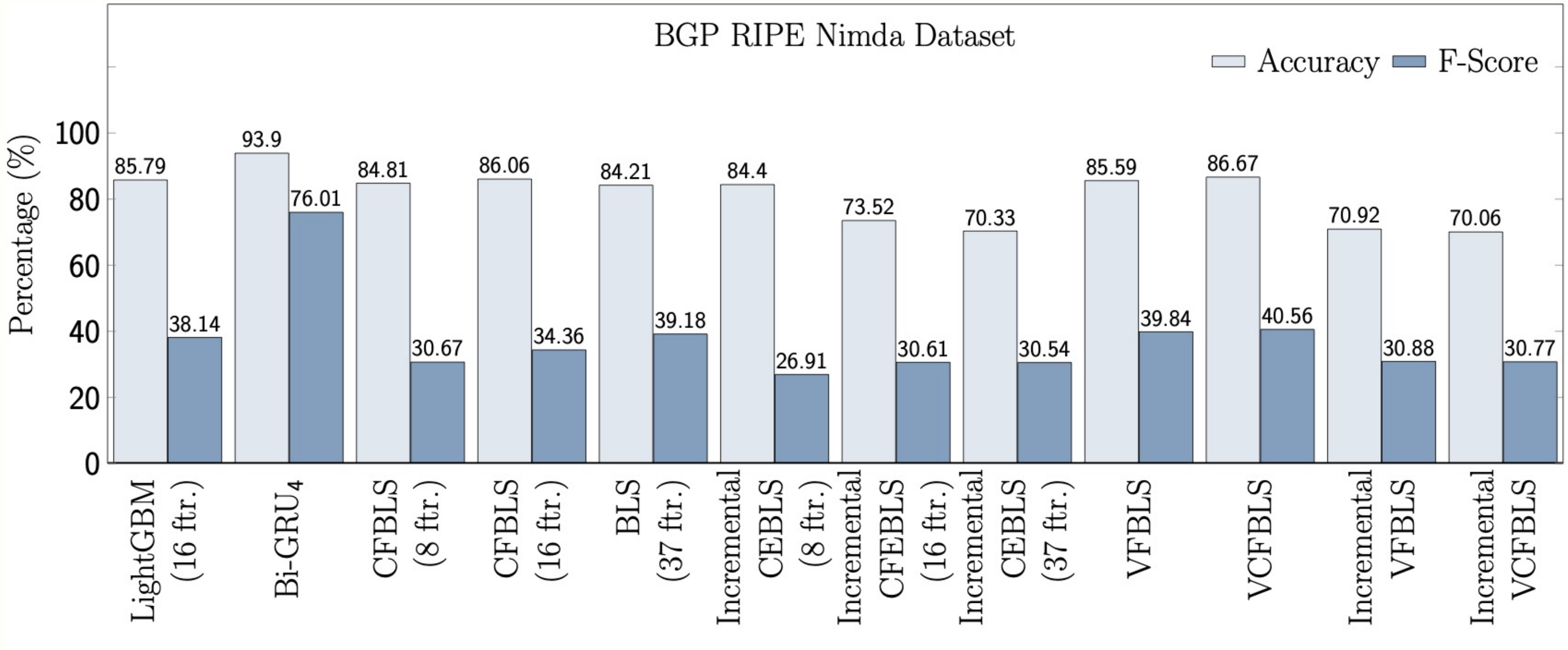
Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Best performance: Slammer



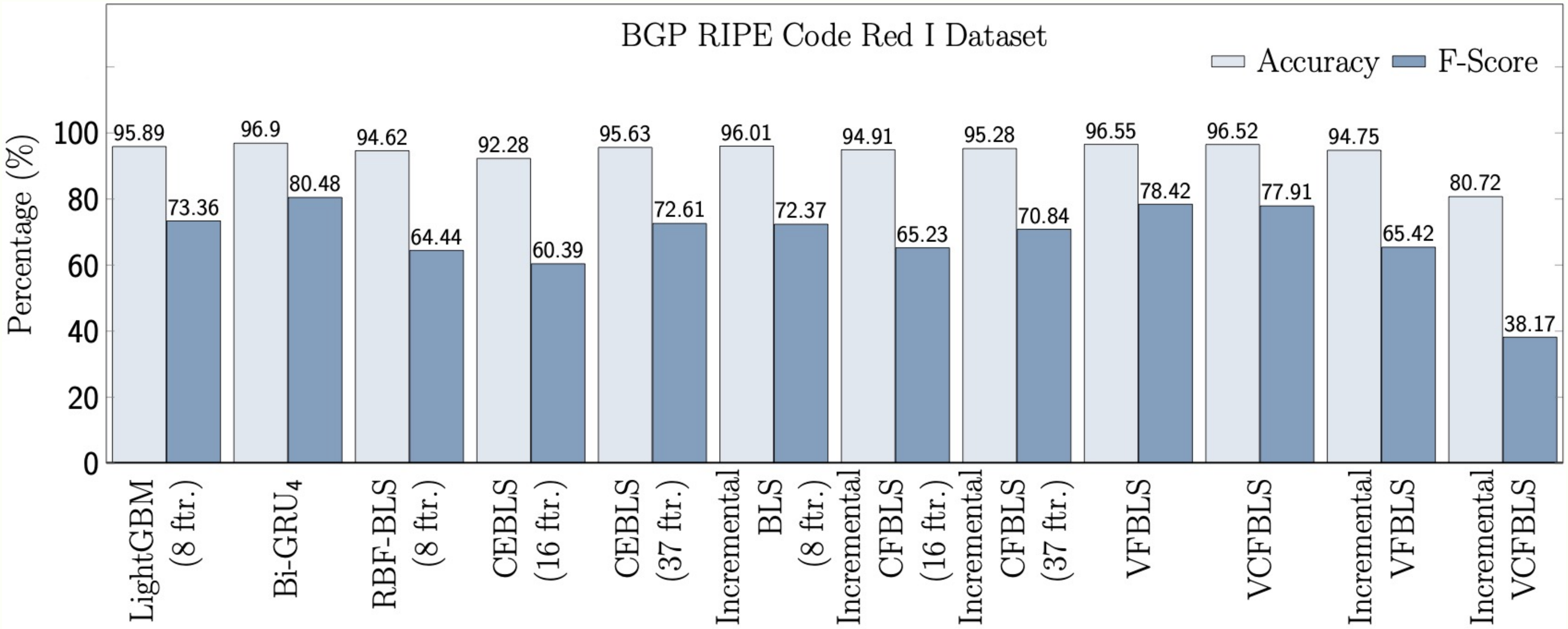
Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Best performance: Nimda



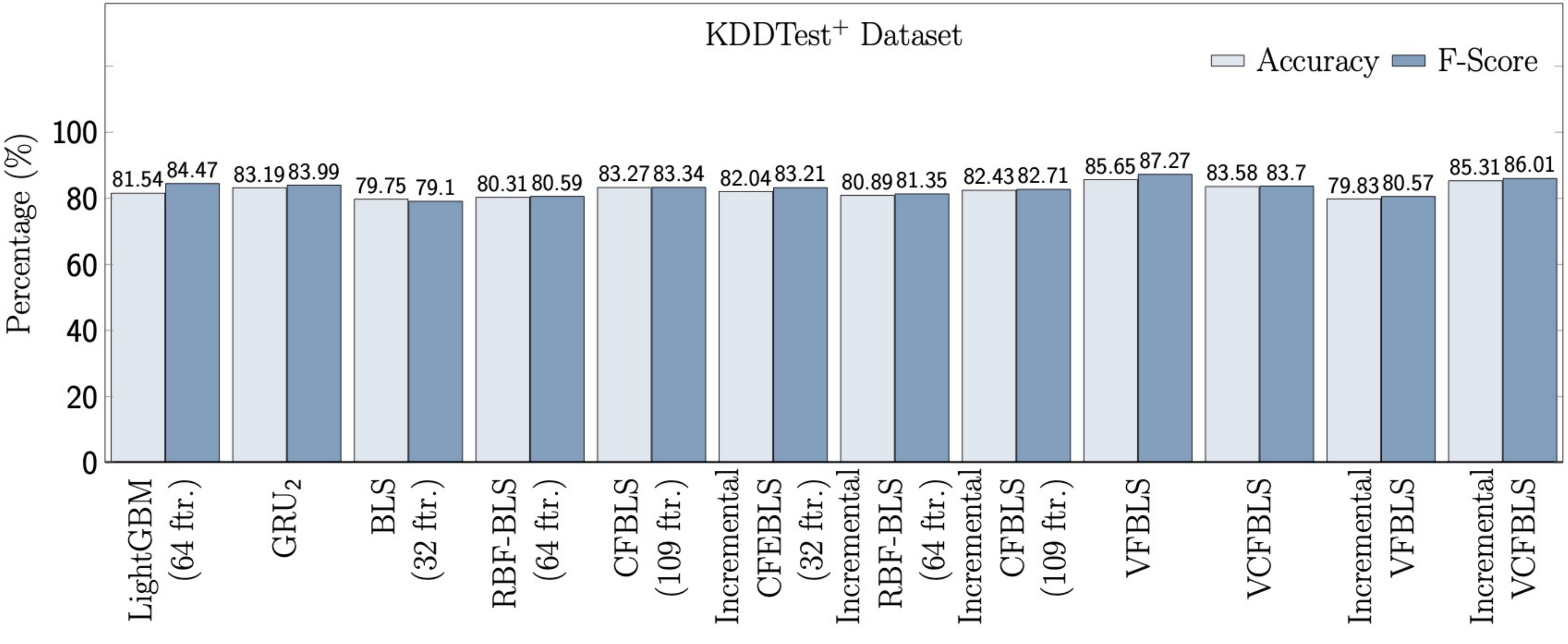
Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Best performance: Code Red



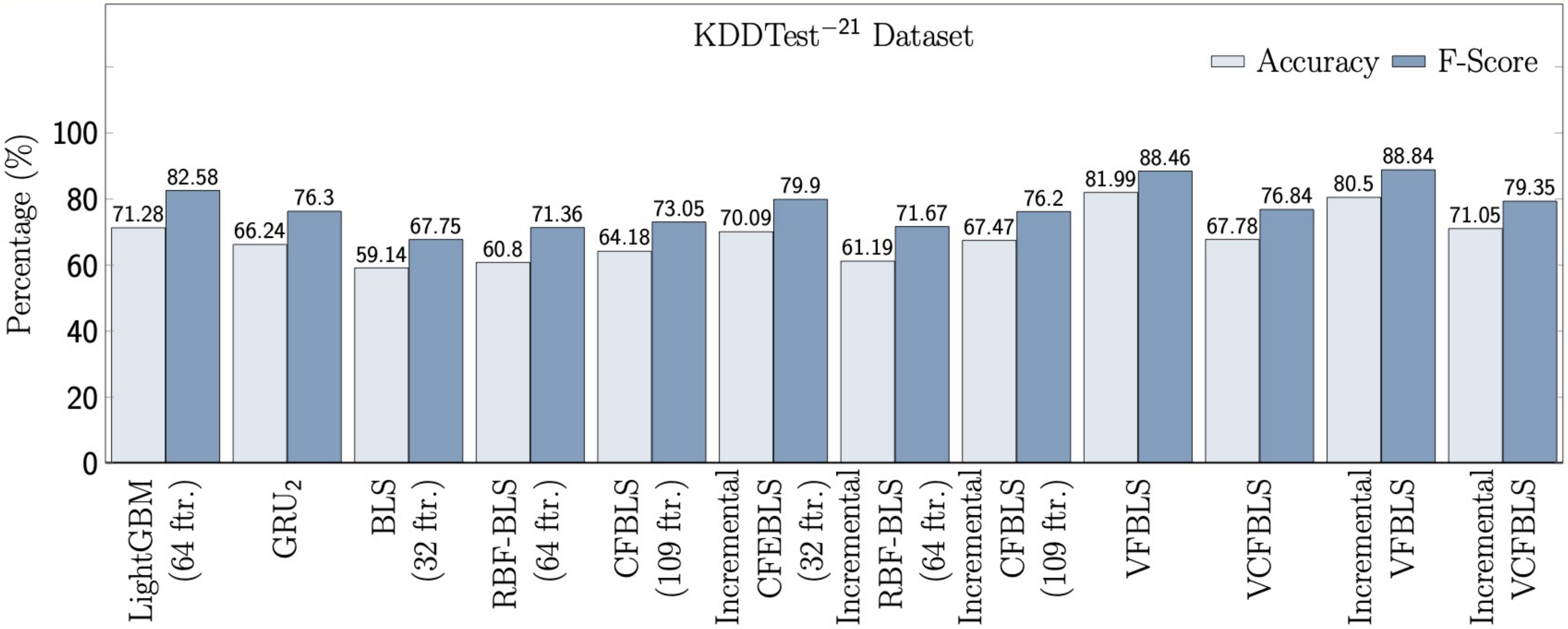
Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Best performance: KDDTest$^+$ (NSL-KDD)



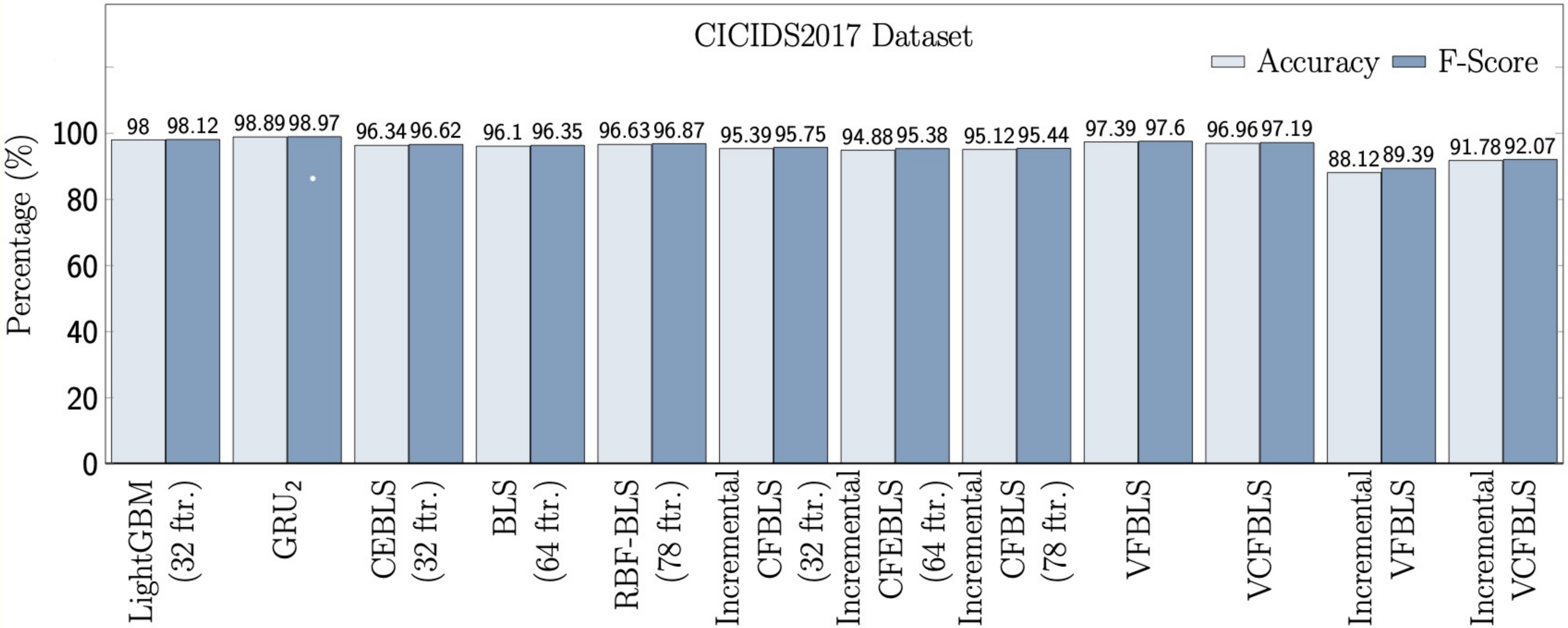
Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Best performance: KDDTest^{-21} (NSL-KDD)



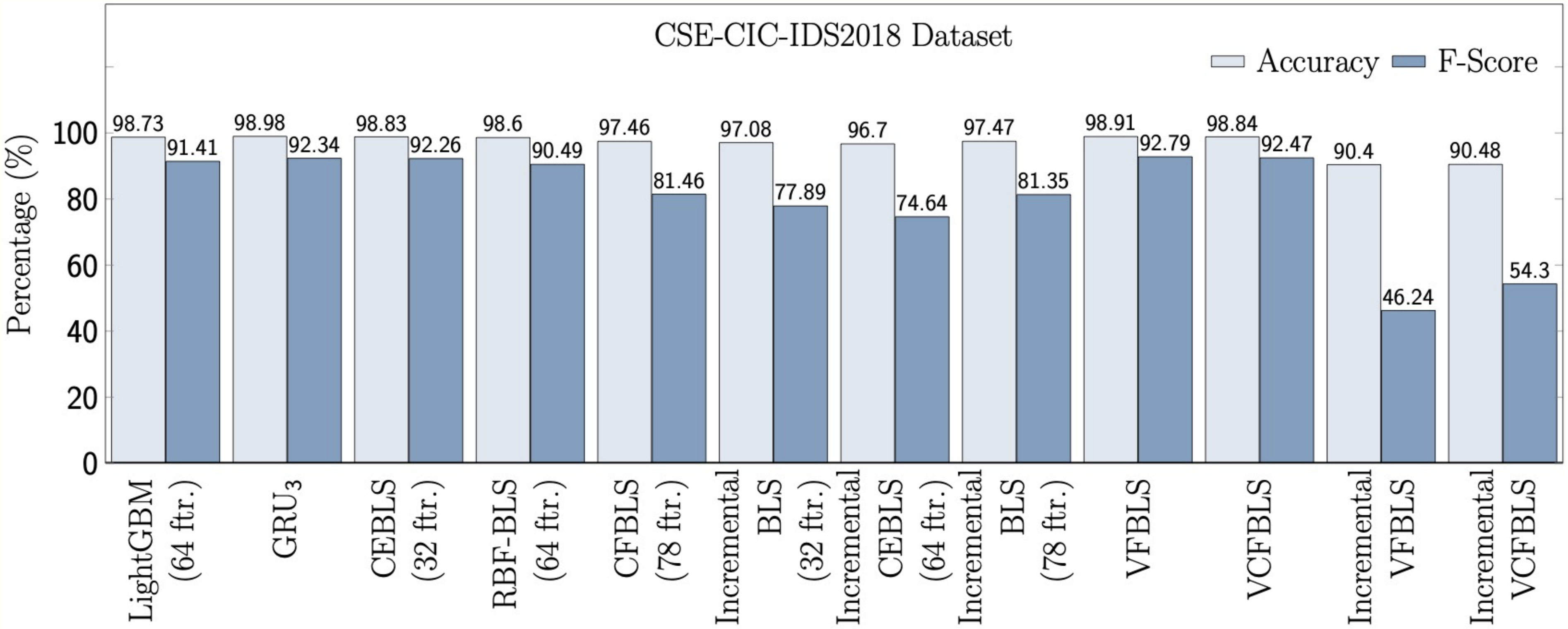
Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Best performance: CICIDS2017



Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Best performance: CSE-CIC-IDS2018



Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, “Machine learning for detecting anomalies and intrusions in communication networks,” *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Performance comparison: training time

- BGP datasets: Slammer, Nimda, Code Red

Dataset		LightGBM	RNN	BLS			Incremental BLS			Variable BLS		Incremental Variable BLS	
	Model		Bi-GRU$_3$	BLS	CFEBLS	BLS	BLS	CFEBLS	CEBLS	VFBLS	VCFBLS	VFBLS	VCFBLS
Slammer	**(No. ftr.)**	(16)		(8)	(16)	(37)	(8)	(16)	(37)				
	Time (s)	0.02	212.83	6.47	24.09	15.38	216.62	37.83	8.75	9.22	13.86	1.82	1.66
	Model		Bi-GRU$_4$	CFBLS	CFBLS	BLS	CEBLS	CFEBLS	CEBLS	VFBLS	VCFBLS	VFBLS	VCFBLS
Nimda	**(No. ftr.)**	(16)		(8)	(16)	(37)	(8)	(16)	(37)				
	Time (s)	0.11	219.50	3.51	1.29	2.01	8.27	43.41	13.35	2.12	1.97	10.37	5.98
	Model		Bi-GRU$_4$	RBF-BLS	CEBLS	CEBLS	BLS	CEBLS	CEBLS	VFBLS	VCFBLS	VFBLS	VCFBLS
Code Red	**(No. ftr.)**	(8)		(8)	(16)	(37)	(8)	(16)	(37)				
	Time (s)	0.02	546.54	5.90	174.21	26.63	1.11	2.00	1.12	1.88	2.55	1.33	1.42

Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Performance comparison: training time

- NSL-KDD and CIC datasets:

Dataset		LightGBM	RNN	BLS			Incremental BLS			Variable BLS		Incremental Variable BLS	
NSL-KDD	**Model**		GRU_2	BLS	RBF-BLS	CFBLS	CFEBLS	RBF-BLS	CFBLS	VFBLS	VCFBLS	VFBLS	VCFBLS
	(No. ftr.)	(64)		(32)	(64)	(109)	(32)	(64)	(109)				
	Time (s)	0.92	4,831.55	39.77	11.10	24.84	26.05	36.74	83.03	31.21	31.32	28.92	60.43
CICIDS 2017	**Model**		GRU_2	CEBLS	BLS	RBF-BLS	CFBLS	CFEBLS	CFBLS	VFBLS	VCFBLS	VFBLS	VCFBLS
	(No. ftr.)	(32)		(32)	(64)	(78)	(32)	(64)	(78)				
	Time (s)	1.43	15,483.96	39.25	8.97	15.60	6.39	7.39	3.69	25.25	26.05	25.55	24.19
CSE-CIC-IDS2018	**Model**		GRU_3	CEBLS	RBF-BLS	CFBLS	BLS	CEBLS	BLS	VFBLS	VCFBLS	VFBLS	VCFBLS
	(No. ftr.)	(64)		(32)	(64)	(78)	(32)	(64)	(78)				
	Time (s)	0.99	26,887.14	33.46	4.65	4.13	5.65	11.59	6.78	21.30	21.38	24.83	14.86

Z. Li, A. L. Gonzalez Rios, and Lj. Trajkovic, "Machine learning for detecting anomalies and intrusions in communication networks," *IEEE JSAC*, vol. 39, no. 7, pp. 2254-2264, July 2021.

Roadmap

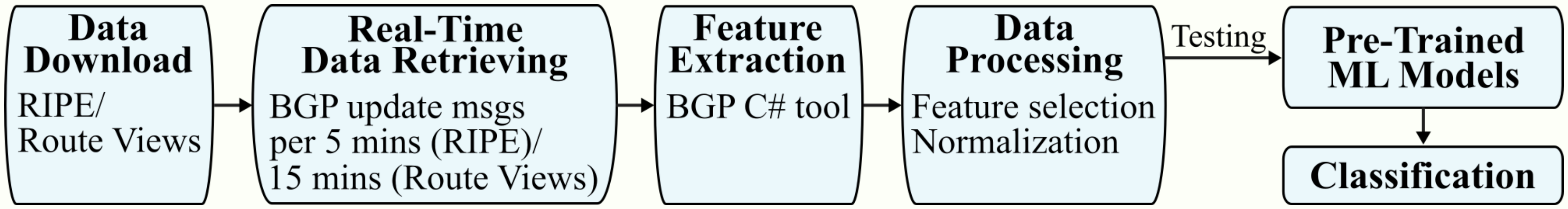
- Introduction
- Network anomalies and intrusions
- Feature selection and dimension reduction
- Applications of machine learning algorithms
- Variable features broad learning systems
- BGPGuard: BGP anomaly detection tool
- Conclusions and future work
- References

BGPGuard: BGP anomaly detection tool

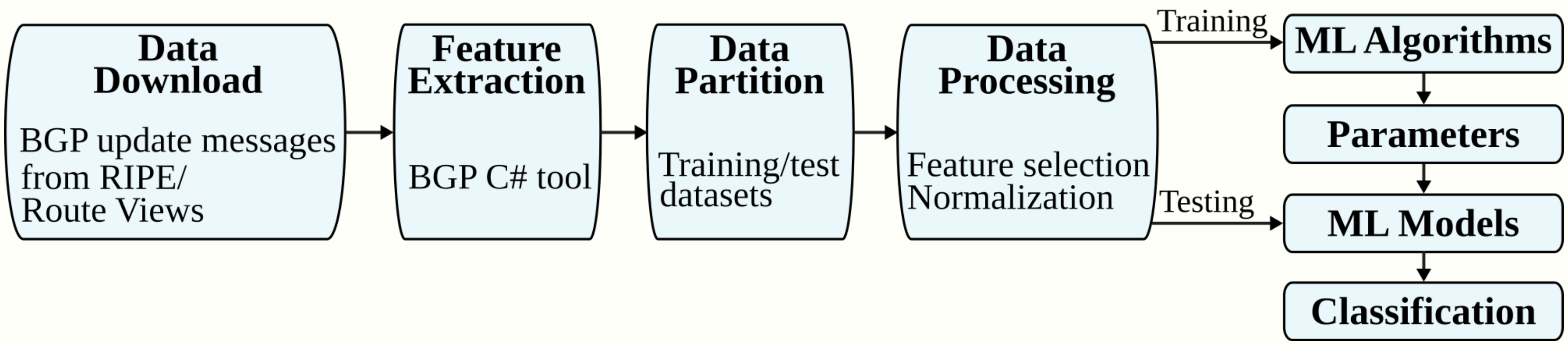
- BGPGuard: developed to integrate various stages of the anomaly detection process
- Modules: data download, feature extraction, data partition, data processing, machine learning algorithms, parameter selection, machine learning models, and classification
- Terminal-based:
  - Based on Python
- Web-based:
  - Front-end: HTML, CSS (Bootstrap: open-source CSS framework), Socket.IO (transport protocol written in a JavaScript for real-time web applications)
  - Back-end: Flask (micro web framework written in Python)

BGPGuard: architectures

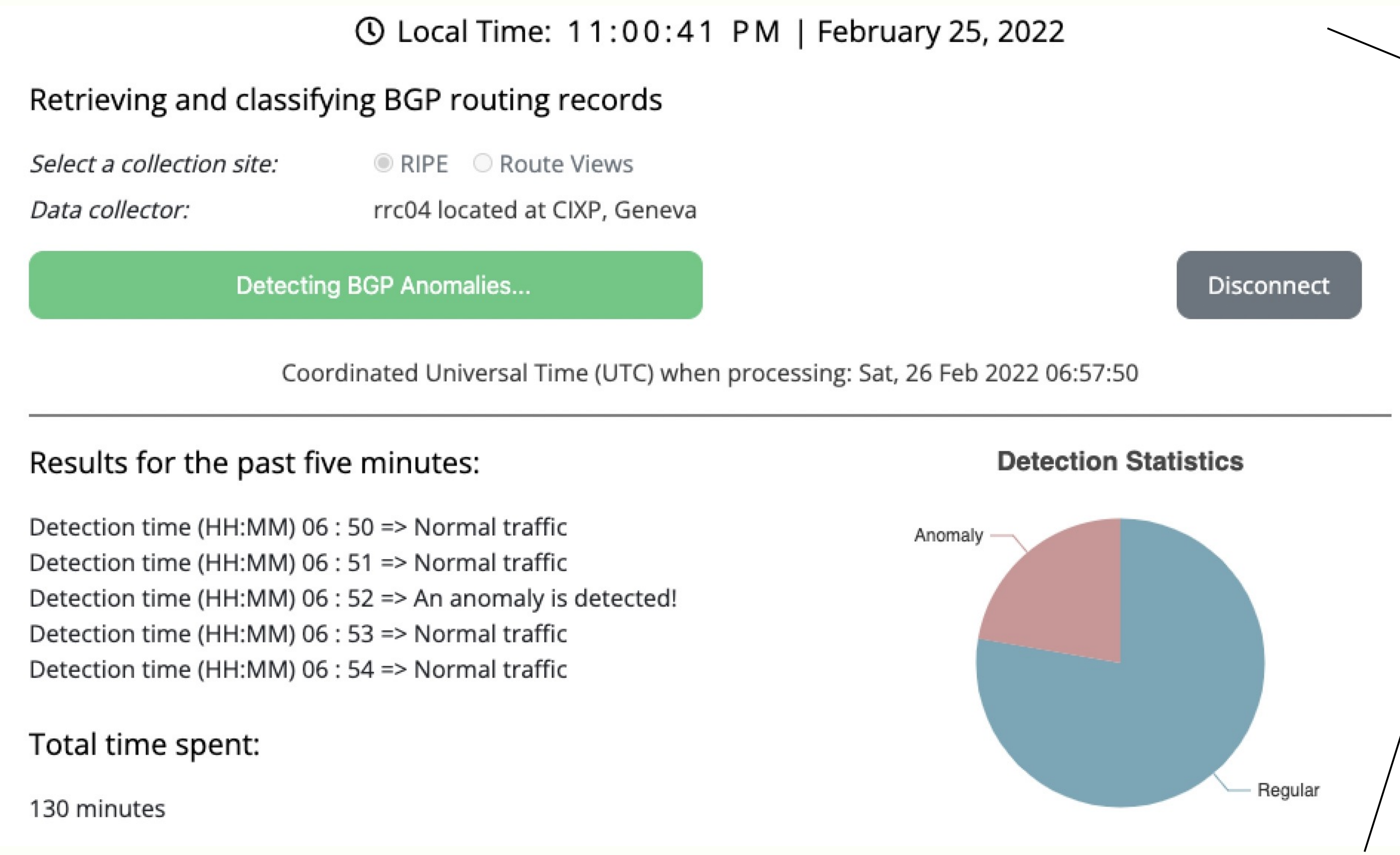
- Real-time detection:

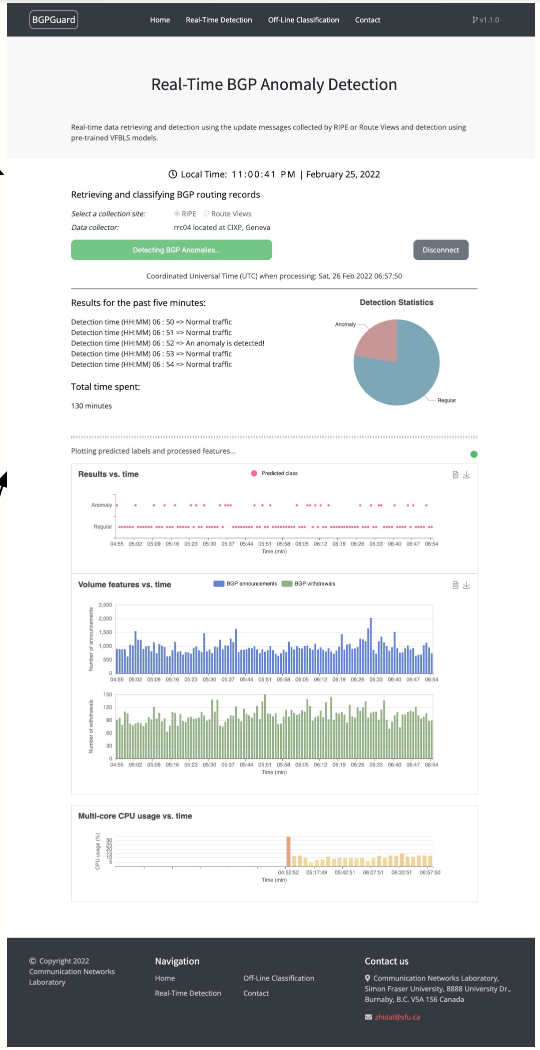


- Off-line classification:

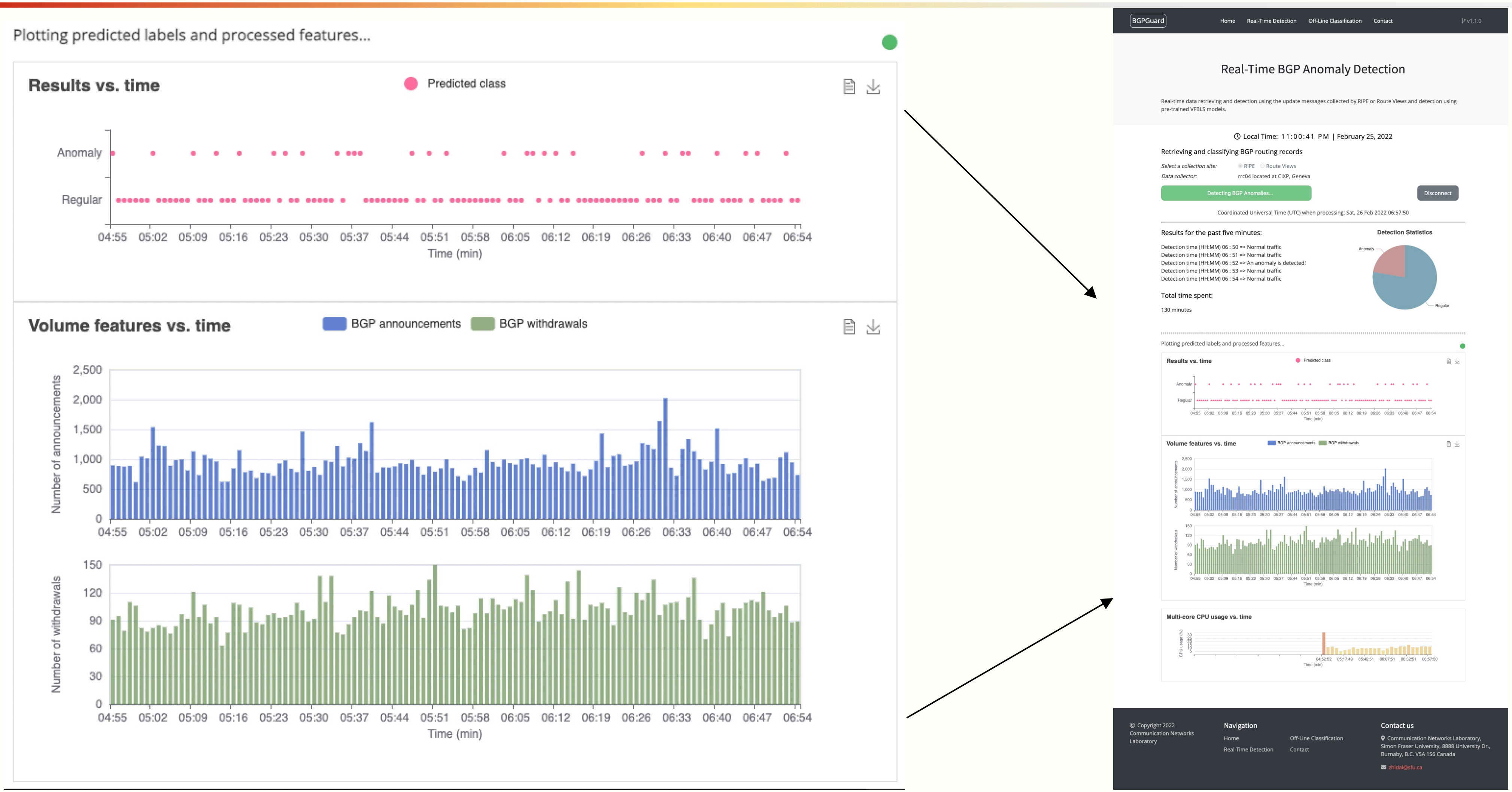


BGPGuard: web-based real-time detection



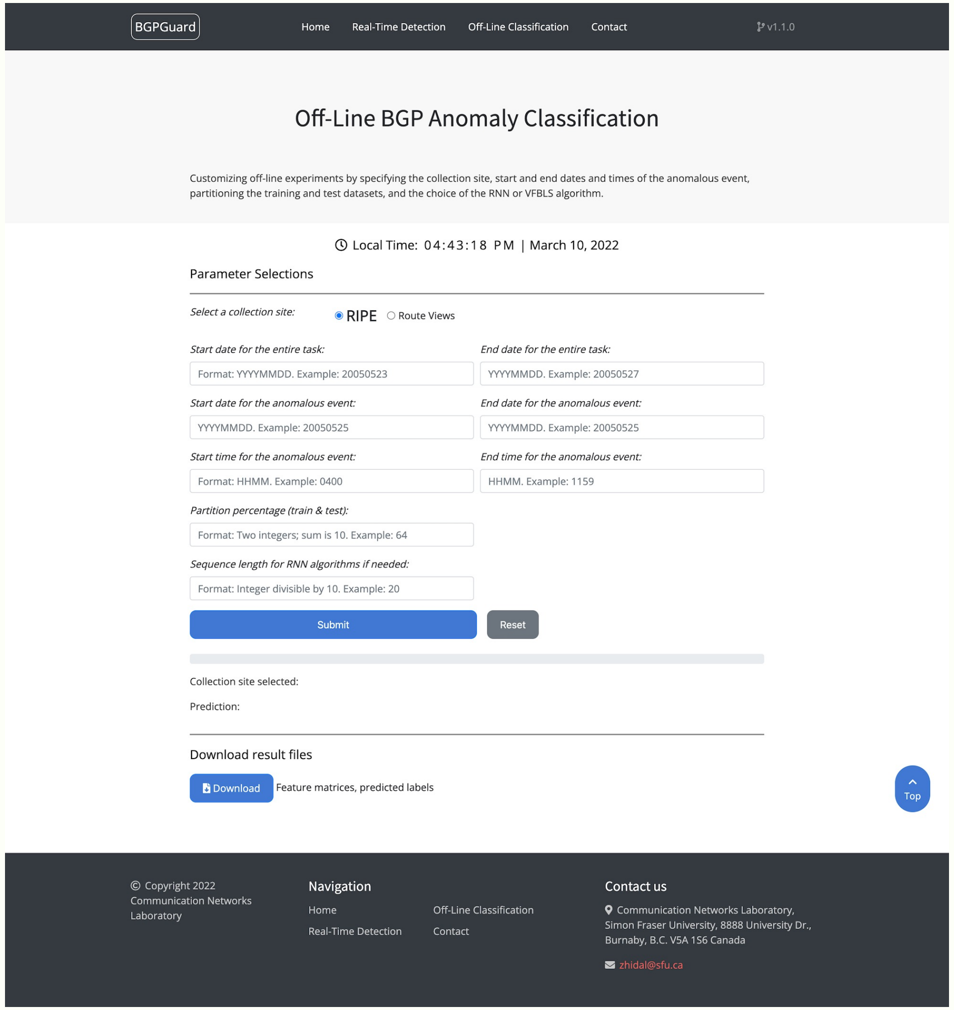


BGPGuard: web-based real-time detection



BGPGuard: web-based off-line classification

- Parameters:
  - "site": "RIPE"
  - "start_date": "20050523"
  - "end_date": "20050527"
  - "start_date_anomaly": "20050525"
  - "end_date_anomaly": "20050525"
  - "start_time_anomaly": "0400"
  - "end_time_anomaly": "1159"
  - ”partition_pct": "64"
  - "rnn_seq": 10



Roadmap

- Introduction
- Network anomalies and intrusions
- Feature selection and dimension reduction
- Applications of machine learning algorithms
- Variable features broad learning systems
- BGPGuard: BGP anomaly detection tool
- Conclusions and future work
- References

Conclusions

- We evaluated the performance of:
  - traditional, deep learning, and fast machine learning algorithms
- SVM, HMM, naïve Bayes, decision tree, and ELM algorithms
- LSTM and GRU deep recurrent neural networks with a variable number of hidden layers
- GBDT: XGBoost, LightGBM, CatBoost
- BLS models with and without incremental learning:
  - extensions (RBF-BLS, CFBLS, CEBLS, CFEBLS)
  - integrated extra-trees for feature selection (VFBLS, VCFBLS)
- Datasets collected from deployed network traffic (BGP) and testbeds (NSL-KDD, CIC)

Conclusions

- BLS models offer comparable performance to deep learning RNNs (LSTM, GRU, Bi-LSTM, Bi-GRU) models while requiring shorter training time
- LightGBM models required the shortest training time
- VFBLS and VCFBLS algorithms:
  - employed variable number of mapped features and groups of mapped features and an integrated feature selection algorithm
- VFBLS and VCFBLS models:
  - use various subsets of input data to generate mapped features leading to generalized models
  - outperform RNN, Bi-RNN, and other BLS models (most cases)
  - offer higher accuracy and F-Score
- BGPGuard: real-time detection and off-line classification

Future work

- Enhancing VFBLS and VCFBLS algorithms by implementing:
  - multiple feature selection algorithms to create subsets of input data
  - recurrent networks with various hidden layers to replace enhancement nodes and capture dynamic characteristics of the time-series data
- Implementing echo state networks and transformers
- Extracting additional features based on network topology
- BGPGuard:
  - additional datasets: NSL-KDD, CIC, UNSW-NB15
  - combining the existing algorithms
  - web server implementation

Roadmap

- Introduction
- Network anomalies and intrusions
- Feature selection and dimension reduction
- Applications of machine learning algorithms
- Variable features broad learning systems
- BGPGuard: BGP anomaly detection tool
- Conclusions and future work
- References

References: data sources

- RIPE NCC:
  https://www.ripe.net
- University of Oregon Route Views project:
  http://www.routeviews.org
- NSL-KDD dataset:
  https://www.unb.ca/cic/datasets/nsl.html
- CICIDS2017 dataset:
  https://www.unb.ca/cic/datasets/ids-2017.html
- CSE-CIC-IDS2018 dataset:
  https://www.unb.ca/cic/datasets/ids-2018.html
- CICDDoS2019 dataset:
  https://www.unb.ca/cic/datasets/ddos-2019.html

References: tools

- Python: https://pypi.org
- Pandas: https://pandas.pydata.org/
- PyTorch: https://pytorch.org/docs/stable/nn.html
- zebra-dump-parser:
  https://github.com/rfc1036/zebra-dump-parser
- BGP C# tool:
  http://www.sfu.ca/~ljilja/cnl/projects/BGP_datasets/index.html
- IEEE DataPort:
  Border Gateway Protocol (BGP) datasets:
  - https://ieee-dataport.org/open-access/border-gateway-protocol-bgp-routing-records-reseaux-ip-europeens-ripe-and-bcnet
  - https://ieee-dataport.org/open-access/border-gateway-protocol-bgp-routing-records-route-views
- BLS: http://www.broadlearning.ai/

References: intrusion detection

- J. P. A. Maranhão, J. P. C. L. da Costa, E. P. de Freitas, E. Javidi, and R. T. de Sousa, Jr., “Noise-robust multilayer perceptron architecture for distributed denial of service attack detection,” *IEEE Commun. Lett*., vol. 25, no. 2, pp. 402–406, Feb. 2021.
- P. Mishra, V. Varadharajan, U. Tupakula, and E. S. Pilli, “A detailed investigation and analysis of using machine learning techniques for intrusion detection,” *IEEE Commun. Surveys Tut*., vol. 21, no. 1, pp. 686–728, First quarter 2019.
- A. L. Buczak and E. Guven, “A survey of data mining and machine learning methods for cyber security intrusion detection,” *IEEE Commun. Surv. Tut*., vol. 18, no. 2, pp. 1153–1176, 2016.
- M. C. Libicki, L. Ablon, and T. Webb, The Defenders Dilemma: Charting a Course Toward Cybersecurity. Santa Monica, CA, USA: RAND Corporation, 2015.
- V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: a survey,” *ACM Comput. Surv.,* vol. 41, no. 3, pp. 15:1–15:58, July 2009.

References:
feature selection and dimension reduction

- P. Geurts, D. Ernst, and L. Wehenkel, “Extremely randomized trees,” *Mach. Learn*., vol. 63, no. 1, pp. 3–42, Apr. 2006.
- G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *Science*, vol. 313, pp. 504–507, July. 2006.
- I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: The MIT Press, 2016.

References: deep learning

- S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Oct. 1997.
- G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, “Improving neural networks by preventing co-adaptation of feature detectors,” *Computing Research Repository (CoRR)*, abs/1207.0580, pp. 1–18, Jul. 2012.
- K. Cho, B. van Merriënboer, C. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using RNN encoder–decoder for statistical machine translations,” in *Proc. 2014 Conf. Empirical Methods Natural Lang. Process*., Doha, Qatar, Oct. 2014, pp. 1724–1734.
- I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning. Cambridge*, MA, USA: The MIT Press, 2016.
- K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, “LSTM: a search space odyssey,” *IEEE Trans. Neural Netw. Learn. Syst*., vol. 28, no. 10, pp. 2222–2232, Oct. 2017.

References: BLS and GBDT

- C. L. P. Chen and Z. Liu, “Broad learning system: an effective and efficient incremental learning system without the need for deep architecture,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 1, pp. 10–24, Jan. 2018.
- C. L. P. Chen, Z. Liu, and S. Feng, “Universal approximation capability of broad learning system and its structural variations,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 4, pp. 1191–1204, Apr. 2019.

- T. Chen and C. Guestrin, “XGBoost: a scalable tree boosting system,” in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, San Francisco, CA, USA, Aug. 2016, pp. 785–794.
- G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye and T.-Y. Liu, “LightGBM: a highly efficient gradient boosting decision tree,” in *Proc. Int. Conf. Neural Inform. Process. Syst.*, Long Beach, CA, USA, Dec. 2017, 3146–3154.
- L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, “CatBoost: unbiased boosting with categorical features,” in *Proc. Int. Conf. Neural Inform. Process. Syst.*, Montreal, Quebec, Canada, Dec. 2018, 6639–6649.

Acknowledgements

- Examining committee:
  - External examiner: Francesco Sorrentino
  - Internal examiner: Bernhard Rabus
  - Committee member: Uwe Glässer
  - Committee member: Qianping Gu
  - Chair: Ivan V. Bajić
- Advisor: Ljiljana Trajković
- Colleagues and friends:
  - Ana Laura Gonzalez Rios and Hardeep Kaur Takhar
  - Prerna Batta, Kamila Bekshentayeva, Qingye Ding, Soroush Haeri, and Guangyu Xu
  - Jiawei He and Xiaoyu Liu

Thank you!