Detecting Network Anomalies and Intrusions

Ljiljana Trajković
ljilja@cs.sfu.ca

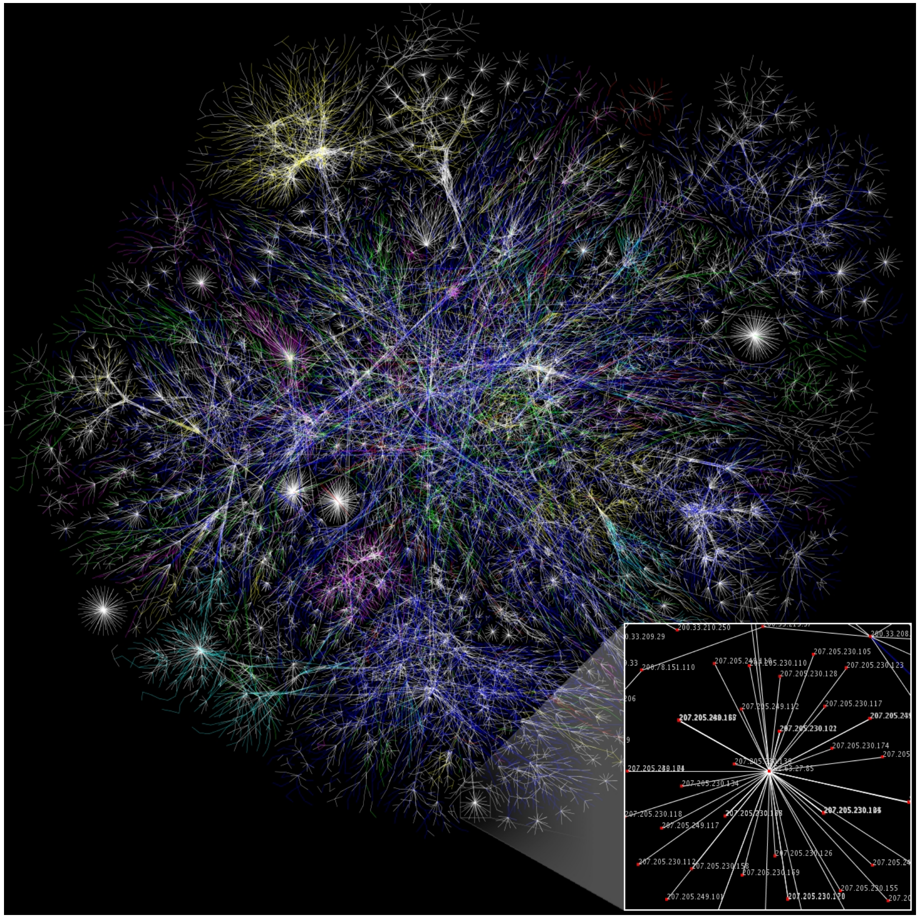
Communication Networks Laboratory
http://www.sfu.ca/~ljilja/cnl
School of Engineering Science
Simon Fraser University, Vancouver, British Columbia, Canada

Simon Fraser University Burnaby Campus

Roadmap

- Introduction:
  - Complex networks
  - Machine learning
- Data processing
- Machine learning models
- Experimental procedure
- Performance evaluation
- Conclusions and references

Complex Networks: The Internet



https://en.wikipedia.org/wiki/Complex_network#/media/File:Internet_map_1024.jpg
By The Opte Project - Originally from the English Wikipedia
https://commons.wikimedia.org/w/index.php?curid=1538544

Roadmap

- Introduction:
  - Complex networks
  - Machine learning
- Data processing
- Machine learning models
- Experimental procedure
- Performance evaluation
- Conclusions and references

Machine Learning

- Using machine learning techniques to detect network intrusions is an important topic in cybersecurity.
- Machine learning algorithms have been used to successfully classify network anomalies and intrusions.
- Supervised machine learning algorithms:
  - Support vector machine: SVM
  - Long short-term memory: LSTM
  - Gated recurrent unit: GRU
  - Broad learning system: BLS

Roadmap

- Introduction
- Data processing:
  - BGP datasets
  - NSL-KDD dataset
  - CICIDS2017
  - CSE-CIC-IDS2018
- Machine learning models
- Experimental procedure
- Performance evaluation
- Conclusions and references

CICIDS2017 and CSE-CIC-IDS2018

- CICIDS2017 and CSE-CIC-IDS2018:
  - Testbed used to create the publicly available dataset that includes multiple types of recent cyber attacks.
  - Network traffic collected between:
    - Monday, 03.07.2017
    - Friday, 07.07.2017
    - Wednesday, 14.02.2018
    - Friday, 02.03.2018

CICD2017 Dataset: Types of Intrusion Attacks

Attack	Label	Day	Number of intrusions
Brute force	FTP, SSH	Tuesday	7,935; 5,897
Heartbleed	Heartbleed	Wednesday	11
Web attack	Brute force, XSS, SQL Injection	Thursday morning	1,507; 652; 21
Infiltration	Infiltration, PortScan	Thursday and Friday afternoons	36; 158,930
Botnet	Bot	Friday morning	1,956
DoS	Slowloris, Hulk, GoldenEye, SlowHTTPTest	Wednesday	5,796; 230,124; 10,293; 5,499
DDos	DDoS	Friday afternoon	128,027

CICD2017 Dataset: Number of Flows

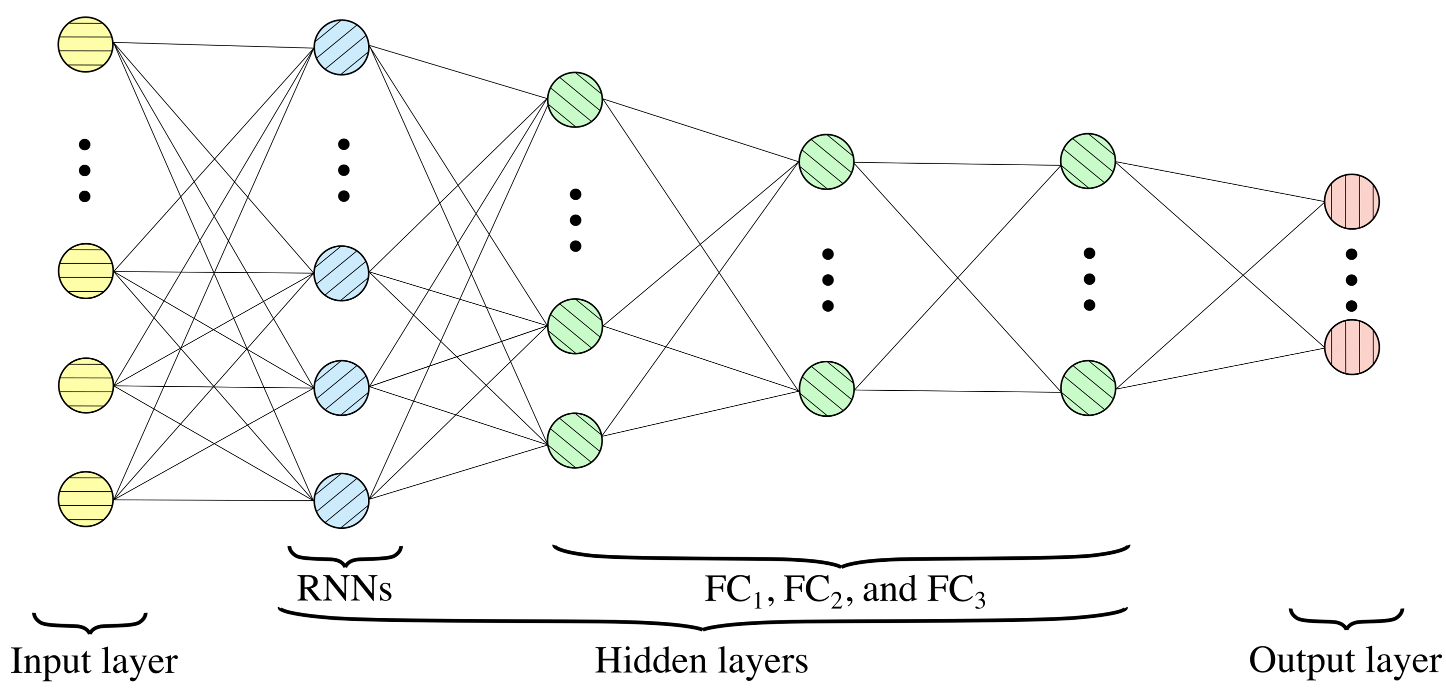
Day	Valid flows	Total
Monday	529,481	529,918
Tuesday	445,645	445,909
Wednesday	691,406	692,703
Thursday (morning)	170,231	170,366
Thursday (afternoon)	288,395	288,602
Friday (morning)	190,911	191,033
Friday (afternoon, PortScan)	286,096	286,467
Friday (afternoon, DDoS)	225,711	225,745

Roadmap

- Introduction
- Data processing
- Machine learning models:
  - Deep learning: multi-layer recurrent neural networks
  - Broad learning system
- Experimental procedure
- Performance evaluation
- Conclusions and references

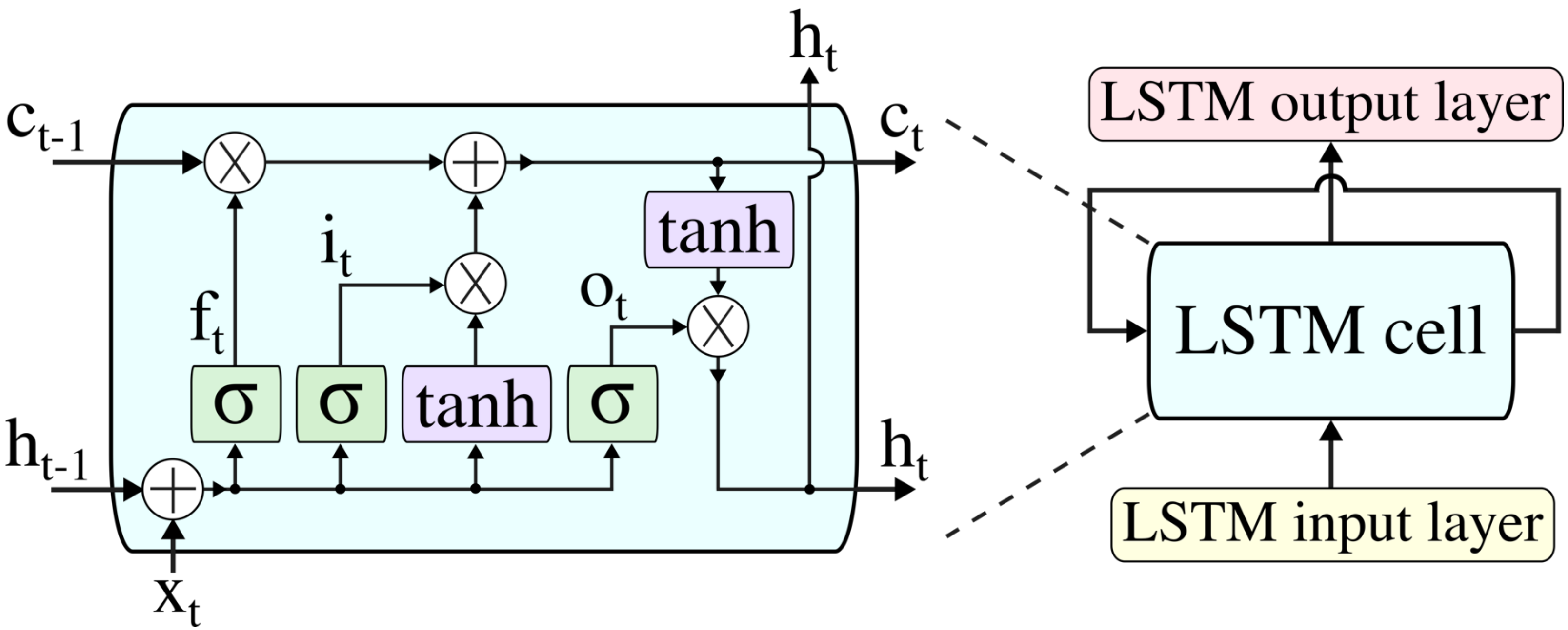
Deep Learning Neural Network

- 37 (BGP)/109 (NSL-KDD) RNNs, 80 FC_1, 32 FC_2, and 16 FC_3 fully connected (FC) hidden nodes:



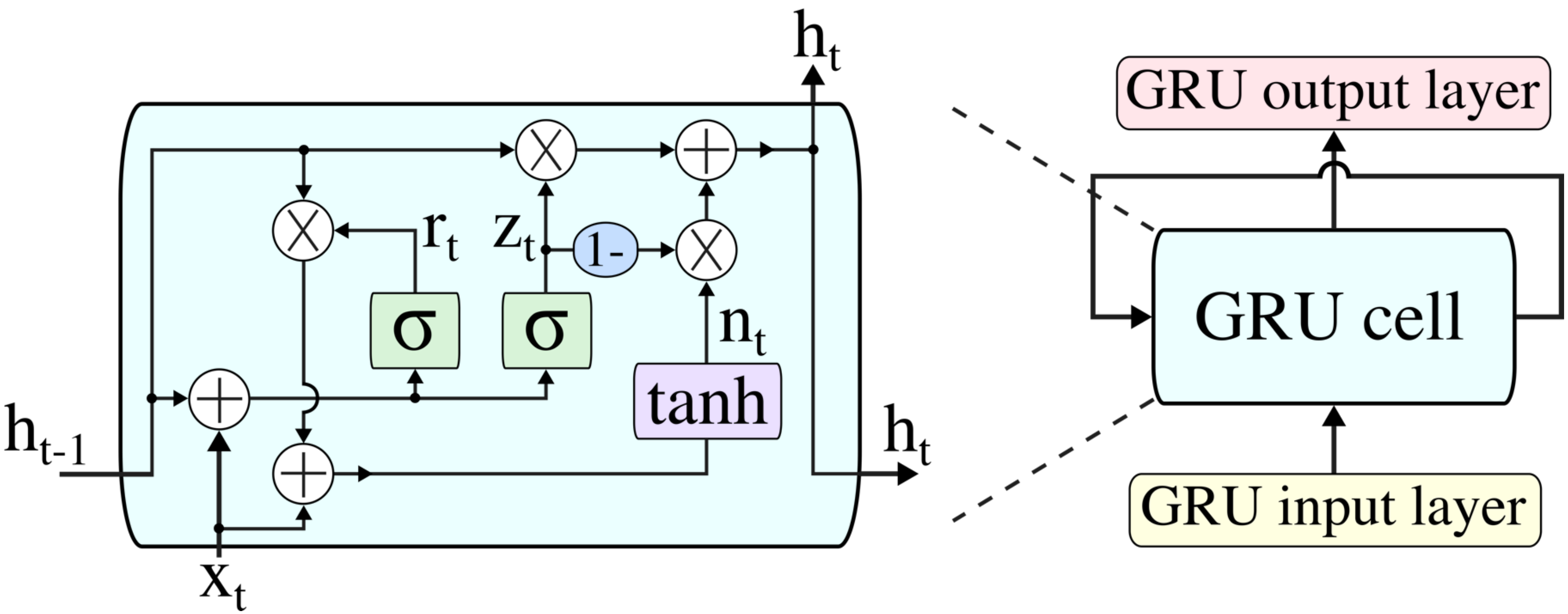
Long Short-Term Memory

- Repeating module for the Long Short-Term Memory (LSTM) neural network:



Gated Recurrent Unit

- Repeating module for the Gated Recurrent Unit (GRU) neural network:

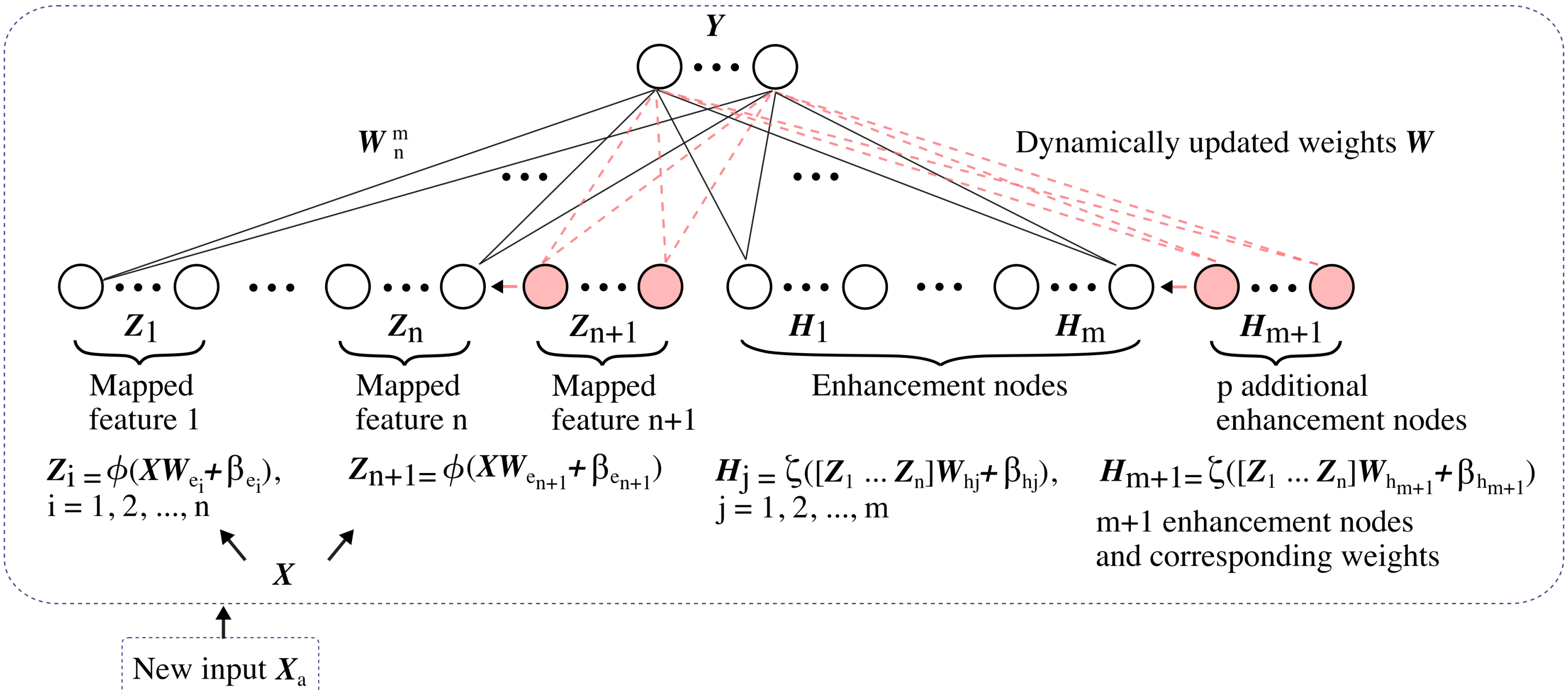


Roadmap

- Introduction
- Data processing
- Machine learning models:
  - Deep learning: multi-layer recurrent neural networks
  - Broad learning system
- Experimental procedure
- Performance evaluation
- Conclusions and references

Broad Learning System

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



Original BLS

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

$$\begin{aligned} A_x &= [\boldsymbol{Z}^n \mid \boldsymbol{H}^m] \\ &= \left[\phi\left(\boldsymbol{X}\boldsymbol{W}_{e_i} + \beta_{e_i}\right) \mid \xi\left(\boldsymbol{Z}_x^n \boldsymbol{W}_{h_j} + \beta_{h_j}\right)\right], \end{aligned}$$

where: $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, m$

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

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

Original BLS

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

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

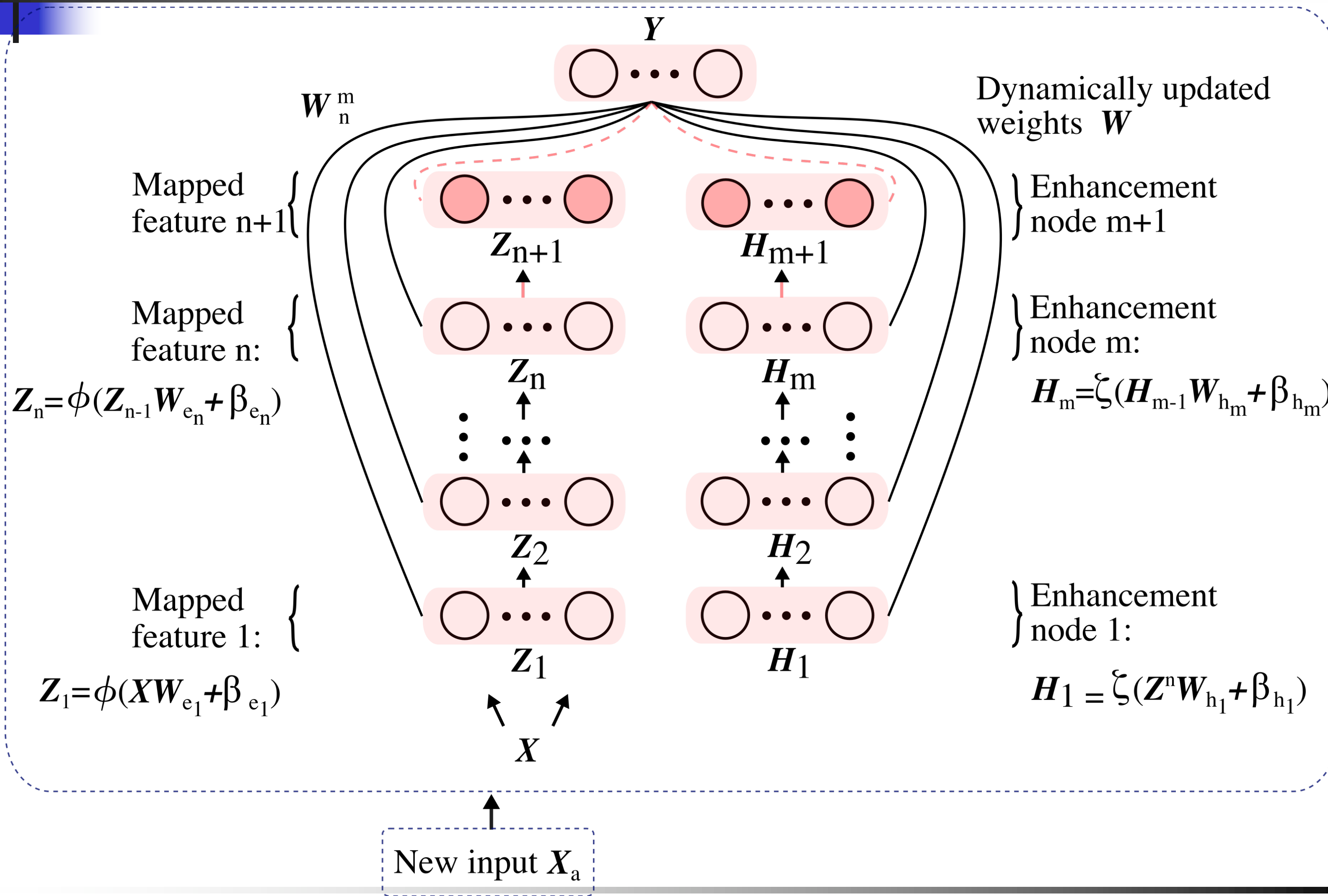
- During the training process, data labels are deduced using the calculated weights $\boldsymbol{W}_n^m$, mapped features $\boldsymbol{Z}_n$, and enhancement nodes $\boldsymbol{H}_m$:

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

BLS Extensions

- Radial Basis Function with Gaussian kernel and BLS: RBF-BLS
- Cascades of Mapped Features: CFBLS
- Cascades of Enhancement Nodes: CEBLS
- Cascades with Incremental Learning: CFEBLS

Cascades with Incremental Learning

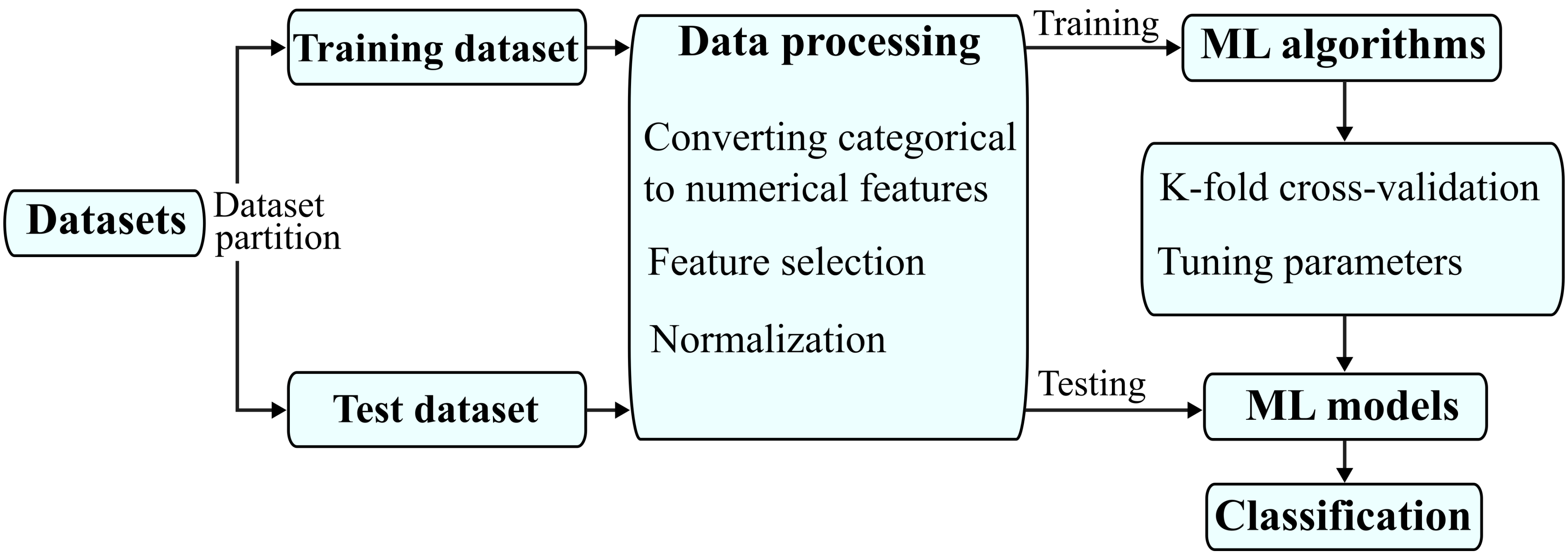


Roadmap

- Introduction
- Data processing:
- Machine learning models:
- Experimental procedure
- Performance evaluation
- Conclusions and references

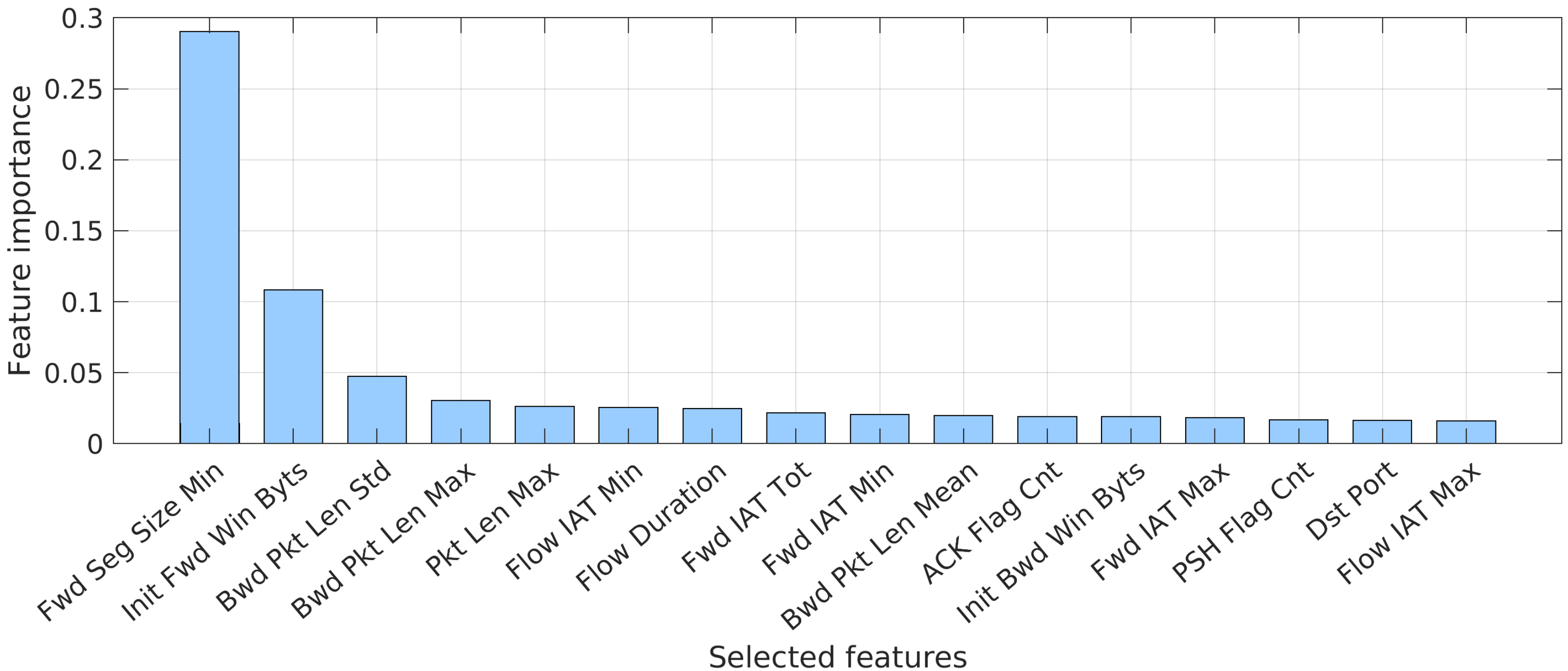
Intrusion Detection System

- Architecture:



Most Relevant Features

- CSE-CIC-IDS2018: 16 most relevant features



Number Training Parameters: BLS

Parameters	**CICIDS2017**			**CSE-CIC-IDS2018**		
	Number of features					
BLS	**78**	**64**	**32**	**78**	**64**	**32**
Model	RBF-BLS	BLS	CEBLS	CFBLS	RBF-BLS	CEBLS
Mapped features	20	10	10	20	20	15
Groups of mapped features	30	30	10	10	10	20
Enhancement nodes	40	20	40	80	80	80

Number of Training Parameters: Incremental BLS

Parameters	CICIDS2017			CSE-CIC-IDS2018		
	Number of features					
Incremental BLS	**78**	**64**	**32**	**78**	**64**	**32**
Model	CFBLS	CFEBLS	CEBLS	BLS	CEBLS	BLS
Mapped features	10	20	10	15	20	10
Groups of mapped features	20	20	20	30	10	20
Enhancement nodes	40	20	40	20	40	20
Incremental learning steps	2	2	2	2	2	2
Data points/step	55,680	55,680	55,680	49,320	49,320	49,320
Enhancement nodes/step	20	20	20	20	20	20

Roadmap

- Introduction
- Data processing:
  - BGP datasets
  - NSL-KDD dataset
- Machine learning models:
  - Deep learning: multi-layer recurrent neural networks
  - Broad learning system
- Experimental procedure
- Performance evaluation
- Conclusions and references

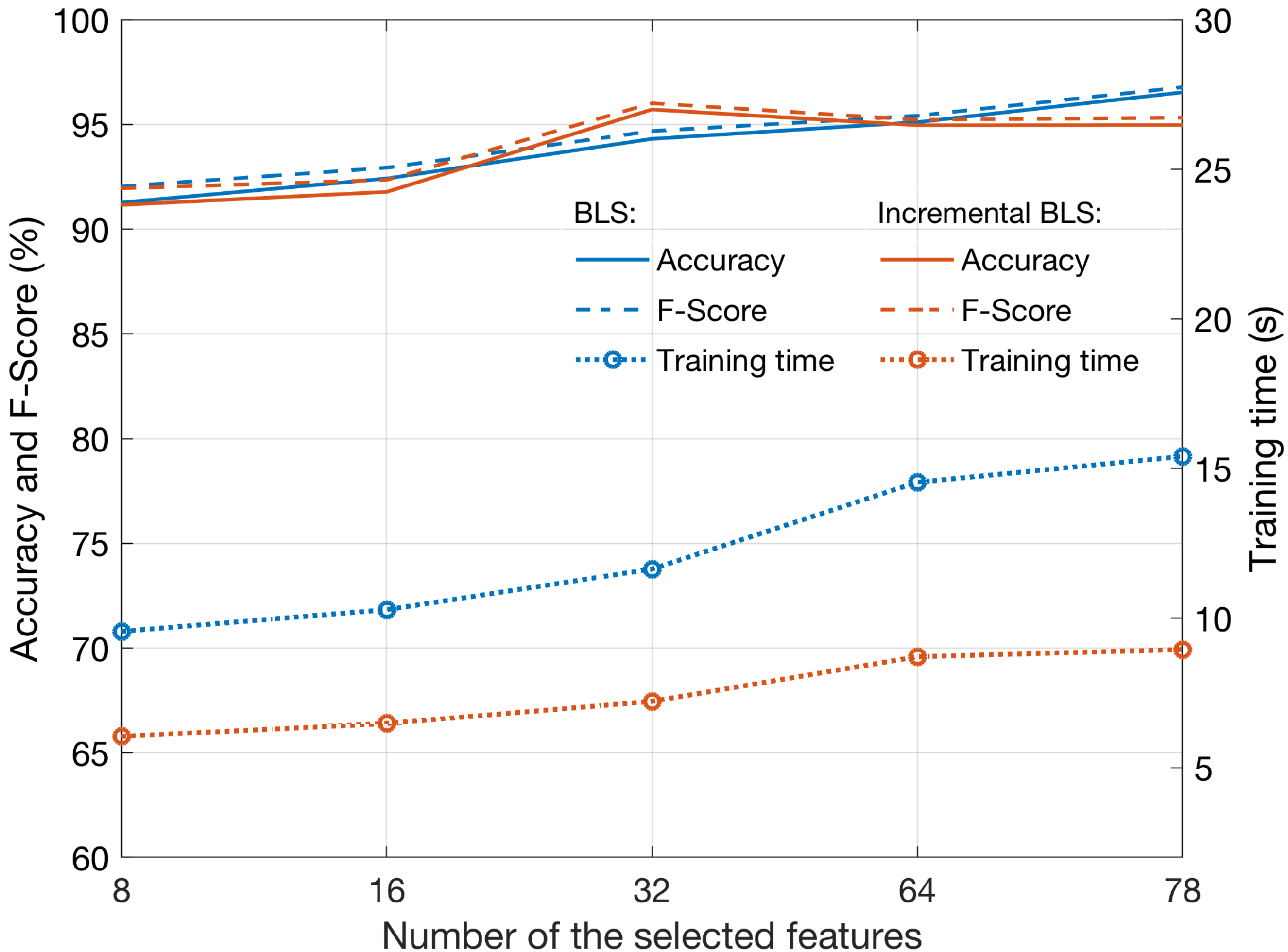
BLS Model:
CICIDS2017 and CSE-CIC-IDS2018 Datasets

Number of features	Dataset	Accuracy (%)	F-Score (%)	Model	Training time (s)
BLS					
78	CICIDS2017	96.63	96.87	RBF-BLS	15.60
	CSE-CIC-IDS2018	97.46	81.46	CFBLS	4.13
64	CICIDS2017	96.10	96.35	BLS	8.97
	CSE-CIC-IDS2018	98.60	90.49	RBF-BLS	4.65
32	CICIDS2017	96.34	96.62	CEBLS	39.25
	CSE-CIC-IDS2018	98.83	92.26	CEBLS	33.46

Incremental BLS Model:
CICIDS2017 and CSE-CIC-IDS2018 Datasets

Number of features	Dataset	Accuracy (%)	F-Score (%)	Model	Training time (s)
Incremental BLS					
78	CICIDS2017	95.12	95.44	CFBLS	3.69
	CSE-CIC-IDS2018	97.47	81.35	BLS	6.78
64	CICIDS2017	94.44	95.38	CFBLS	7.39
	CSE-CIC-IDS2018	96.70	74.64	CEBLS	11.59
32	CICIDS2017	95.39	95.75	BLS	6.39
	CSE-CIC-IDS2018	97.08	77.89	BLS	5.65

Performance: BLS and Incremental BLS, CICIDS2017



Roadmap

- Introduction
- Data processing:
- Machine learning models:
- Experimental procedure
- Performance evaluation
- Conclusions and references

Conclusions

- We evaluated performance of:
  - LSTM and GRU deep recurrent neural networks with a variable number of hidden layers
  - BLS models that employ radial basis function (RBF), cascades of mapped features and enhancement nodes, and incremental learning
- BLS and cascade combinations of mapped features and enhancement nodes achieved comparable performance and shorter training time because of their wide and deep structure.

Conclusions

- BLS models:
  - consist of a small number of hidden layers and adjust weights using pseudoinverse instead of back-propagation
  - dynamically update weights in case of incremental learning
  - better optimized weights due to additional data points for large datasets (NSL-KDD)
- While increasing the number of mapped features and enhancement nodes as well as mapped groups led to better performance, it required additional memory and training time.

Roadmap

- Introduction
- Data processing:
- Machine learning algorithms:
- Experimental procedure
- Performance evaluation
- Conclusions and references

References: Datasets

- BCNET :
  http://www.bc.net/
- RIPE RIS raw data:
  https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris
- 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

References: Broad Learning System

- Z. Liu and C. L. P. Chen, “Broad learning system: structural extensions on single-layer and multi-layer neural networks,” in *Proc. 2017 Int. Conf. Secur., Pattern Anal., Cybern.*, Shenzhen, China, Dec. 2017, pp. 136–141.
- 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.

Publications: http://www.sfu.ca/~ljilja

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.