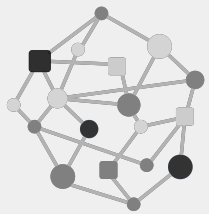
A Method for Network Intrusion Detection Using Flow Sequence and BERT Framework

Loc Gia Nguyen Kohei Watabe

Nagaoka University of Technology

May 30, 2023



ComNets Lab.

Motivation

- **Concept drift** in flow features from different environments (domains) made it difficult for machine learning model to identify flows.
- **Domain adaptation capability** describe the ability of an NIDS to identify flows from different environments.

Goal

Improve the domain adaptation capability of Network Intrusion Detection Systems.

Related Work: Energy-based Flow Classifier (EFC)[1]

[1] Pontes et al., "A New Method for Flow-Based Network Intrusion Detection Using the Inverse Potts Model".

Flow-based NIDS

A network traffic flow is

- a sequence of packets carrying information between two hosts that shares common properties, such as the 5-tuple: Src IP, Src Pt, Dst IP, Dst Pt, Proto.

Flow-based NIDS is

- a system that utilized the properties of network traffic flows to determined if they represent anomalous activity.

Transformers (1/2)

Bidirectional Encoder Representations from Transformers (BERT) is a language model that utilizes Transformer Encoders and Masked Language Modeling (MLM) task pre-training.

In MLM task, 15% of tokens were selected for prediction, and the training objective was to predict the selected token given its context. The selected token is

- replaced with a [MASK] token with probability 80%,
- replaced with a random word token with probability 10%,
- not replaced with probability 10%.

Transformers (2/2)

To identify anomalous flows, BERT is then fine-tuned with Named Entity Recognition (NER) task.

NER is structured as:

- taking an unannotated block of text (a sequence of flows).
- producing an annotated block of text that highlights the names of entities (marks flows as anomalous).

Hypotheses (1/2)

- **Features within a flow** underwent major shift in their distributions between different environments.
- **Context** are flows that appear at roughly the same time at the target flow.
- **A sequence of flows** is the collection of the target flow and its context.
- **Relation information** between each flows can be infer from the sequence.

Hypotheses (2/2)

Duration	Proto	Src Pt	Dst Pt	Packets	Bytes	Flags	Class
9.555	**TCP**	**54731**	**22**	**15**	**2163**	**.AP.SF**	**suspicious**
9.555	**TCP**	**22**	**54731**	**19**	**3185**	**.AP.SF**	**suspicious**
10.412	**TCP**	**22**	**57489**	**19**	**3185**	**.AP.SF**	**suspicious**
10.412	**TCP**	**57489**	**22**	**15**	**2163**	**.AP.SF**	**suspicious**
0	UDP	56475	19	1	46	.....	suspicious
0	ICMP	0	3.3	1	57	.....	suspicious
0.077	TCP	8000	52253	7	702	.AP.SF	normal
0.077	TCP	52253	8000	6	586	.AP.SF	normal
526.089	**TCP**	**22**	**59862**	**178**	**24253**	**.AP.SF**	**normal**
526.089	**TCP**	**59862**	**22**	**180**	**13471**	**.AP.SF**	**normal**
0	TCP	53213	23	1	46	....S.	suspicious
0	TCP	23	53213	1	40	.A.R..	suspicious

Figure: Two flow sequences. Though the flows in bold share the same properties, in sequence one they are benign, in sequence two they are anomalous.

Why use Transformer (BERT)?

and not RNN (LSTM)?

- Flows are not captured in perfect chronological order.
- RNN relies on the state of the previous input.
- Transformer process the sequence as a whole, small changes in the ordering of flow will probably not impact its performance.

Proposal (1/4)

Overview

Pre-processing

- **discretize** and tokenize features.

Model

- **BERT**

Training

- Pre-training with **Masked Language Modeling** task
- Fine-tuning with **Named-entity Recognition** task

Proposal (2/4)

Pre-processing

- Perform data binning, where each bin contains similar number of values
- Each bin is represented as a token

Feature	Upper limit of each bin
Duration	0.001, 0.002, 0.003, 0.004, 0.005, 0.006, 0.01, 0.04, 1, 10, 100, ∞
Protocol	TCP, UDP, GRE, ICMP, IGMP
Src Pt	50, 60, 100, 400, 500, 40000, 60000, ∞
Dst Pt	50, 60, 100, 400, 500, 40000, 60000, ∞
Bytes	50, 60, 70, 90, 100, 110, 200, 300, 400, 500, 700, 1000, 5000, ∞
Packets	2, 3, 4, 5, 6, 7, 10, 20, ∞
Flags	$\{(f_0, f_1, f_2, f_3, f_4, f_5) \mid f_i \in \{0, 1\}\}$

Figure: Feature binning upper thresholds

Proposal (3/4)

Training

Pre-training

- Train with Masked Language Modeling task
- Train with only benign flows
- Weights of the output layer is shared with the embedding layer
- The output is the probability of tokens

Fine-tuning

- Train with Named Entity Recognition task
- Train with benign and anomalous flows
- Output layer is an MLP with two output neuron
- The output is the probability of being benign and anomalous

Proposal (4/4)

Inference

The fine-tuned model is used as an NIDS

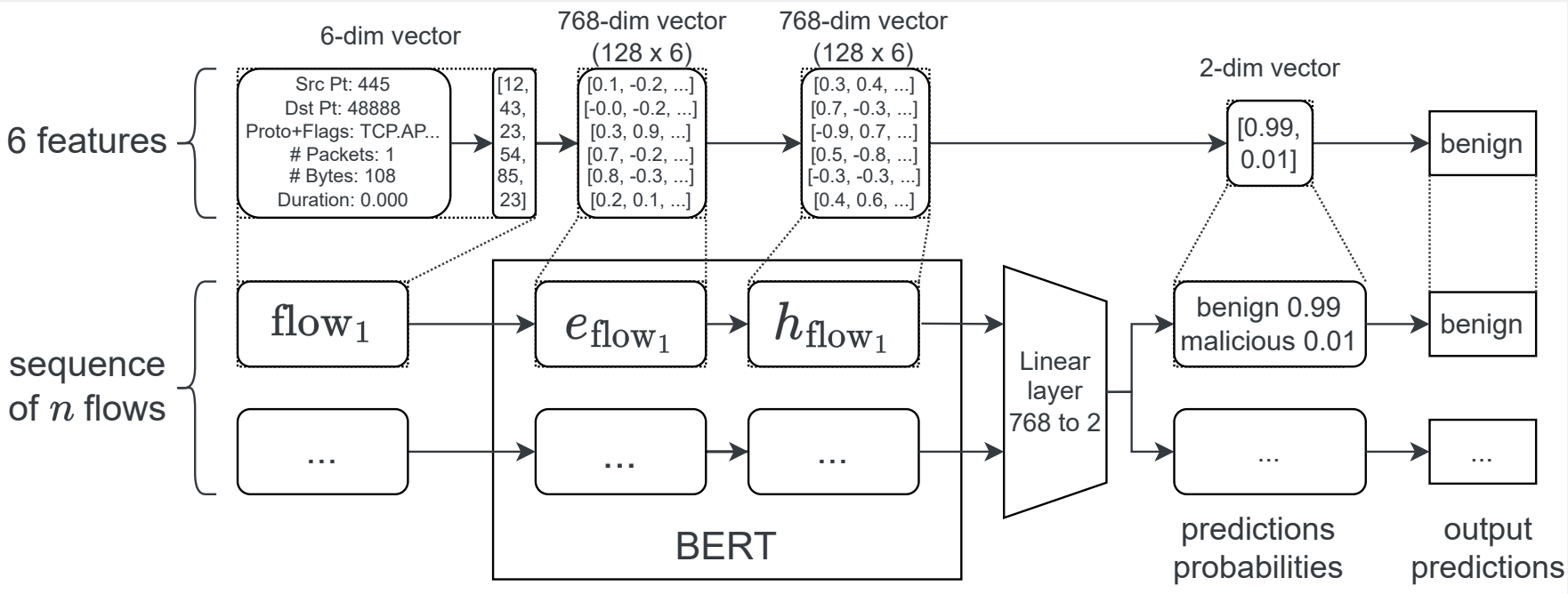


Figure: Proposed method model and data flow

Data sets (1/2)

CIDDS-001 and CIDDS-002 intrusion detection benchmark data sets are used for training and evaluation

#	Name	Description
1	Date first seen	Start time flow first seen
2	Duration	Duration of the flow
3	Proto	Transport Protocol (e.g. ICMP, TCP, or UDP)
4	Src IP	Source IP Address
5	Src Pt	Source Port
6	Dst IP	Destination IP Address
7	Dst Pt	Destination Port
8	Packets	Number of transmitted packets
9	Bytes	Number of transmitted bytes
10	Flags	OR concatenation of all TCP Flags

Figure: CIDDS features, Date first seen, Src IP, Dst IP discarded

Data sets (2/2)

	CIDDS-001 internal	CIDDS-001 balanced	CIDDS-001 external	CIDDS-002
Benign	28051906	3236027	212163	15598543
Malicious	3236027	3236027	459078	562640
Total	31287933	6472054	671241	16161183

Figure: Training data: 4 data sets from 3 network environments

- CIDDS-001 balanced is derived from CIDDS-001 internal by subsampling the benign flows
- Due to subsampling, the distribution of flows in contexts are different from the original

Experiment Setup

BERT configuration: 1 encoder, 1 attention head, hidden size 768

Training parameters:

- batch size 512, sequence length 128
- Optimizer: Adam, learning rate 1e−5
- Criterion: CrossEntropyLoss
- Mask Language Modeling 400 iterations, Named Entity Recognition 1500 iterations

Comparison targets:

- Energy-based Classifier (EFC), Decision Tree (DT), K-Nearest Neighbors (KNN), Multilayer Perceptron (MLP), Naive Bayes (NB), Support Vector Machine (LinSVM), AdaBoost (AB), Random Forest (RF)

Performance Evaluation (1/2)

- Our proposal exhibit better Accuracy and F1-score compare to classical machine learning techniques.
- Training on CIDDS-001 result in better performance in comparison to CIDDS-001 balanced

	Train CIDDS-001 internal Test CIDDS-001 external			
Classifier	Accuracy	F1-score	Recall	Precision
Proposal	**0.9078**	**0.9311**	0.9120	0.9511
EFC	0.8659	0.9044	**0.9278**	0.8822
DT	0.8491	0.8800	0.8088	0.9649
KNN	0.7978	0.8270	0.7067	**0.9967**
LinSVM	0.6784	0.6940	0.5333	0.9933
MLP	0.4380	0.3254	0.1982	0.9087
NB	0.3161	0.0000	0.0000	0.9937
AB	0.4606	0.3812	0.2430	0.8845
RF	0.8177	0.8510	0.7613	0.9647

	Train CIDDS-001 balanced Test CIDDS-001 external			
Classifier	Accuracy	F1-score	Recall	Precision
Proposal	**0.8581**	**0.9002**	**0.9358**	0.8673
EFC	0.8566	0.8985	0.9278	0.8710
DT	0.8496	0.8805	0.8098	0.9646
KNN	0.8128	0.8434	0.7374	**0.9850**
LinSVM	0.8123	0.8631	0.8650	0.8612
MLP	0.6671	0.6910	0.5441	0.9463
NB	0.2279	0.0282	0.0164	0.1013
AB	0.4742	0.4268	0.2862	0.8386
RF	0.8357	0.8679	0.7893	0.9639

Figure: Test results on CIDDS-001 external

Performance Evaluation (2/2)

- Similar results are achieved on CIDDS-002
- Changing the distribution of flows in a context greatly affect anomaly detection performance

	Train CIDDS-001 internal Test CIDDS-002			
Classifier	Accuracy	F1-score	Recall	Precision
Proposal	**0.9913**	**0.8578**	**0.7531**	**0.9962**
EFC	0.9084	0.3317	0.6534	0.2223
DT	0.9880	0.7948	0.6655	0.9864
KNN	0.9879	0.7924	0.6656	0.9789
LinSVM	0.9503	0.1042	0.0830	0.1400
MLP	0.9867	0.7722	0.6479	0.9554
NB	0.9638	0.0006	0.0003	0.0072
AB	0.9837	0.7148	0.5873	0.9131
RF	0.9881	0.7961	0.6670	0.9871

	Train CIDDS-001 balanced Test CIDDS-002			
Classifier	Accuracy	F1-score	Recall	Precision
Proposal	0.9870	0.7717	0.6315	**0.9921**
EFC	0.9088	0.3327	0.6534	0.2232
DT	0.9874	0.7871	0.6676	0.9586
KNN	0.9874	0.7867	0.6698	0.9532
LinSVM	0.5388	0.0260	0.1767	0.0140
MLP	0.9865	0.7832	**0.7021**	0.8854
NB	0.7908	0.0007	0.0021	0.0004
AB	0.9817	0.7170	0.6653	0.7774
RF	**0.9875**	**0.7891**	0.6692	0.9613

Figure: Test results on CIDDS-002

Conclusion and Future Work

- **Better domain adaptation capability** is observed for the proposed model over traditional ML approaches.
- **There is contextual information in flow data.** Classification performance of the proposed model degrade when flows are shuffle during training.
- Using the full transformer architecture is **very resource intensive** for an NIDS.

Future Work

- **Statistical approach**, using unlabeled, score-based classification.
- **Lightweight mechanism** to summarize information from flow sequence.