InterpretableSAD: Interpretable Anomaly Detection in Sequential Log Data

Xiao Han[1], He Cheng[1], Depeng Xu[2], and Shuhan Yuan[1]
[1]Utah State University
[2]University of Arkansas

UtahState University

Outline

- Background
  - Anomaly Detection
  - System Logs
  - Challenges
- Preliminary
- Method： InterpretableSAD
- Experiments
- Conclusion

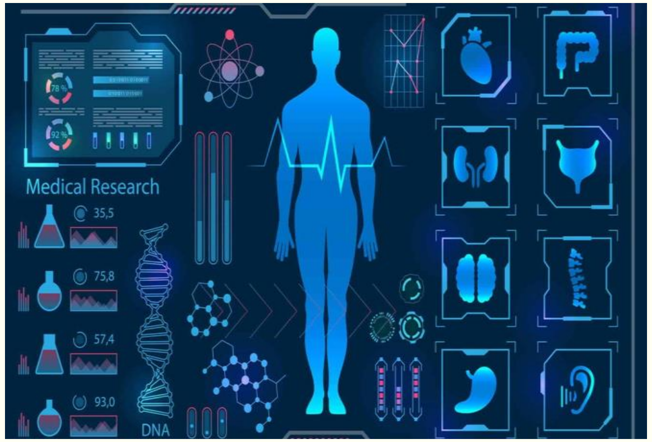
What is Anomaly Detection

- Anomaly detection in sequential data aims to identify sequences that deviate from the expected behavior or patterns.
- Anomaly detection receives much attention due to its broad applications.

Credit Card Fraud

Cyber Intrusions

Medical Diagnostics

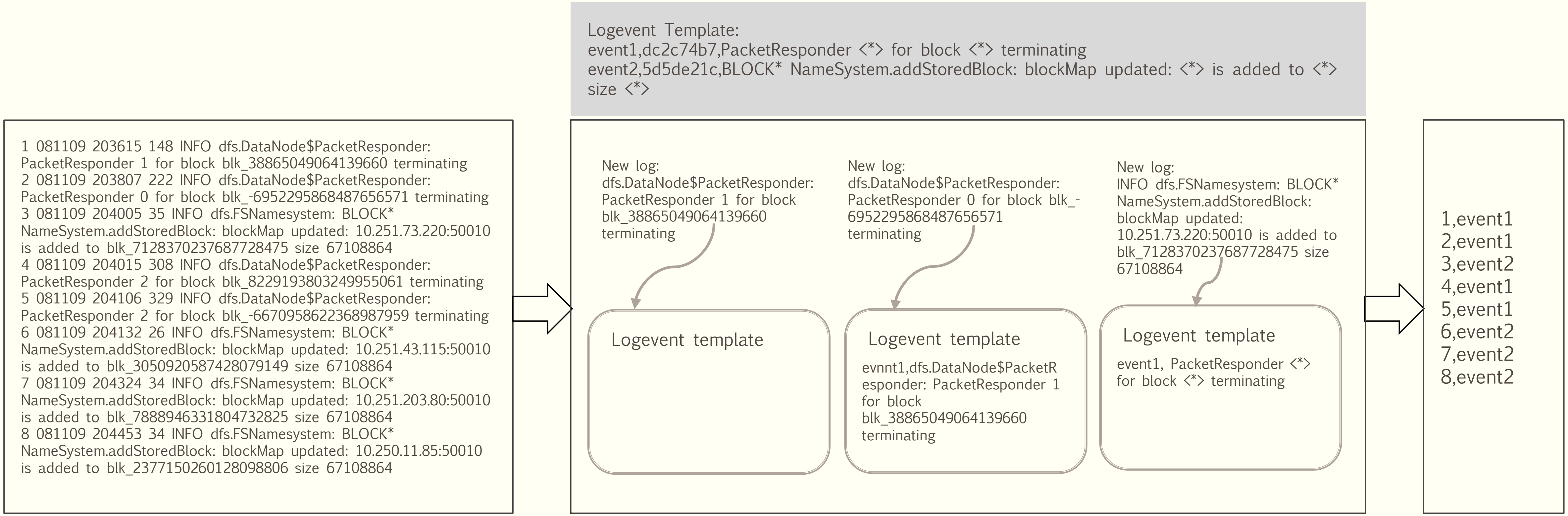


- Log anomaly detection uses system logs to detect anomalous events or patterns in computer systems.

Images from: https://traderdefenseadvisory.com/financial-protection/credit-card-fraud/
https://www.vodigynetworks.com/how-intrusion-prevention-systems-sniff-out-sneaky-cyber-attacks/
https://www.trademed.com/news/294779083/AI-Paving-the-Way-for-New-Generation-of-Medical-Diagnostic-Devices.html

Log Preprocessing

- Logs are generated by logging statements (print, logging.info)
- A log message (log entry) records timestamp, PID, level, and content
- Logs are unstructured data



Raw Data

Log Parsing

Goals

- To detect anomalous sequences as well as anomalous events in the sequences (interpretability).
- To train a binary classifier for the sequence-level log anomaly detection, we propose a novel negative sampling strategy to generate potential anomalous samples.
- To achieve anomalous event detection, we novelty apply a model interpretation approach, Integrated Gradients (IG).
- As IG relies on an appropriate baseline input for feature attributions, we further propose a novel baseline generation algorithm for log anomaly detection.

Outline

- Background
- Preliminary
  - Anomaly Detection in Sequential Log Data
  - Data Augmentation
  - Interpretable Machine Learning
- Method： InterpretableSAD
- Experiments
- Conclusion

Anomaly Detection in Sequential Log Data

- Traditional supervised learning -> Require an enormous number of labeled data
  - Logistic regression, decision tree, and SVM
- Traditional unsupervised learning -> Hard to capture the order information of sequence data
  - PCA, Isolation forest, and OC-SVM
- Deep learning -> No detailed information on the sub-sequence level
  - DeepLog and LogAnomaly

Data Augmentation

- Data augmentation technique is to tackle the scarcity of labeled data issue by artificially expanding the labeled dataset.

- Extensively used in image classification and natural language processing.
    - Rotation and flip for image data, synonym replacement for text data

- Negative sampling is a special data augmentation technique.

Interpretable Machine Learning

- Interpretable machine learning aims at providing a human understandable explanation about the decisions.
- The interpretable anomaly detection models are very limited.
- The attention mechanism provides an attention score that is more about the correlation among events instead of the correlation between events and the label.

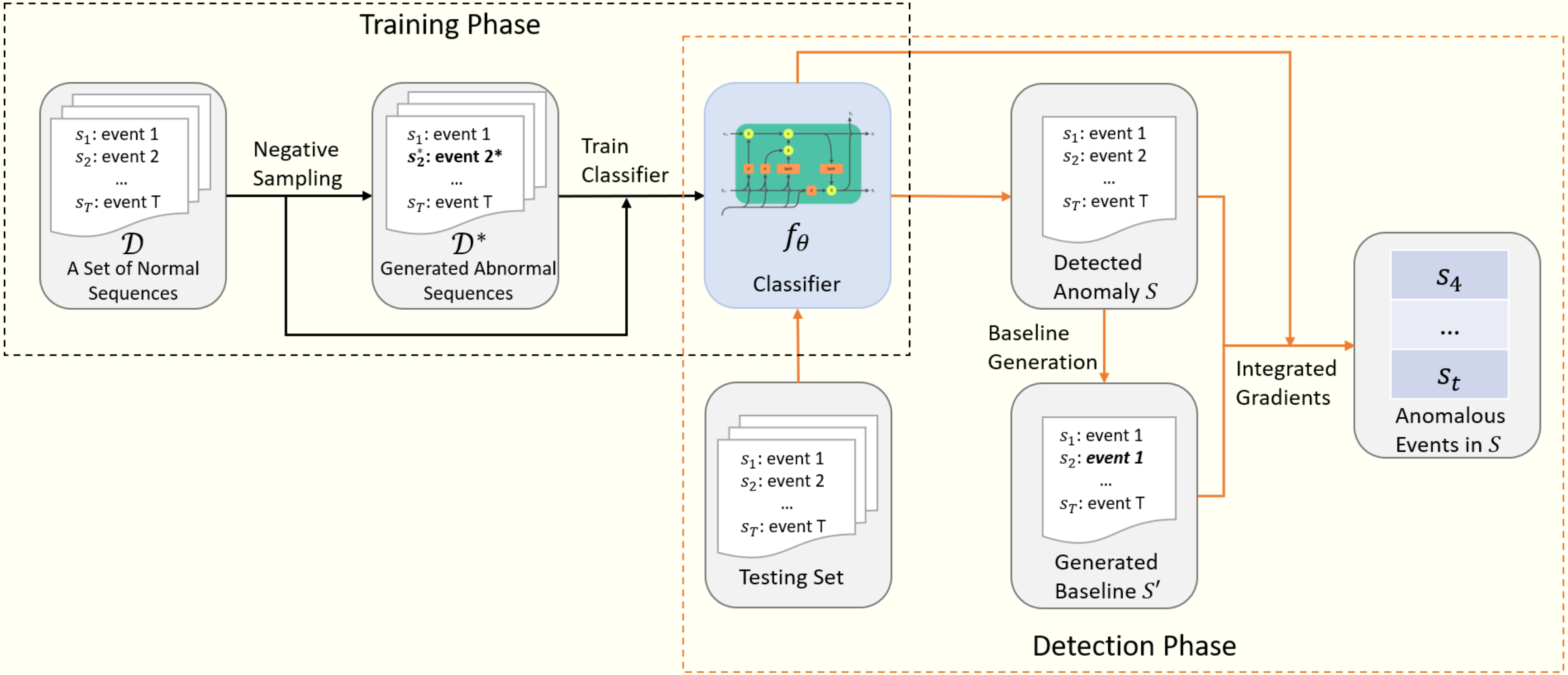
Outline

- Background
- Preliminary
- Method： InterpretableSAD
  - Data Augmentation via Negative Sampling
  - Anomaly Detection at a Sequence Level
  - Anomalous Event Detection via Integrated Gradients
- Experiments
- Conclusion

Problem Statement

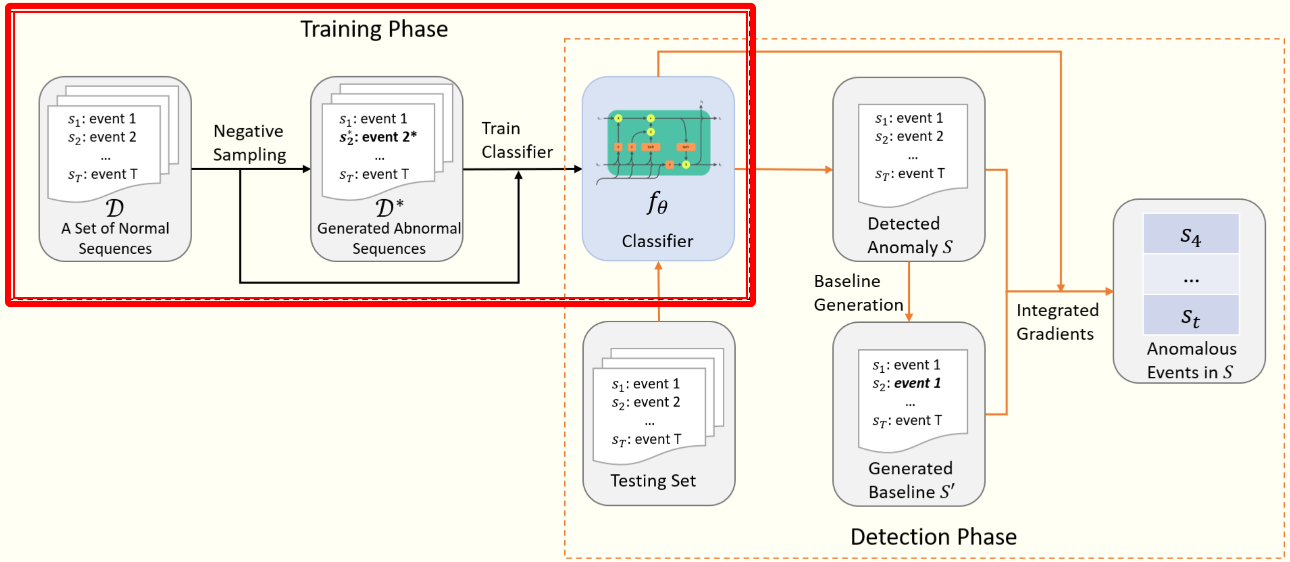
- Consider a log sequence of discrete events $S = \{s_1, \ldots, s_t, \ldots, s_T\}$, where $s_t \in \mathcal{E}$ indicates the event at the t-th position, and $\mathcal{E}$ is a set of unique events.
  - Sequence-level detection phase: predicting whether a log sequence S is anomalous based on a training dataset $\mathcal{D} = \{S^i\}_{i=1}^{N}$ that consists of only normal sequences.
  - Event-level detection phase: identifying anomalous events in the sequence.

Framework of InterpretableSAD



Training Phase – Negative Sampling

- In order to train an accurate binary classifier, we aim to generate a dataset $\mathcal{D}^*$ with sufficient anomalous samples that can cover common anomalous scenarios.



Training Phase – Negative Sampling Cont.

- Two anomalous scenarios for anomalous log sequence generation:
  1. Anomalous events in the sequences
  2. Regular events happen in an unusual context

Algorithm 1: Negative Sampling

Input : Training set $\mathcal{D}$, Negative sample size M

Output: Negative sample set $\mathcal{D}^*$

Generate a bigram event dictionary $\mathcal{B}$ based on $\mathcal{D}$

for $i = 0$ **to** M **do**

  Randomly select S from $\mathcal{D}$

  $ind \leftarrow$ Randomly select r indices of events from S

  for t *in* ind **do**

    $(s_t, s^*_{t+1}) \leftarrow$ randomly select or generate a rare or never observed bigram in $\mathcal{B}$

    $(s_t, s_{t+1}) \leftarrow (s_t, s^*_{t+1})$

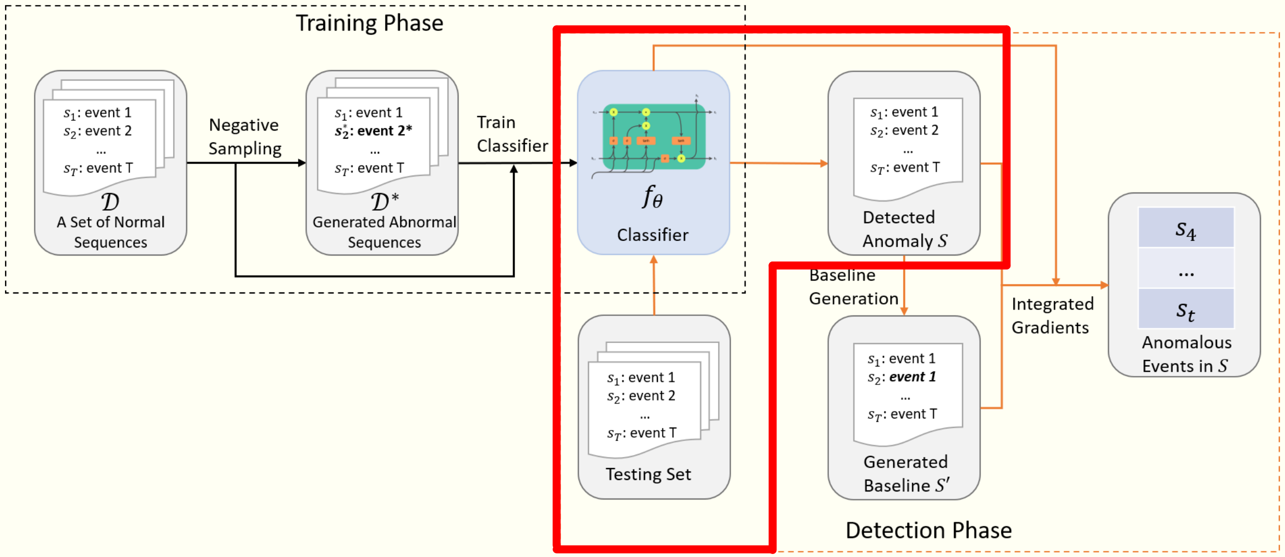
  $S^* \leftarrow S, \quad \mathcal{D}^* += S^*$

return $\mathcal{D}^*$

Detection Phase – Anomaly Detection on a Sequence Level

- After generating a set of anomalous sequences $\mathcal{D}^*$, we use both $\mathcal{D}$ and $\mathcal{D}^*$ to train a binary classification model $f : S \rightarrow [0,1]$.
- Specifically, we use an LSTM with a linear layer as the classification model f.
- We further adapt the cross-entropy loss to train the neural network:

$$\mathcal{L} = \sum_{j \in \mathcal{D}^* \cup \mathcal{D}} -y_j log\hat{y}_j - (1 - y_j)\log(1 - \hat{y}_j)$$



Detection Phase – Anomalous Event Detection

- Integrated Gradients (IG) is a model interpretable technique that can interpret prediction results by attributing input features.
- For example,

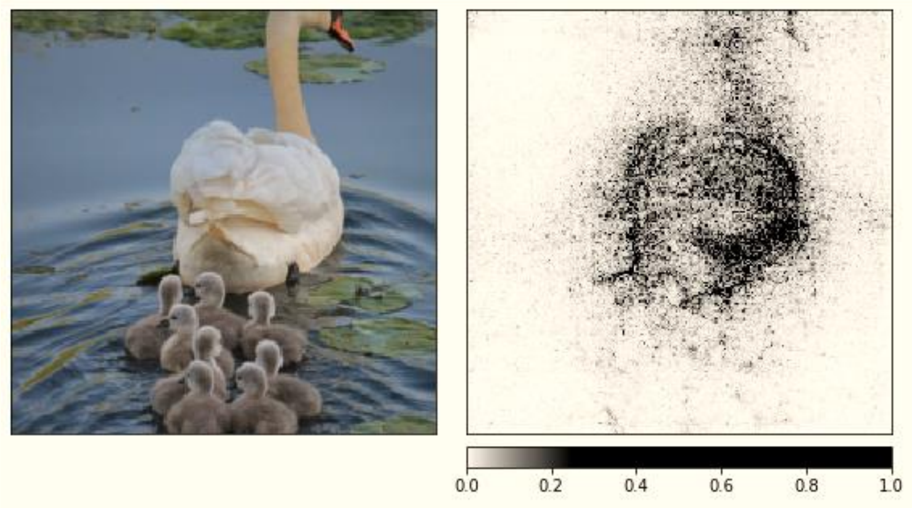
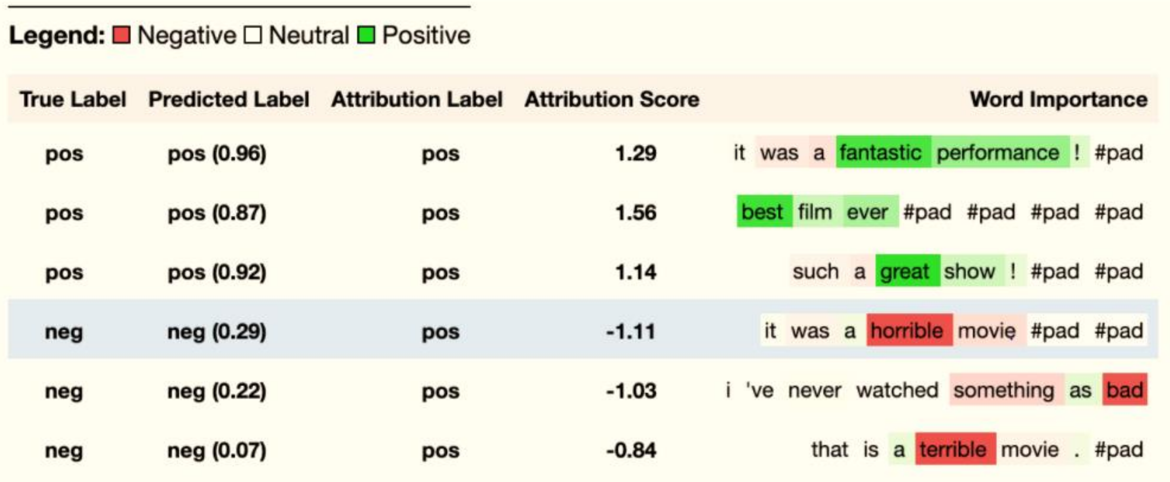


Image Classification



Legend: Negative Neutral Positive

True Label	Predicted Label	Attribution Label	Attribution Score	Word Importance
pos	pos (0.96)	pos	1.29	it was a fantastic performance ! #pad
pos	pos (0.87)	pos	1.56	best film ever #pad #pad #pad #pad
pos	pos (0.92)	pos	1.14	such a great show ! #pad #pad
neg	neg (0.29)	pos	-1.11	it was a horrible movie #pad #pad
neg	neg (0.22)	pos	-1.03	i 've never watched something as bad
neg	neg (0.07)	pos	-0.84	that is a terrible movie . #pad

Sentiment Analysis

Images from: https://captum.ai/tutorials/

Detection Phase – Anomalous Event Detection Cont.

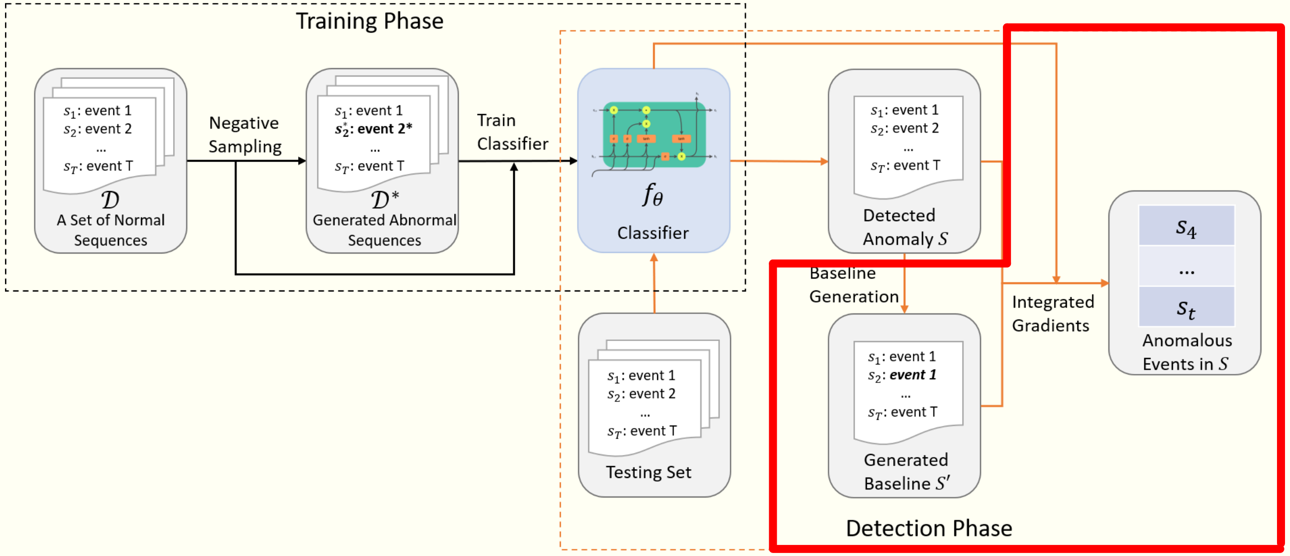
- Formally, given a neural network $f_\theta : S \rightarrow [0,1]$, integrated gradients are attributions of the prediction at input S relative to a baseline input S' as a vector $A_{f_\theta}(S, S') = (a_1, \ldots, a_T)$, where a_t is the contribution score of s_t to the prediction $f_\theta(S)$.

EventID	Score
09a53393	-0.20
09a53393	-0.20
3d91fa85	-0.51
09a53393	-0.20
0567184d	1.73
d38aa58d	-0.62
e3df2680	0.17
0567184d	1.73
d38aa58d	-0.62
e3df2680	0.17
…	
5d5de21c	0.00
…	
d63ef163	-0.34
…	
dba996ef	-1.00

09a53393	Receiving block <*> src: <*> dest: <*>
3d91fa85	BLOCK* NameSystem. allocateBlock: <*> <*>
0567184d	Receiving empty packet for block <*>
d38aa58d	PacketResponder <*> for block <*> <*>
e3df2680	Received block <*> of size <*> from <*>
5d5de21c	BLOCK* NameSystem. addStoredBlock: blockMap updated: <*> is added to <*> size <*>
d63ef163	BLOCK* NameSystem.delete: <*> is added to invalidSet of <*>
dba996ef	Deleting block <*> file <*>

Detection Phase – Baseline Generation

- Finding a reasonable baseline is an essential step for applying the IG method.
- For the image classification models, the black image is widely used as a baseline, while the zero-embedding matrix is a common baseline for the text classification task.
- Therefore, we propose to generate a unique baseline for each sequence.



Detection Phase – Baseline Generation Cont.

- Sort the events based on their frequencies in the training set
- Replace the lowest frequent event with its preceding event
- Check whether the sequence is normal

Algorithm 2: Baseline Generation

Input : Neural network f_θ, Anomalous sample S, Training set $\mathcal{D}$, Replacement Threshold τ

Output: Baseline S'

$i = 0$

while $f_\theta(S)$ *is not normal* & $i < \tau$ **do**

  $s_t \leftarrow$ Select the event in S with the lowest frequency based on $\mathcal{D}$

  $s_t \leftarrow s_{t-1}$, $i += 1$

$S' \leftarrow S$

return S'

Outline

- Background
- Preliminary
- Method： InterpretableSAD
- Experiments
  - Datasets
  - Baselines
  - Experimental Results
- Conclusion

Datasets

- Log parser – Drain; Window size – 100
- Training dataset consists of 100,000 normal log sequences and 2,000,000 generated anomalous sequences for each log dataset.

Dataset	# of Unique Log Keys	# of Log Sequences		# of Log Keys in Anomalous Sequences	
		Normal	Anomalous	Normal	Anomalous
HDFS	48 (19)	458,223	16,838	N/A	N/A
BGL	396 (318)	19,430	4,190	326,491	7,139
Thunderbird	806 (774)	22,538	76,189	6,866,417	479,883

TABLE I: Statistics of Test Datasets

Baselines for Anomalous Log Sequence Detection

- Traditional machine learning models:
  - Principal Component Analysis (PCA)
  - One-Class SVM (OCSVM)
  - Isolation Forest (iForest)
  - LogCluster
- Deep learning models:
  - DeepLog
  - LogAnomaly

Results on Anomalous Log Sequence Detection

Method	BGL			Thunderbird			HDFS		
	Precision	Recall	F-1 score	Precision	Recall	F-1 score	Precision	Recall	F-1 score
PCA	67.91	99.79	80.82	94.83	84.43	89.33	97.77	42.12	58.88
iForest	73.13	38.19	50.17	95.06	17.92	30.15	41.59	58.80	48.72
OCSVM	24.60	100	39.49	87.13	100	93.12	6.68	90.58	12.44
LogCluster	8.03	15.97	10.69	86.56	22.94	36.26	98.37	67.45	80.03
DeepLog	42.39	52.08	46.74	82.42	81.36	81.89	56.98	48.37	52.32
LogAnomaly	42.58	53.17	47.29	81.69	82.11	81.90	55.85	48.03	51.65
InterpretableSAD	94.25	88.47	**91.27**	97.31	96.42	**96.86**	92.31	87.04	**89.60**

Results on Anomalous Log Sequence Detection

Method	BGL			Thunderbird			HDFS		
	Precision	Recall	F-1 score	Precision	Recall	F-1 score	Precision	Recall	F-1 score
PCA	67.91	99.79	80.82	94.83	84.43	89.33	97.77	42.12	58.88
iForest	73.13	38.19	50.17	95.06	17.92	30.15	41.59	58.80	48.72
OCSVM	24.60	100	39.49	87.13	100	93.12	6.68	90.58	12.44
LogCluster	8.03	15.97	10.69	86.56	22.94	36.26	98.37	67.45	80.03
DeepLog	42.39	52.08	46.74	82.42	81.36	81.89	56.98	48.37	52.32
LogAnomaly	42.58	53.17	47.29	81.69	82.11	81.90	55.85	48.03	51.65
InterpretableSAD	94.25	88.47	**91.27**	97.31	96.42	**96.86**	92.31	87.04	**89.60**

Results on Anomalous Log Sequence Detection

Method	BGL			Thunderbird			HDFS		
	Precision	Recall	F-1 score	Precision	Recall	F-1 score	Precision	Recall	F-1 score
PCA	67.91	99.79	80.82	94.83	84.43	89.33	97.77	42.12	58.88
iForest	73.13	38.19	50.17	95.06	17.92	30.15	41.59	58.80	48.72
OCSVM	24.60	100	39.49	87.13	100	93.12	6.68	90.58	12.44
LogCluster	8.03	15.97	10.69	86.56	22.94	36.26	98.37	67.45	80.03
DeepLog	42.39	52.08	46.74	82.42	81.36	81.89	56.98	48.37	52.32
LogAnomaly	42.58	53.17	47.29	81.69	82.11	81.90	55.85	48.03	51.65
InterpretableSAD	94.25	88.47	**91.27**	97.31	96.42	**96.86**	92.31	87.04	**89.60**

Baselines for Anomalous Event Detection

- Anchors – A model-agnostic method that explains the behavior of complex models
- Low-Freq – Labeling low frequency events as anomalous
- Integrated Gradients without our IG baseline generation

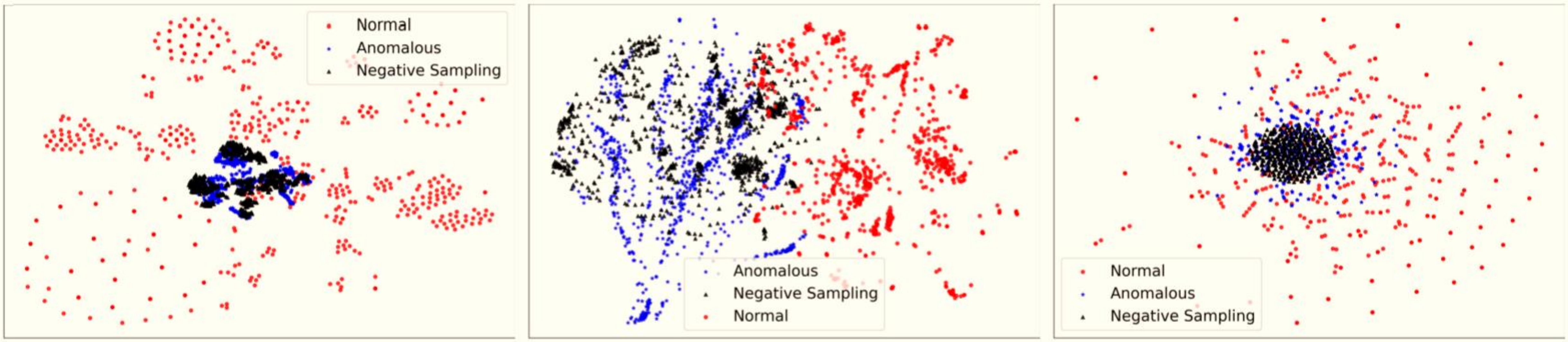
Results on Anomalous Event Detection

- We consider two scenarios, with or without a validation set consisting of 10% anomalous sequences in the testing datasets to tune a detection threshold for anomalous event detection.

Method	BGL			Thunderbird		
	Precision	Recall	F-1 score	Precision	Recall	F-1 score
Anchors	0.31	8.56	0.60	4.58	14.62	6.98
Low-Freq	38.76	93.59	54.82	52.61	99.00	68.70
IG w/o val	6.56	90.27	12.23	10.36	85.65	18.49
IG w/ val	42.43	73.83	53.89	20.92	44.48	28.45
InterpretableSAD w/o val	50.87	89.23	64.80	94.98	86.79	90.70
InterpretableSAD w/ val	68.92	82.53	**75.11**	93.84	98.31	**96.02**

Visualization of the Normal, Anomalous, and Generated Anomalous Sequences

- We randomly select 1,000 sequences of normal, anomalous, and generated samples, separately.



(a) BGL (b) Thunderbird (c) HDFS

Outline

- Background
- Preliminary
- Method： InterpretableSAD
- Experiment
- Conclusion

Conclusion

- Propose a novel framework to detect anomalous sequences as well as anomalous events in the sequences.
- Propose a novel negative sampling algorithm that can accurately generate anomalous samples.
- Apply an interpretable machine learning technique, Integrated Gradients (IG), to detect potential anomalous events.
- Experimental results on three log datasets show that our model can achieve state-of-the-art performance on the anomalous sequence and event detection.

Thank You For Your Attention!

This work was supported in part by NSF 1502273 and 2103829.