ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9414197

SEQUENTIAL ADVERSARIAL ANOMALY DETECTION WITH DEEP FOURIER KERNEL

Shixiang Zhu, Henry Shaowu Yuchi, Minghe Zhang, Yao Xie

Georgia Institute of Technology

ABSTRACT

We present a novel adversarial detector for the anomalous sequence when there are only one-class training samples. The detector is developed by finding the best detector that can discriminate against the worst-case, which statistically mimics the training sequences. We explicitly capture the dependence in sequential events using the marked point process with a deep Fourier kernel. The detector evaluates a test sequence and compares it with an optimal time-varying threshold, which is also learned from data. Using numerical experiments on simulations and real-world datasets, we demonstrate the superior performance of our proposed method.

***Index Terms*—** sequential anomaly detection, adversarial learning, Fourier kernel

1. INTRODUCTION

Spatio-temporal event data are ubiquitous in our daily lives, ranging from electronic transaction records, earthquake activities recorded by seismic sensors, and police reports. Such data consist of sequences of discrete events that indicate when and where each event occurred and other additional descriptions, such as its category or volume. In many scenarios, when an anomalous incident occurs, it may be followed by a series of anomalous events related to the incident.

Consider a motivating example of detecting credit card fraudulence at a department store. The events in this setting correspond to a sequence of unauthorized transactions. Each transaction record typically includes information on time, location, and transaction amount of the purchase. To stop frauds and prevent further losses for both consumers and retailers, identifying whether a sequence is an anomaly *as early as possible* has become an urgent need for the merchants. To protect consumer privacy, the merchants can not provide normal transaction data from customers, which creates a situation where only anomalous transaction data are available for developing algorithms. Such a "one-class" problem makes the task of building a fraud detector even more challenging.

There has been much research effort in machine learning and statistics for sequential anomaly detection [1–4]. However, most of the existing methods cannot be directly applied here for the following reasons. (1) Many existing works consider detecting anomalous sequences "as a whole" rather than detect in an online fashion, and decisions can not be made until the sequence has been fully observed. (2) The one-class data situation requires an unsupervised approach for anomaly detection. However, most sequential anomaly detection algorithms are based on supervised learning.

This paper presents an adversarial anomaly detection algorithm for one-class sequential detection, where only anomalous data are available. On a high-level, our adversarial anomaly detector is formulated as a minimax problem: the detector is optimized to detect the "worst-case" counterfeit sequences from a generator that maximally mimics the provided anomalous sequence data. The minimax formulation is inspired by imitation learning [5], which can be explained as minimizing the maximum mean discrepancy [6] (MMD). Our algorithm's notable feature is that our detector uses a *time-varying* threshold that is learned from data (also solved from the minimax problem), which provides a "tightest" control of the false-alarms. Such a time-varying threshold is hard to obtain precisely in theory. Here, we provide a data-driven computational approach crucial to achieving good performance as validated by our numerical experiments. The time-varying threshold learned sequentially from data is a drastic departure from prior approaches in sequential anomaly detection. In particular, we parametrize the detector as the likelihood function of marked Hawkes processes and present a novel deep Fourier kernel in the model. The resulted likelihood function is computationally efficient to carry out the online detection, and at the same time, allows the capture of complex dependence between events in anomalous sequences. We demonstrate the proposed method's superior performance on synthetic and real data for sequential anomaly detection by comparing state-of-the-art methods.

2. PROPOSED DETECTION FRAMEWORK

Now we focus on a setting where only anomalous sequences are available. We aim to develop a detector that can detect the anomalous sequence in an online fashion and raise the alarm as soon as possible. Denote such a detector as ℓ with parameter θ. At each time t, the detector evaluates a statistic and compares it with a threshold. For a length-N sequence $\boldsymbol{x}_{1:i} := [x_1, \dots, x_i]^\top$, $i = 1, 2, \dots, N$, the detector is a stopping rule: it stops and raises an alarm the first time that the detection statistic exceeds the threshold: $T = \inf\{t : \ell(\boldsymbol{x}_{1:i}; \theta) > \eta_i,\ t_i \leq t < t_{i+1}\}$. Once an alarm is raised, the sequence is flagged as an anomaly. If there is no alarm raised till the end of the time horizon, the sequence is considered normal.

Adversarial anomaly detection. Since normal sequences are not available, we introduce an *adversarial generator*, which produces "normal" sequences that are statistically similar to the real anomalous sequences. The detector has to discriminate the true anomalous sequence from the counterfeit "normal" sequences. We introduce competition between the anomaly detector, and the generator drives both models to improve their performances until anomalies can be distinguishable from counterfeits in the worst-case scenarios. Assume a set of anomalous sequences drawn from an empirical distribution π. Formally, we formulate this as a minimax problem as follows:

$$\min_{\varphi \in \mathcal{G}} \max_{\theta \in \Theta} J(\theta, \varphi) := \mathbb{E}_{\boldsymbol{x} \sim \pi}\ \ell(\boldsymbol{x}; \theta) - \mathbb{E}_{\boldsymbol{z} \sim G_z(\varphi)}\ \ell(\boldsymbol{z}; \theta), \quad (1)$$

where G_z is an adversarial generator specified by parameter $\varphi \in \mathcal{G}$ and $\mathcal{G}$ is a family of candidate generators. The adversarial generator is built upon the Long Short-Term Memory (LSTM) [7]; the output of our LSTM specifies the distribution rather than the exact occurrence (time and location) of the next event, which is expressive enough to simulate sequential data (see our arXiv paper[1]). Here the detection statistic corresponds to $\ell(\theta)$, the log-likelihood function of the sequence specified by $\theta \in \Theta$ and Θ is its parameter space. The choices of the adversarial generator and the detector will be further discussed in Section 3. The detector compares the detection statistic to a threshold. We define the following:

Definition 1 (Adversarial sequential anomaly detector)**.** *Denote the solution to the minimax problem* (1) *as* (θ^*, φ^*)*. A sequential adversarial detector raises an alarm at the time* i *if*

$$\ell(\boldsymbol{x}_{1:i}; \theta^*) > \eta_i^*, \quad (2)$$

where the time-varying threshold $\eta_i^* \propto \mathbb{E}_{\boldsymbol{z} \sim G_z(\varphi^*)}\ \ell(\boldsymbol{z}_{1:i}; \theta^*)$*.*

Time-varying threshold. Now we explain the choice of the time-varying threshold. Since the value of log-likelihood function $\ell(x_{1:i}; \theta^*)$ for partial sequence observation $\boldsymbol{x}_{1:i}$ may vary over the time step i (the i-the event is occurred), we need to adjust the threshold accordingly for making decisions as a function of i. Note that our time-varying threshold η_i^* is drastically different from statistical sequential analysis, where the threshold for performing detection is usually constant or pre-set (not adaptive to data) based on the known distributions of the data sequence. For instance, we can set the threshold growing over time as $\sqrt{t}$ [8]). The rationale behind the design of the threshold η_i^* is as follows. At any given time step, the log-likelihood of the data sequence is larger than that of the generated adversarial sequence; therefore, η_i^* provides the tight lower bound for the likelihood of anomalous sequences $\ell(\boldsymbol{x}; \theta^*)$ due to the minimization in (1). Formally, for any $\varphi \in \mathcal{G}$, $0 \leq \mathbb{E}_{\boldsymbol{x} \sim \pi}\ \ell(\boldsymbol{x}_{1:i}; \theta^*) - \eta_i^* \leq \mathbb{E}_{\boldsymbol{x} \sim \pi}\ \ell(\boldsymbol{x}_{1:i}; \theta^*) - \mathbb{E}_{\boldsymbol{z} \sim G_z(\varphi)}\ \ell(\boldsymbol{z}_{1:i}; \theta^*)$. The adversarial sequences drawn from $G_z(\varphi^*)$ can be viewed as the normal sequences that are statistically "closest" to anomalous sequences. Therefore, the log-likelihood of such sequences in the "worst-case" scenario defines the "border region" for detection. In practice, the threshold η_i^* can be estimated by $1/n' \sum_{l=1}^{n'} \ell(\boldsymbol{z}_{1:i}^l; \theta^*)$, where $\{\boldsymbol{z}^l\}_{l=1,\ldots,n'}$ are adversarial sequences sampled from $G_z(\varphi)$ and n' is the number of the sequences.

Connection to imitation learning. Our framework can be viewed as an instance of imitation learning [9]. The problem formulation (1) resembles the minimax formulation in inverse reinforcement learning (IRL) proposed by seminal works [9, 10]. We regard anomalous samples $\boldsymbol{x} \sim \pi$ as expert demonstrations sampled from the expert policy π. Each event $x_i, i = 1, \ldots, N$ of the sequence is analogous to the i-th action made by the expert given the history of past events $\{x_1, x_2, \ldots, x_{i-1}\}$ as the corresponding state. Accordingly, the generator can be regarded as a learner that generates convincing counterfeit trajectories. The log-likelihood of observed sequences can be interpreted as undiscounted *return*, i.e., the accumulated sum of rewards evaluated at past actions. The ultimate goal of the proposed framework (1) is to close the gap between the expert and the learner's returns so that the counterfeit trajectories can meet the lower bound of the real demonstrations.

Connection to MMD-like distance. The proposed approach can also be viewed as minimizing a maximum mean discrepancy (MMD)-like distance metric [6] as illustrated in Fig. 1. More specifically, the maximization in (1) is analogous to an MMD metric in a reduced function class specified by Θ, i.e., $\sup_{\theta \in \Theta} \mathbb{E}_{\boldsymbol{x} \sim \pi}\ \ell(\boldsymbol{x}; \theta) - \mathbb{E}_{\boldsymbol{z} \sim g}\ \ell(\boldsymbol{z}; \theta)$, where Θ may not necessarily be a space of continuous, bounded functions on sample space. As shown in [6], if Θ is sufficiently expressive (universal), e.g., the function class on reproducing kernel Hilbert space (RKHS), then maximization over such Θ is equivalent to the original definition. Based on this, we select a function class that serves our purpose for anomaly detection (characterizing the sequence's log-likelihood), which has enough expressive power for our purposes. Therefore, the problem defined in (1) can be regarded as minimizing such an MMD-like metric between the empirical distribution of anomalous sequences and the distribution of adversarial sequences. The minimal MMD distance corresponds to the best "detection ra-

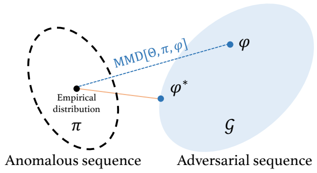

Fig. 1: The empirical distribution of anomalous sequences is π. The assumed family of candidate generators is $\mathcal{G}$. Our proposed framework aims to minimize the MMD in a reduced function class Θ between π and $\varphi \in \mathcal{G}$.

[1] arXiv version is at `https://arxiv.org/abs/1910.09161`.

dius” (threshold) that we can find without observing normal sequences.

3. POINT PROCESS VIA DEEP FOURIER KERNEL

Now we present a marked Hawkes process model for the discrete events, which will lead to the detection statistic (i.e., the form of the likelihood function $\ell(\boldsymbol{x};\theta)$).

Hawkes processes with deep Fourier kernel. We represent the triggering function of the Hawkes process via a *deep Fourier kernel*. The spectrum for the Fourier features is parameterized by a deep neural network, as shown in Fig. 2. Assume each observation is a *marked spatio-temporal tuple* which consists of time, location, and marks: (t_i, m_i), where $t_i \in [0, T)$ is the time of occurrence of the ith event, and $m_i \in \mathcal{M} \subseteq \mathbb{R}^d$ is the d-dimensional mark (here we treat location as one of the mark). For notational simplicity, denote $x := (t, m) \in \mathcal{X}$ as the most recent event and $x' := (t', m') \in \mathcal{X}$, $t' < t$ as an occurred event in the past, where $\mathcal{X} := [0, T] \times \mathcal{M} \subset \mathbb{R}^{d+1}$ is the space for time and mark. Define the conditional intensity function as

$$\lambda(x|\mathcal{H}_t;\theta) = \mu + \alpha \sum\nolimits_{t'<t} K(x, x'), \tag{3}$$

where α represents the magnitude of the influence from the past, $\mu \geq 0$ is the constant background intensity of events, which can be estimated from data. The kernel function measures the influence of the past event on the current event x, x'.

The formulation of deep Fourier kernel function relies on Bochner’s Theorem [11], which states that any bounded, continuous, and shift-invariant kernel is a Fourier transform of a bounded non-negative measure. Assume such shift-invariant kernel is positive semi-definite and scaled such that $g(0) = 1$, Bochner’s theorem ensures that its Fourier transform p_ω can be viewed as a probability distribution function since it normalize to 1 and is non-negative. In this sense, the spectrum p_ω can be viewed as the distribution of r-dimensional Fourier features indexed by $\omega \in \Omega \subset \mathbb{R}^r$. Hence, we may obtain a triggering function in (3) which satisfies the “kernel embedding”:

Proposition 1. *Let the triggering function K be a continuous real-valued shift-invariant kernel and p_ω a probability distribution function. Then*

$$K(x, x') := \mathbb{E}_{\omega\sim p_\omega}\left[\phi_\omega(x) \cdot \phi_\omega(x')\right], \tag{4}$$

where $\phi_\omega(x) := \sqrt{2}\cos(\omega^\top W x + u)$ and $W \in \mathbb{R}^{r\times(d+1)}$ is a weight matrix These Fourier features $\omega \in \Omega \subset \mathbb{R}^r$ are sampled from p_ω and u is drawn uniformly from $[0, 2\pi]$.

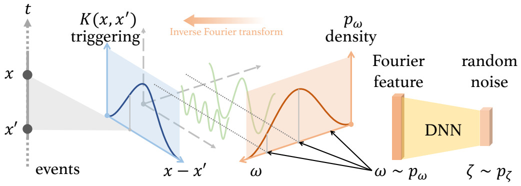

Fig. 2: An illustration for the Fourier kernel function $K(x, x')$ and its Fourier representation; the spectrum of Fourier features are represented by a deep neural network.

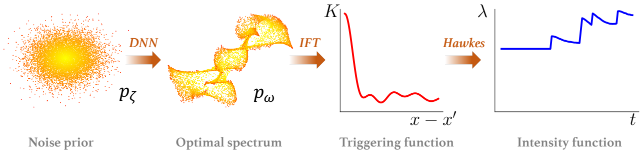

Fig. 3: An instance of calculating the conditional intensity λ through performing inverse Fourier transform.

In practice, the expression (4) can be approximated empirically, i.e., $\widetilde{K}(x, x') = \frac{1}{D}\sum_{k=1}^{D} \phi_{\omega_k}(x) \cdot \phi_{\omega_k}(x') = \Phi(x)^\top \Phi(x')$, where ω_k, $k = 1, \dots, D$ are D Fourier features sampled from the distribution p_ω. The vector $\Phi(x) := [\phi_{\omega_1}(x), \dots, \phi_{\omega_D}(x)]^\top$ can be viewed as the approximation of the kernel-induced feature mapping. In the experiments, we substitute $\exp\{\boldsymbol{i}w^\top(x - x')\}$ with a real-valued feature mapping, such that the probability distribution p_ω and the kernel K are real [12].

Fourier feature generator. To represent the distribution p_ω, we assume it is a transformation of random noise $\zeta \sim p_\zeta$ through a non-linear mapping $\psi_0 : \mathbb{R}^q \to \mathbb{R}^r$, as shown in Fig. 2, where ψ_0 is differentiable, and it is represented by a deep neural network, and q is the dimension of the noise. Roughly speaking, p_ω is the probability density function of $\psi_0(\zeta)$, $\zeta \sim p_\zeta$. Note that the triggering kernel is jointly controlled by the deep network parameters and the weight matrix W; we denote all of these parameters as $\theta \in \Theta$. Fig. 3 gives an illustrative example of representing the conditional intensity given sequence history using our approach. The optimal spectrum learned from data uniquely specifies a kernel function capable of capturing various non-linear triggering effects.

Efficient computation of log-likelihood. Given a sequence of events $\boldsymbol{x}$, the log-likelihood function of our model is written by substituting the conditional intensity function with (3), and thus we need to evaluate $\int_{\mathcal{X}} \lambda(x|\mathcal{H}_t;\theta)dx$. In many existing works, this term is carried out by some numerical integration techniques. Here we present a way to simplify the computation by deriving closed-form expression for the integral as the following proposition as a benefit given by the Fourier kernel.

Proposition 2 (Integral of conditional intensity function)**.** *Let $t_{N_T+1} = T$ and $t_0 = 0$. Given ordered events $\{x_1, \dots, x_{N_T}\}$ in the time horizon $[0, T]$. The integral term in the log-likelihood function can be written as*

$$\int_{\mathcal{X}} \lambda(x|\mathcal{H}_t;\theta)dx = \mu T(b-a)^d + \frac{1}{D}\sum\nolimits_{k=1}^{D}\sum\nolimits_{i=0}^{N_T}\sum\nolimits_{t_j<t_i}$$

$$\cos\left(-\omega_k^\top W x_j\right)\cos\left(\frac{t_{i+1}+t_i}{2}\right)\sin\left(\frac{t_{i+1}-t_i}{2}\right)$$

$$\cos^d\left(\frac{b+a}{2}\right)\sin^d\left(\frac{b-a}{2}\right)\prod\nolimits_{\ell=1}^{d+1}\frac{2e^{\omega_k^\top w_\ell}}{\omega_k^\top w_\ell},$$

where $w_\ell, \ell = 1, \dots, d$ is the ℓ-th column vector in the matrix W, and $[a, b]$ are the range for each dimension of the mark space $\mathcal{M}$.

Therefore the log-likelihood $\ell(\boldsymbol{x}_{1:i}; \theta^*)$ can be computed recursively as follows:

$$\ell(\boldsymbol{x}_{1:1}; \theta^*) = \log f(x_1 | \mathcal{H}_{t_1});$$
$$\ell(\boldsymbol{x}_{1:i}; \theta^*) = \ell(\boldsymbol{x}_{1:i-1}; \theta^*) + \log f(x_i | \mathcal{H}_{t_i}; \theta^*),\ \forall i > 1,$$

where $f(x_i | \mathcal{H}_{t_i}; \theta) = \lambda(x_i | \mathcal{H}_{t_i}; \theta) e^{-\mu(t_i - t_{i-1})(2\pi)^d}$ if we rescale the range of each coordinate of the mark to be $[0, 2\pi]$, i.e., $b = 2\pi$ and $a = 0$. This recursive expression makes it convenient to evaluate the detection statistic sequentially and perform online detection.

4. NUMERICAL EXPERIMENTS

We perform comprehensive numerical studies to compare the performance of the proposed adversarial anomaly detector with the state-of-the-art. Consider two synthetic and one real data sets: (1) **singleton synthetic data** consists of 1,000 anomalous sequences with an average length of 32. Each sequence is simulated by a Hawkes process with an exponential kernel; (2) **composite synthetic data** consists of 1,000 mixed anomalous sequences with an average length of 29. Every 200 of the sequences are simulated by five Hawkes processes with different exponential kernels; and a real dataset (3) **Macy's fraudulent credit transaction data** consists of 1,121 fraudulent credit transaction sequences with an average length of 21. Each anomalous transaction in a sequence includes the occurrence location, time, and transaction amount in the dollar. We then mix the above data sets, respectively, with 5,000 random "normal" sequences simulated by multiple Poisson processes. Detailed experimental settings can be found in the arXiv paper[1]. We compare our method (referred to as `AIL`) with three state-of-the-art approaches: one-class support vector machines [13] (`One-class SVM`), the cumulative sum of features extracted by principal component analysis [14] (`PCA+CUMCUM`), local outlier factor [15] (`LOF`); and a recent study on using IRL to attack sequential anomaly detection [16] (`IRL-AD`).

We summarize the results of our method on three data sets in Fig. 4 and confirm that the proposed time-varying threshold can optimally separate the anomalies from normal sequences. As we can see, the anomalous sequences attain a higher average log-likelihood than the normal sequences for all three data sets. Their log-likelihoods fall into different value ranges with rare overlap. Additionally, the time-varying threshold indicated by blue dash lines lies between the value ranges of anomalous and normal sequences, which produces an amicable separation of these two types of sequences at any given time. Colored cells of these heat-maps are calculated with different constant thresholds η at each step i by performing cross-validation. The brightest regions indicate the "ground truth" of the optimal choices of the threshold. As shown in the third column, the time-varying thresholds (blue dash line) are very close to the optimal choices found by cross-validation. We also compare the step-wise F_1 scores with the other four baselines in Fig. 5. The results show that (1) from an overall standpoint, our method outperforms other baselines with significantly higher F_1 scores, and (2) our approach allows for easier and faster detection of anomalous sequences, which is critically important in sequential scenarios for most of the applications.

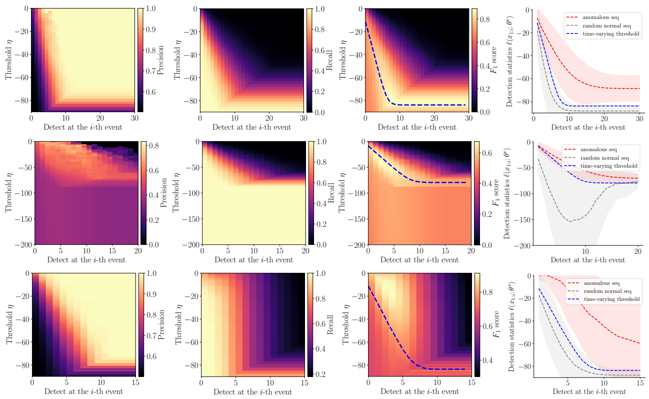

Fig. 4: Results of our method (`AIL`) on three data sets (rows from top to bottom correspond to synthetic, composite, and Macy's data, respectively). The blue lines in the third column indicate our time-varying thresholds. The fourth column shows the step-wise detection statistics for both anomalous and normal sequences.

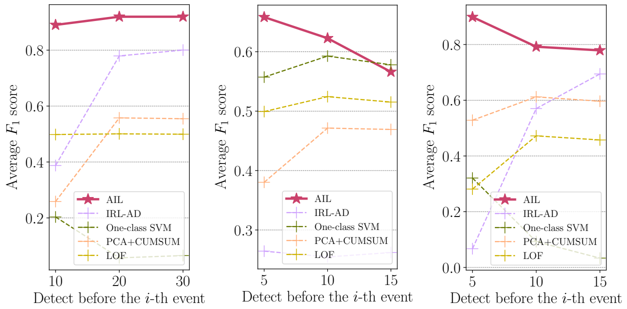

Fig. 5: Performance of our method (`AIL`) and other four baselines on three data sets (sub-figures from left to right correspond to synthetic, composite, and Macy's data, respectively).

5. CONCLUSION

We have presented a novel unsupervised anomaly detection framework on sequential data based on adversarial learning. A robust detector can be found by solving a minimax problem, and the optimal generator also helps in defining the time-varying threshold for making decisions in an online fashion. We model the sequential event data using a marked point process model with a deep Fourier kernel. We believe the proposed framework is a natural way to tackle the one-class anomaly detection problem. This new formulation may provide a first step towards bridging adversarial learning and sequential anomaly detection.

6. REFERENCES

[1] Varun Chandola, Arindam Banerjee, and Vipin Kumar, "Anomaly detection for discrete sequences: A survey," *IEEE transactions on knowledge and data engineering*, vol. 24, no. 5, pp. 823–839, 2010.

[2] Xin Xu, "Sequential anomaly detection based on temporal-difference learning: Principles, models and case studies," *Applied Soft Computing*, vol. 10, no. 3, pp. 859–867, 2010.

[3] Junyoung Chung, Kyle Kastner, Laurent Dinh, Kratarth Goel, Aaron C Courville, and Yoshua Bengio, "A recurrent latent variable model for sequential data," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds., pp. 2980–2988. Curran Associates, Inc., 2015.

[4] Keval Doshi and Yasin Yilmaz, "Any-shot sequential anomaly detection in surveillance videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 934–935.

[5] Ahmed Hussein, Mohamed Medhat Gaber, Eyad Elyan, and Chrisina Jayne, "Imitation learning: A survey of learning methods," *ACM Comput. Surv.*, vol. 50, no. 2, Apr. 2017.

[6] Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola, "A kernel two-sample test," *Journal of Machine Learning Research*, vol. 13, no. 25, pp. 723–773, 2012.

[7] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[8] D. Siegmund and American Mathematical Society, *Sequential Analysis: Tests and Confidence Intervals*, Springer Series in Statistics. Springer, 1985.

[9] Pieter Abbeel and Andrew Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the Twenty-First International Conference on Machine Learning*, New York, NY, USA, 2004, ICML '04, p. 1, Association for Computing Machinery.

[10] Andrew Y. Ng and Stuart Russell, "Algorithms for inverse reinforcement learning," in *in Proc. 17th International Conf. on Machine Learning*. 2000, pp. 663–670, Morgan Kaufmann.

[11] Walter Rudin, *Fourier analysis on groups*, vol. 121967, Wiley Online Library, 1962.

[12] Ali Rahimi and Benjamin Recht, "Random features for large-scale kernel machines," in *Advances in Neural Information Processing Systems 20*, J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, Eds., pp. 1177–1184. Curran Associates, Inc., 2008.

[13] Rui Zhang, Shaoyan Zhang, Sethuraman Muthuraman, and Jianmin Jiang, "One class support vector machine for anomaly detection in the communication network performance data," in *Proceedings of the 5th Conference on Applied Electromagnetics, Wireless and Optical Communications*, Stevens Point, Wisconsin, USA, 2007, ELECTROSCIENCE'07, p. 31–37, World Scientific and Engineering Academy and Society (WSEAS).

[14] Ewan S Page, "Continuous inspection schemes," *Biometrika*, vol. 41, no. 1/2, pp. 100–115, 1954.

[15] Markus Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander, "Lof: Identifying density-based local outliers," in *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*. 2000, pp. 93–104, ACM.

[16] Min-hwan Oh and Garud Iyengar, "Sequential anomaly detection using inverse reinforcement learning," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, New York, NY, USA, 2019, KDD '19, p. 1480–1490, Association for Computing Machinery.