

Special Issue on Deep Learning for Anomaly Detection

Anomaly detection (also known as outlier/novelty detection) aims at identifying data points which are rare or significantly different from the majority of data points. Due to the significance to many critical domains like cybersecurity, fintech, healthcare, public security and AI safety, anomaly detection has been one of the most active research areas in various communities, such as data mining, machine learning and computer vision. Many techniques are explored to build highly efficient and effective anomaly detection systems, but they are confronted with many difficulties when dealing with complex data, such as failing to capture intricate feature interactions or extract good feature representations. Deep learning techniques have shown very promising performance in tackle different types of complex data in a broad range of tasks/problems, but its development in the area of anomaly detection is relatively slow due to some unique characteristics of anomalies, e.g., rarity, heterogeneity, boundless nature, and prohibitively high cost of collecting large-scale anomaly data. This special issue aims to promote the development of deep learning techniques specially designed for anomaly detection.

Scope of The Special Issue

This special issue will feature the most recent advances in deep learning techniques for anomaly detection. It targets both academic researchers and industrial practitioners from data mining, machine learning and computer vision communities, and solicits original and high-quality research on but not limited to the following topics.

- * Deep anomaly detection in different types of data
 - Multivariate data
 - Graph data
 - Video data
 - Image data
 - Text data
 - Sequence data
 - Time series data
 - Spatial-temporal data
 - Multimodal data
- * Representation learning for anomaly detection
 - Un/weakly/semi/fully-supervised models
 - Multilayer perceptron
 - Convolutional neural network
 - Recurrent neural network
 - Graph neural network
- * End-to-end anomaly classification
 - Un/weakly/semi/fully-supervised models
 - Deep neural network models
 - Deep reinforcement learning models
- Deep transfer learning models
- Deep reasoning models
- * Deep anomaly detection theories/foundation
 - Mathematical formalization
 - Optimization
 - Generalization bounds and learnability
 - Anomaly explanation
- * Relevant applications of deep anomaly detection
 - Fraud detection in finance/insurance
 - Disease/tumor detection in healthcare
 - Intrusion/malware detection in cybersecurity
 - Malicious activity detection in social networks
 - Event detection in video surveillance
 - Safety analysis in AI systems
- * Related areas addressing similar issues
 - Out-of-distribution example detection
 - Curiosity learning
 - Anti-spoofing techniques
 - Adversarial example detection

Timeline

The timeline of the special issue is to be scheduled as follows:

- Paper Submission: November 30, 2020
- First Decision: February 15, 2021
- Revision: April 15, 2021
- Final Decision: May 31, 2021
- Camera-ready: June 30, 2021
- Publication: July 2021

Guest Editors

- Guansong Pang, University of Adelaide, Australia (guansong.pang@adelaide.edu.au)
- Charu Aggarwal, IBM T. J. Watson Research Center, United States (charu@us.ibm.com)
- Chunhua Shen, University of Adelaide, Australia (chunhua.shen@adelaide.edu.au)
- Nicu Sebe, University of Trento, Italy (sebe@disi.unitn.it)

Submission Instructions

- Read the Information for Authors at <http://cis.ieee.org/tnnls>
- Submit your manuscript at the TNNLS webpage (<http://mc.manuscriptcentral.com/tnnls>) and follow the submission procedure. Please, clearly indicate on the first page of the manuscript and in the cover letter that the manuscript is submitted to this special issue. Send an email to the leading guest editor Dr. Guansong Pang (guansong.pang@adelaide.edu.au) with subject "TNNLS special issue submission" to notify about your submission
- Early submissions are welcome. We will start the review process as soon as we receive your contributions