

Spatially-Aware Autoencoders for Detecting Contextual Anomalies in Geo-Distributed Data

Roberto Corizzo¹[0000-0001-8366-6059], Michelangelo
Ceci^{1,3}[0000-0002-6690-7583], Gianvito Pio²[0000-0003-2520-3616], Paolo
Mignone¹[0000-0002-8641-7880], and Nathalie Japkowicz¹[0000-0003-1176-1617]

Dept. of Computer Science, American University, Washington, DC 20016, USA¹

{rcorizzo, japkowicz}@american.edu

Dept. of Computer Science, University of Bari Aldo Moro, Bari, Italy²

{michelangelo.ceci, gianvito.pio, paolo.mignone}@uniba.it

Dept. of Knowledge Technologies, Jožef Stefan Institute, Ljubljana, Slovenia³

Abstract. The huge amount of data generated by sensor networks enables many potential analyses. However, one important limiting factor for the analyses of sensor data is the possible presence of anomalies, which may affect the validity of any conclusion we could draw. This aspect motivates the adoption of a preliminary anomaly detection method. Existing methods usually do not consider the spatial nature of data generated by sensor networks. Properly modeling the spatial nature of the data, by explicitly considering spatial autocorrelation phenomena, has the potential to highlight the degree of agreement or disagreement of multiple sensor measurements located in different geographical positions. The intuition is that one could improve anomaly detection performance by considering the spatial context. In this paper, we propose a spatially-aware anomaly detection method based on a stacked auto-encoder architecture. Specifically, the proposed architecture includes a specific encoding stage that models the spatial autocorrelation in data observed at different locations. Finally, a distance-based approach leverages the embedding features returned by the auto-encoder to identify possible anomalies. Our experimental evaluation on real-world geo-distributed data collected from renewable energy plants shows the effectiveness of the proposed method, also when compared to state-of-the-art anomaly detection methods.

Keywords: Anomaly detection · Auto-encoders · Geo-distributed data

1 Introduction

The increasing adoption of sensor networks leads to the generation of a large amount of data, that could fruitfully be analyzed to support decision-making processes in multiple real-world sectors. Machine learning and data mining methods for the analysis of data generated by sensor networks have been adopted in multiple application domains. However, it is noteworthy that data collected through sensor networks are inherently affected by anomalies. This is due to the nature of the sensors, which operate in an external environment, and to the nature of the network (grid), which can be subject to communication issues. Therefore,

directly using raw sensor data to solve the task at hand may result in a degraded accuracy of the models [10]. For this reason, incorporating preliminary anomaly detection phases in the data analysis workflow appears fundamental. Recently, this thread has attracted increasing interest, with emerging approaches tailored for specific representations and for the detection of specific anomalies [19].

Sensor networks also open to the possibility to collect observations for a set of properties of interest in multiple geographical locations. In the literature, statistical techniques have been investigated to analyze geo-distributed sensor data in a combined manner, trying to improve the performance of the learning models. For instance, the incorporation of statistical indicators of spatio-temporal autocorrelation in classical machine learning algorithms has been successfully investigated in [8, 17]. However, this opportunity has been often disregarded by recent anomaly detection approaches, often based on deep neural network architectures [19]. The goal of this paper is to fill this gap. Specifically, we propose a method to solve *unsupervised* anomaly detection tasks, where the considered anomalies are *contextual* [9, 10]. More in detail, anomalies are detected on a single geographic position on the basis of the multi-dimensional sensor data observed at that location and its neighboring locations (*diffused context* [9, 10]).

Methodologically, we propose a neural network architecture, based on stacked auto-encoders, that incorporates a specific spatial encoding component to capture spatial autocorrelation phenomena. We argue that capturing the agreement (or disagreement) of the measurement of the same physical property, at the same time point, in multiple locations may boost the anomaly detection accuracy of the model. The adoption of stacked auto-encoders in our method is motivated by their ability to learn non-linear representations that effectively incorporate salient features [7]. The hidden layers of the model architecture are usually chosen to have a reduced number of neurons, compared to the input layer, representing data with a reduced dimensionality.

Auto-encoders have already been exploited to solve anomaly detection tasks in [3, 20], mainly leveraging the reconstruction error. A popular approach is to train the auto-encoder on background data, which is assumed to belong to the *normal* class (i.e., without anomalies). After the training stage, new instances fed to the model are expected to exhibit a low reconstruction error if they belong to the normal class. On the contrary, anomalies are expected to show a high reconstruction error, due to the fact that they belong to a different distribution.

Although this approach appears relatively intuitive, *i)* it does not take into account the spatial dimension in the data for the identification of the anomalies and *ii)* it might be susceptible to noise introduced in the data. Both these aspects are typical of data generated by sensor networks and the analysis of such type of data requires to overcome them. As for *i)*, we explicitly consider spatial autocorrelation in the learning phase and, as for *ii)* we propose to identify anomalies in the embedding space rather than in the original feature space, to be more robust to the presence of noise in the data. For this purpose, we propose to leverage the feature extraction capability of the model and perform anomaly detection by analyzing the embedding bottleneck features of the stacked auto-encoder.

In summary, the contributions of this paper are the following: *i)* we propose a stacked auto-encoder architecture which incorporates a spatial encoding stage in its architecture, to explicitly model spatial autocorrelation in geo-distributed multi-variate sensor data; *ii)* we devise a distance-based anomaly detection technique that leverages the distance among data observations, represented according to an embedding space learned by the stacked auto-encoder; *iii)* we evaluate the proposed approach on real-world datasets related to the renewable energy field.

2 Background

Data anomalies can usually be classified in three categories: point, contextual, and collective anomalies [9]. In this paper we address the detection of contextual anomalies, where the context is represented by the spatial dimension of a data observation [12, 16]. For instance, a contextual anomaly could be represented by an abrupt temperature value measurement at one geographical location.

In general, the identification of contextual anomalies can be carried out with supervised, semi-supervised or unsupervised machine learning approaches [9]. Although there are several machine learning based methods, unsupervised ones are better suited for domains characterized by a scarce availability (or by the total absence) of labeled data, which is the case in many real-world scenarios.

Among existing methods, it is worth mentioning One-Class SVM (OCSVM) [15], that learns a separating hyperplane in a high-dimensional space [15]. Once the model is learned, OCSVM can classify a new data observation as similar (i.e., normal) or different (i.e., anomaly) with respect to the training data distribution, according to its position within the decision boundary. In this line of research, OCSVM models have also been adopted in ensemble settings [1, 18].

Isolation Forests [13] exploit a combination of tree-based models, through which calculate an isolation score for each data observation. Specifically, the score of an observation is computed as the average path length from the root of the tree to the node containing the single observation. A short path indicates that an observation is easy to isolate from the others due to significantly different attribute values compared to the training data points.

In this scenario, methods based on auto-encoders and stacked auto-encoders [20] have demonstrated superior performance. This behavior is theoretically motivated by their ability to construct representations, with a low reconstruction error, based on non-linear combinations of the input features [4].

Although auto-encoders have seen particular interest for anomaly detection from images [20], in this work we adopt such models and investigate their effectiveness for the detection of abrupt changes in multivariate time-series data. Moreover, we introduce a novel component in the architecture to explicitly model spatial autocorrelation phenomena.

3 Method

The method proposed in this paper is able to analyze multi-variate sensor data (related, for example, to temperature, wind speed, pressure, etc.) collected from

multiple geo-distributed locations. Specifically, considering a discrete timeline and a set of locations L , let $x_{t,l}$ be the vector of measurements at time t and location $l \in L$. The multi-variate data coming from sensors can be represented as an unbounded sequence (i.e., a stream) of sets:

$$D = \langle \{x_{1,1}, \dots, x_{1,|L|}\}, \{x_{2,1}, \dots, x_{2,|L|}\}, \dots, \{x_{t,1}, \dots, x_{t,|L|}\} \rangle$$

We learn a stacked auto-encoder using D as input data representation to subsequently carry out the anomaly detection task. The adoption of stacked auto-encoders is motivated by their ability to extract layer-wise representations, at increasing levels of abstraction. In general, the first layer of a stacked auto-encoder learns simple features (e.g., edges, in the image domain), whereas deeper layers learn features at increasing levels of complexity and summarization (e.g., co-occurring edges that form corners). This characteristic allows to model complex properties of background data in the embedding space, that may in turn lead to an increased ability to discriminate between normal and anomalous instances.

In the following subsection, we describe the proposed strategy to explicitly consider spatial autocorrelation phenomena in the auto-encoder architecture.

3.1 Spatial encoding stage

The proposed auto-encoder architecture features a *spatial encoding* stage which is based on LISA (Local Indicators of Spatial Autocorrelation)[2], that simultaneously exploits data available at every location.

In order to describe how the spatial encoding stage works, we introduce how the computation of LISA is performed. The spatial neighborhood of the sensor network is expressed as a matrix, and for each location and data observation at a time point t , LISA is computed using such a matrix. Specifically, the first step is to define a neighborhood matrix $A \in \mathbb{R}^{|L| \times |L|}$, such that:

$$A[i, j] = 1 - \text{dist}(l_i, l_j) / \text{maxDist} \quad (1)$$

where $l_i \in L$ and $l_j \in L$ are two locations, $\text{dist}(l_i, l_j)$ is the spatial distance (in kilometers) between the two locations l_i and l_j , and maxDist is the maximum pairwise spatial distance observed in the sensor network.

A subsequent step computes the deviation of each data feature with respect to the mean, leveraging z-score normalization. Intuitively, in our approach we are interested in identifying the contribution of the neighboring locations for each feature observed at each time point. Therefore, given $D_t = \{x_{t,1}, \dots, x_{t,|L|}\} \in D$, the subset of all data observations for all the locations at a specific time t , the z-scores for a location $l \in L$ are calculated as: $z_{t,l}^{(f)} = (x_{t,l}^{(f)} - \overline{D_t^{(f)}}) / \sigma_{D_t^{(f)}}$, where f is a generic feature measured by a sensor (that is, a generic element of the vector $x_{t,l}$); $\overline{D_t^{(f)}}$ represents the average value of the feature f in all the locations; $\sigma_{D_t^{(f)}}$ represents the standard deviation of the feature f in all the locations. Leveraging $z_{t,l}^{(f)}$, it is possible to compute LISA for the variable f of the location l_i for time t (according to [2]) as follows:

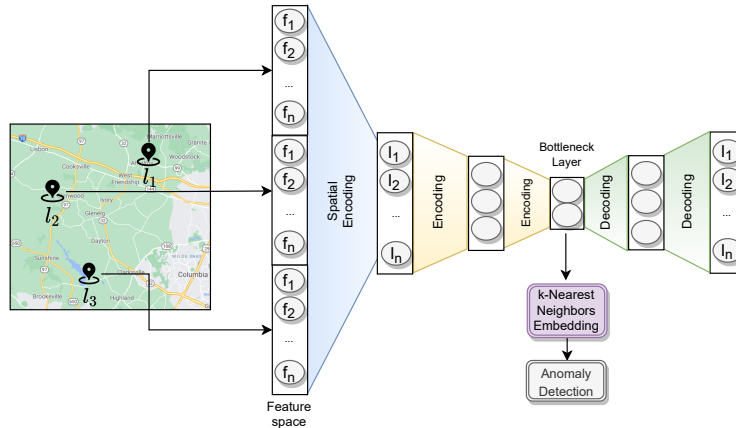


Fig. 1. A graphical representation of our spatially-aware auto-encoder architecture.

$$I_{l_i,t}^{(f)} = z_{t,l_i}^{(f)} \cdot \sum_{l_j \in L, i \neq j} A[i,j] \cdot z_{t,l_j}^{(f)} \quad (2)$$

Following the aforementioned process, for each time point t , the spatial encoding stage extracts a new representation S_t as follows:

$$S_t = \{[I_{l_1,t}^{(f_1)}, \dots, I_{l_1,t}^{(f_n)}], [I_{l_2,t}^{(f_1)}, \dots, I_{l_2,t}^{(f_n)}], \dots, [I_{|L|,t}^{(f_1)}, \dots, I_{|L|,t}^{(f_n)}]\} \quad (3)$$

3.2 Encoding and decoding stage

The subsequent encoding stages extract new representations with a lower dimensionality than the input data, similarly to the typical auto-encoder architecture. In our model, we perform two encoding stages after the spatial encoding stage (see Figure 1), with 1/2 and 1/4 of the input features, respectively. The architecture is trained end-to-end leveraging historical data which represent normal behavior conditions. We assume that historical data contains no anomalies (or a negligible amount), and use the trained model for anomaly detection purposes.

Starting from the dataset D , the stacked auto-encoder aims at learning the encoding function $e : \mathcal{X} \rightarrow \mathcal{F}$ and the decoding function $d : \mathcal{F} \rightarrow \mathcal{X}$, such that:

$$\langle e(\cdot), d(\cdot) \rangle = \underset{\langle e(\cdot), d(\cdot) \rangle}{\operatorname{argmin}} \|D - d(e(D))\|^2, \quad (4)$$

where \mathcal{X} is the input space of D , and \mathcal{F} is the learned embedding space.

The functions $e(\cdot)$ and $d(\cdot)$ should be parametric and differentiable according to a distance function. Consequently, the parameters of the encoding and decoding functions defined above are optimized by minimizing the reconstruction loss.

3.3 Embedding-based anomaly detection

To detect anomalies, we propose a k -Nearest Neighbors approach that leverages the encoded data representation. Once the auto-encoder is trained with the available historical data, we compute the average Euclidean distance between each data observation and its nearest k observations in the embedding space. Coherently, when a new observation is available, we encode it in the embedding space, and compute its average distance w.r.t. the nearest k observations. If the distance is greater than a given threshold, then the observation is considered as an anomaly. In this work, we do not adopt a manual threshold, but estimate it from the data distribution. Specifically, we use $[\bar{d} + 3 \cdot \sigma]$, where \bar{d} is the average pairwise distance observed between each training data observation and its nearest observations, while σ is the standard deviation of the observed distances.

Note that the identification of the k nearest neighbors in our method is based on the Hybrid Spill Tree (HSP) [14], a distributed data structure (variant of metric trees) for high-dimensional indexing, that allows to retrieve the k nearest neighbors of an observation in $O(\log|D|)$.

4 Experiments

4.1 Datasets

The datasets considered in our experiment consist of weather variables (such as temperature, humidity, etc.) monitored at hourly granularity by sensors placed on renewable energy plants, located in different geographical areas. In particular, we considered the following datasets analyzed also in previous studies [11]:

- **PV Italy.** The dataset consists of data collected every 15 minutes (from 2:00 AM to 8.00 PM, every day) by sensors located on 17 photovoltaic power plants located in Italy. The time period spans from January 1st, 2012 to May 4th, 2014. More details about data preprocessing steps can be found in [6].
- **Wind NREL.** This dataset (www.nrel.gov/wind) was modeled using the Weather Research & Forecasting model. Five plants with the highest production have been selected, obtaining the time series of wind speed and production observed every 10 minutes, for a time period of two years (from January 1st, 2005 to December 31st, 2006). Hourly aggregation was performed.

For both datasets, we consider the following features: latitude and longitude of each plant; day and hour; altitude and azimuth; weather conditions, i.e., ambient temperature, irradiance, pressure, wind speed, wind bearing, humidity, dew point, cloud cover, and a descriptive weather summary. Weather conditions are either measured (training phase) or forecasted (detection phase). In particular, all the weather data were extracted from Forecast.io, except for the expected altitude and azimuth, that were extracted from SunPosition (www.susdesign.com/sunposition), and the expected irradiance (PV Italy dataset only), that was extracted from PVGIS (re.jrc.ec.europa.eu/pvg_tools/en/#MR).

For each dataset, we build the testing set by selecting all the instances (measurements at hourly granularity, observed at all the plants) belonging to 10

randomly selected days. We analyze the anomaly detection capabilities of the method, considering three different training window sizes: 30, 60 and 90 days. This means that, for each day in the testing set, we train the model using historical data belonging to 30, 60 or 90 days, respectively, preceding the considered testing day, with the goal of identifying anomalies for all the measurements belonging to the considered day of the testing set. For evaluation purposes, anomalies are artificially introduced by perturbing the correct attribute values. This is done on 25% of instances on 50% of the features.

4.2 Competitor systems and experimental setup

In line with the discussion of existing works reported in Section 2, in our experiments we considered, as possible competitors, the most suitable class of approaches to address the task of interest in our study, that are mainly based one-class classification. Indeed, they offer the flexibility to learn a model from an initial (regular) data distribution and are able to flag data that significantly differ from the learned distribution. In particular, we considered three state-of-the-art competitor methods falling in this class, namely One-Class SVM (**OCSVM**) [15], **Isolation Forest** [13], and an **Auto-encoder** architecture that bases the detection of anomalies on the reconstruction error [3, 20]. These approaches are widely adopted, and generally provide highly accurate detections.

Their parameters were set to the values suggested in their respective papers. In particular, for One-Class SVM, we choose a Radial Basis Function (RBF) kernel and select the best value for the γ parameter in the set $\gamma \in \{0.1, \textit{scale}, \textit{auto}\}$. The *auto* configuration corresponds to $\gamma = \frac{1}{n_features}$, whereas the *scale* configuration corresponds to $\gamma = \frac{1}{n_features \cdot var(X)}$, where $var(X)$ represents the variance of the training data. For Isolation Forest, we set: the number of base estimators in the ensemble $n_estimators \in \{10, 25, 50\}$; the number of features to draw at random for each base estimator equal to the number of the whole set of features. For the auto-encoder, we followed the heuristics proposed by [5]: we initially experimented with different configurations for *learning rate* (negative powers of 10, starting from a default value of 0.01) and *batch size* (powers of 2) using a 20% validation set. Preliminary results suggested that the different configurations did not affect performance metrics significantly. For this reason, the experiments were performed with the following parameters: *epochs*=500, *learning rate*=0.0001, *batch size*=32. Moreover, we experimented with two different values of its parameter p , i.e., $p \in \{1.5, 3\}$ (if the reconstruction error deviates more than $p \cdot \sigma$ from the one observed on the training set, the instance is marked as an anomaly).

As regards our method, we report the results with different values of k , namely $k \in \{50, 100, 150\}$. Finally, in order to specifically evaluate the contribution provided by the spatial embedding component, we also report the results obtained by a simplified version of the proposed architecture, that does not exploit the spatial embedding step. We call this variant **Without SE**.

All the results were collected in terms of Precision, Recall and F-Score.

4.3 Results and discussion

In Table 1 we report all the results obtained in our experiments. First, we can observe that our approach generally obtains the best results among all the considered methods. Looking specifically at the results on PV Italy, we can observe that the best F-Score results are obtained with a time window of 30 days. This means that the kNN-based approach that we propose achieves optimal results even with a limited view on historical data. Looking at Precision and Recall, it is clear that our approach is sensitive to anomalies, but robust to false detections: the results in terms of precision ($\sim 98-99\%$) indicate that the false positive rate is around 1-2%, while the Recall results indicate a good rate of detected anomalies, i.e., around 75%. Such results are not obtained by competitor systems, that show a significantly lower Precision ($\sim 85\%$ in the best case, obtained by the Auto-encoder, $\sigma = 1.5$), and recall, especially in the case of Isolation Forest.

Looking at the simpler version of our method (Without SE), we can observe comparable, but lower results than those achieved by the full variant of our method. This behavior confirms that the proposed architecture, based on kNN on the embedded instances, is generally effective and is further supported by the spatial encoding step that takes into account spatial autocorrelation phenomena.

A closer look at the results obtained on the Wind NREL dataset reveals a similar situation. In this case, we can only observe one case (i.e., window size = 90 days and $k = 150$) in which the best results are achieved by the variant of our method that does not exploit the spatial embedding component. However, the difference with the full version of our method is negligible, and it may be possibly due to the fact that, in this dataset, less features are correlated to the spatial dimension, with respect to the photovoltaic power plants in PV Italy (see, e.g., the irradiance feature). Nevertheless, a contribution of the spatial encoding step can still be observed when the time window is limited to 30 or 60 days.

Focusing on the best F-score results achieved by the considered methods, measured over all the values of their parameters, we can easily observe that the proposed method always outperforms all the other competitors. We can also observe a slightly higher Recall exhibited by the auto-encoder, but at the price of a significantly lower Precision. However, our method generally leads to a 7%-9% improvement in terms of F-score in all the cases with respect to the auto-encoder.

5 Conclusion

In this paper we presented a novel anomaly detection method based on an auto-encoder architecture that features a spatial encoding stage to model spatial autocorrelation. The proposed architecture is unsupervised, and the model is trained using historical data. The anomaly detection task is carried out by comparing the projection of new observations in the embedding space to their nearest neighbors. This strategy allows us to detect anomalies using a distance-based approach that exploits a threshold automatically estimated from training embedded data.

The experimental evaluation performed on two datasets, related to real-world sensor networks of power plants, showed significant improvements in terms of F-Score, that reaches 9.18% compared to auto-encoders based on reconstruction

Table 1. Anomaly detection results obtained considering varying training sliding window sizes. Best F-Score results for each Window size configuration are marked in bold.

<i>PV Italy</i>	30 days			60 days			90 days		
Our Method	Prec	Rec	F-Score	Prec	Rec	F-Score	Prec	Rec	F-Score
<i>k</i> =50	0.9853	0.7433	0.8472	0.9801	0.7437	0.8444	0.9764	0.7412	0.8419
<i>k</i> =100	0.9896	0.7452	0.8500	0.9853	0.7455	0.8478	0.9787	0.7430	0.8436
<i>k</i> =150	0.9925	0.7464	0.8519	0.9871	0.7458	0.8488	0.9814	0.7433	0.8451
Without SE	Prec	Rec	F-Score	Prec	Rec	F-Score	Prec	Rec	F-Score
<i>k</i> =50	0.9248	0.7729	0.8250	0.9161	0.7948	0.8344	0.9118	0.7965	0.8337
<i>k</i> =100	0.9517	0.7528	0.8261	0.9323	0.7687	0.8260	0.9228	0.7711	0.8236
<i>k</i> =150	0.9651	0.7427	0.8277	0.9451	0.7553	0.8246	0.9349	0.7600	0.8229
Auto-encoder	Prec	Rec	F-Score	Prec	Rec	F-Score	Prec	Rec	F-Score
<i>p</i> = 1.5	0.8516	0.8179	0.7836	0.8502	0.8145	0.7774	0.8467	0.8132	0.7771
<i>p</i> = 3	0.7880	0.7703	0.7055	0.7877	0.7698	0.7049	0.7892	0.7722	0.7087
OCSVM	Prec	Rec	F-Score	Prec	Rec	F-Score	Prec	Rec	F-Score
	0.7880	0.7386	0.7305	0.7880	0.7386	0.7305	0.7880	0.7386	0.7305
Isolation Forest	Prec	Rec	F-Score	Prec	Rec	F-Score	Prec	Rec	F-Score
n_estimators=10	0.6277	0.4409	0.4033	0.6277	0.4409	0.4033	0.6277	0.4409	0.4033
n_estimators=25	0.6799	0.5092	0.5007	0.6799	0.5092	0.5007	0.6799	0.5092	0.5007
n_estimators=50	0.6879	0.5406	0.5450	0.6879	0.5406	0.5450	0.6879	0.5406	0.5450
<i>Wind NREL</i>	30 days			60 days			90 days		
Our Method	Prec	Recall	F-Score	Prec	Recall	F-Score	Prec	Recall	F-Score
<i>k</i> =50	0.9871	0.7558	0.8523	0.9726	0.7675	0.8490	0.9636	0.7675	0.8453
<i>k</i> =100	0.9992	0.7508	0.8569	0.9887	0.7567	0.8530	0.9837	0.7533	0.8494
<i>k</i> =150	0.9992	0.7508	0.8569	0.9938	0.7542	0.8553	0.9915	0.7533	0.8534
Without SE	Prec	Recall	F-Score	Prec	Recall	F-Score	Prec	Recall	F-Score
<i>k</i> =50	0.9768	0.7586	0.8478	0.9585	0.7664	0.8421	0.9590	0.7700	0.8439
<i>k</i> =100	0.9930	0.7547	0.8548	0.9781	0.7559	0.8474	0.9772	0.7647	0.8503
<i>k</i> =150	0.9970	0.7526	0.8564	0.9874	0.7529	0.8511	0.9884	0.7581	0.8537
Auto-encoder	Prec	Rec	F-Score	Prec	Rec	F-Score	Prec	Rec	F-Score
<i>p</i> = 1.5	0.8371	0.8344	0.7907	0.8704	0.8408	0.7997	0.8652	0.8384	0.7971
<i>p</i> = 3	0.7159	0.7880	0.7068	0.7164	0.7888	0.7085	0.6924	0.7888	0.7083
OCSVM	Prec	Rec	F-Score	Prec	Rec	F-Score	Prec	Rec	F-Score
	0.8341	0.8040	0.8050	0.8341	0.8040	0.8050	0.8341	0.8040	0.8050
Isolation Forest	Prec	Rec	F-Score	Prec	Rec	F-Score	Prec	Rec	F-Score
n_estimators=10	0.7045	0.3792	0.3198	0.7045	0.3792	0.3198	0.7045	0.3792	0.3198
n_estimators=25	0.8076	0.4128	0.3585	0.8076	0.4128	0.3585	0.8076	0.4128	0.3585
n_estimators=50	0.8181	0.4272	0.3729	0.8181	0.4272	0.3729	0.8181	0.4272	0.3729

error. A direct comparison with a variant of the proposed method, that does not exploit the spatial encoding component, also revealed the positive contribution coming from the explicit consideration of the spatial information.

As future work we will investigate other approaches to model spatio-temporal autocorrelation, as part of the neural network architecture. Moreover, we will conduct an extensive experimental evaluation involving datasets related to other domains, and affected by different amounts and types of anomalies.

Acknowledgement

The authors acknowledge the support of the U.S. DARPA through the project “Lifelong Streaming Anomaly Detection” (Grant N. A19-0131-003 and A21-0113-002), and of the EU Commission through the H2020 project “IMPETUS-Intelligent Management of Processes, Ethics and Technology for Urban Safety” (Grant n. 883286). GP acknowledges the support of Ministry of Universities and Research (MUR) through the project “Big Data Analytics”, AIM 1852414, activity 1, line 1. PM acknowledges the support of Apulia Region through the

project “Metodi per l’ottimizzazione delle reti di distribuzione di energia e per la pianificazione di interventi manutentivi ed evolutivi” (CUP H94I20000410008, Grant n. 7EDD092A) in the context of “Research for Innovation - REFIN”.

References

1. Aggarwal, C.C.: Outlier ensembles: position paper. *ACM SIGKDD Explorations Newsletter* **14**(2), 49–58 (2013)
2. Anselin, L.: Local indicators of spatial association—LISA. *Geographical analysis* **27**(2), 93–115 (1995)
3. Beggel, L., Pfeiffer, M., Bischl, B.: Robust anomaly detection in images using adversarial autoencoders. In: *ECMLPKDD 2019*. pp. 206–222. Springer (2019)
4. Bengio, Y.: *Learning deep architectures for AI*. Now Publishers Inc (2009)
5. Bengio, Y.: Practical recommendations for gradient-based training of deep architectures. In: *Neural networks: Tricks of the trade*, pp. 437–478. Springer (2012)
6. Ceci, M., Corizzo, R., Fumarola, F., Malerba, D., Rashkovska, A.: Predictive modeling of pv energy production: How to set up the learning task for a better prediction? *IEEE Transactions on Industrial Informatics* **13**(3), 956–966 (2016)
7. Ceci, M., Corizzo, R., Japkowicz, N., Mignone, P., Pio, G.: Echad: Embedding-based change detection from multivariate time series in smart grids. *IEEE Access* **8**, 156053–156066 (2020)
8. Ceci, M., Corizzo, R., Malerba, D., Rashkovska, A.: Spatial autocorrelation and entropy for renewable energy forecasting. *Data Mining and Knowledge Discovery* **33**(3), 698–729 (2019)
9. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. *ACM computing surveys (CSUR)* **41**(3), 15 (2009)
10. Corizzo, R., Ceci, M., Japkowicz, N.: Anomaly detection and repair for accurate predictions in geo-distributed big data. *Big Data Research* **16**, 18–35 (2019)
11. Corizzo, R., Pio, G., Ceci, M., Malerba, D.: DENCAST: distributed density-based clustering for multi-target regression. *J. Big Data* **6**, 43 (2019)
12. Kou, Y., Lu, C.T., Chen, D.: Spatial weighted outlier detection. In: *Proc. of SIAM international conference on data mining 2006*. pp. 614–618. SIAM (2006)
13. Liu, F.T., Ting, K.M., Zhou, Z.H.: Isolation forest. In: *2008 Eighth IEEE International Conference on Data Mining*. pp. 413–422. IEEE (2008)
14. Liu, T., Moore, A.W., Gray, A., Yang, K.: An investigation of practical approximate nearest neighbor algorithms. In: *NIPS 2004*. p. 825–832. MIT Press (2004)
15. Schölkopf, B., Williamson, R.C., Smola, A.J., Shawe-Taylor, J., Platt, J.C.: Support vector method for novelty detection. In: *Advances in neural information processing systems*, pp. 582–588 (2000)
16. Shekhar, S., Lu, C.T., Zhang, P.: Detecting graph-based spatial outliers: algorithms and applications (a summary of results). In: *ACM SIGKDD*. pp. 371–376 (2001)
17. Stojanova, D., Ceci, M., Appice, A., Malerba, D., Džeroski, S.: Dealing with spatial autocorrelation when learning predictive clustering trees. *Ecological Informatics* **13**, 22–39 (2013)
18. Xing, H.J., Liu, W.T.: Robust adaboost based ensemble of one-class support vector machines. *Information Fusion* **55**, 45–58 (2020)
19. Zhang, C., Song, D., Chen, Y., Feng, X., et al.: A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In: *AAAI 2019*. vol. 33, pp. 1409–1416 (2019)
20. Zhou, C., Paffenroth, R.C.: Anomaly detection with robust deep autoencoders. In: *ACM SIGKDD 2017*. pp. 665–674 (2017)