Artificial Intelligence In Medicine 135 (2023) 102454

Contents lists available at ScienceDirect

Artificial Intelligence In Medicine

journal homepage: www.elsevier.com/locate/artmed

Research paper

A self-supervised algorithm to detect signs of social isolation in the elderly from daily activity sequences

Bardh Prenkaj [a,1], Dario Aragona [a,1], Alessandro Flaborea [a], Fabio Galasso [a], Saverio Gravina [b], Luca Podo [a,1], Emilia Reda [c], Paola Velardi [a,*,1]

[a] Sapienza University of Rome, Italy
[b] DS TECH, Italy
[c] Giomi RSA, Italy

ARTICLE INFO

Keywords:
Anomaly detection
ADL
Elderly social isolation
HyperNN
Hyperbolic uncertainty

ABSTRACT

Considering the increasing aging of the population, multi-device monitoring of the activities of daily living (ADL) of older people becomes crucial to support independent living and early detection of symptoms of mental illnesses, such as depression and Alzheimer's disease. Anomalies can anticipate the diagnosis of these pathologies in the patient's normal behavior, such as reduced hygiene, changes in sleep habits, and fewer social interactions. These abnormalities are often subtle and hard to detect. Especially using non-intrusive monitoring devices might cause anomaly detectors to generate false alarms or ignore relevant clues. This limitation may hinder their usage by caregivers. Furthermore, the notion of abnormality here is context and patient-dependent, thus requiring untrained approaches.

To reduce these problems, we propose a self-supervised model for multi-sensor time series signals based on Hyperbolic uncertainty for Anomaly Detection, which we dub HypAD. HypAD estimates uncertainty end-to-end, thanks to hyperbolic neural networks, and integrates it into the "classic" notion of reconstruction loss in anomaly detection. Based on hyperbolic uncertainty, HypAD introduces the principle of a detectable anomaly. HypAD assesses whether it is sure about the input signal and fails to reconstruct it because it is anomalous or whether the high reconstruction loss is due to the model uncertainty, e.g., a complex but regular signal (cf. this parallels the residual model error upon training).

The proposed solution has been incorporated into an end-to-end ADL monitoring system for elderly patients in retirement homes, developed within a funded project leveraging an interdisciplinary consortium of computer scientists, engineers, and geriatricians. Healthcare professionals were involved in the design and verification process to foster trust in the system. In addition, the system has been equipped with explainability features.

1. Problem statement and motivations

Ambient Assisted Living (AAL) promotes the independent living of older adults thanks to the continuous monitoring of their Activities of Daily Living (ADL), such as sleep, eating, personal hygiene, and socialization. Detecting abnormality in these activities is of paramount importance, especially for the elderly, since it allows prompt intervention by family members and caregivers. With the growing elderly population, numerous studies have focused on preserving the independent living of older people. Factors like pre-existing pathologies or difficulties in interacting with the environment determine the patient's level of independence and influence their quality of life. The progress of technology (e.g., wearable sensors, cameras, smartphones, smartwatches, wireless communications) enables the development of solutions to help older adults continue living independently in their smart homes. Knowledge extracted from these devices can enrich the information delivered to the medical staff and facilitate the early detection of incoming health problems, preventing, in some cases, hospitalization.

State-of-the-art AAL architectures are made of two main components: i.e. IoT devices, such as ambient sensors and wearable devices, and analytic data components, often powered by machine intelligence. The former capture the behavioral and physical signals of monitored patients. The latter process and analyze the generated data streams to provide synthetic descriptions of patients' health status, predict disease evolutions, and capture early signs of future risks. As summarized hereafter, designing effective AAL solutions implies solving several

* Corresponding author.
E-mail address: velardi@di.uniroma1.it (P. Velardi).
[1] Intelligent Information Mining (IIM) Research Group.

https://doi.org/10.1016/j.artmed.2022.102454
Received 1 March 2022; Received in revised form 10 November 2022; Accepted 15 November 2022
Available online 17 November 2022

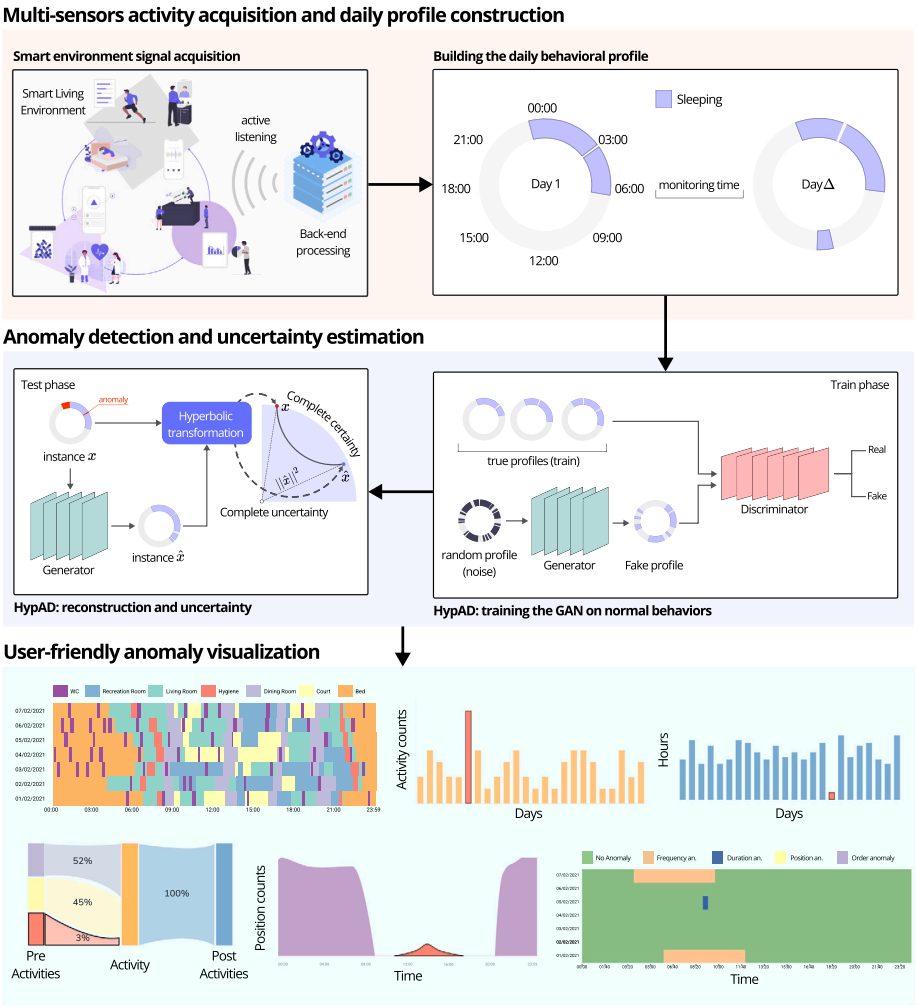


Fig. 1. Teaser: workflow of the proposed solution. (a) Behavioral signals are acquired in an AAL environment; (b) A patient-dependent behavioral profile is learned; (c,d) Anomalous behaviors (w.r.t. the model learned in step b) are reliably detected using an algorithm based on hyperbolic uncertainty, named HypAD; (e) Several anomaly types are detected and shown to the health professionals through a visual medical interface.

open research problems in the IoT and data analytic domains. In this paper, we focus on the specific problem of automated detection of abnormal behaviors and behavioral drifts that sign social isolation in the elderly.

Social isolation is a significant cause of disease in older people [1]. In other words, loneliness and the lack of social interactions may easily lead them to suffer from depression or other serious health problems, such as Alzheimer's. The recent COVID-19 pandemic has increased their discomfort since these diseases most often affect those living in contexts with fewer chances of social contact. Following the COVID-19 data analysis in [2,3], the most vulnerable people in this respect have been those over 70. As shown in [3], they had consistently avoided daily habits that, before the pandemic, had helped them keep active and fit (e.g., physical, well-being, and social activities). Furthermore, their sleep quality has worsened, as reported in [4].

Gerontologists monitor and record specific indicators [5], including lethargy, motor restlessness, reduced interactions, or personal hygiene, to detect early signs of social isolation and depression. However, direct monitoring by health professionals is costly in terms of time and resources and often cannot be conducted systematically. In this context, machine learning, particularly anomaly detection algorithms, may leverage data from multiple monitoring devices in AAL environments to help predict and prevent social isolation.

Although several anomaly detection algorithms have been proposed in the literature (see Section 2), even for the specific application domain of AAL, open problems are related to (i) the type and complexity of input devices used for monitoring, (ii) the need for customized (patient-dependent) solutions and (iii) the reliability and explainability of generated predictions. More in detail:

- Concerning the input sensors, several devices have been used to monitor patients' health status and daily living routines in their homes or residences. In this regard, according to [6], cameras and microphones are excessively intrusive. However, they can automate the monitoring and data collection phases encompassing high-performing action recognition [7] and mood classification [8,9]. From this point of view, ambient sensors (such as pressure, contact switch, and water sensors) and wearable devices are more respectful of the privacy of the elderly. Albeit, anomaly detection from multi-sensor time series signals (rather than from images) is a complex task because they are rare and non-linearly and temporally correlated. Furthermore, detecting anomalies is challenging because one needs to combine several devices to make predictions effectively. For example, one can jointly utilize pressure sensors and smartwatches to ensure the reliable detection of sleep quality and disorders.
- The second issue regards the subjectiveness and context dependency of defining what an anomaly is. The term "context" here refers to a potential anomaly (where, when, and how often) and to the social environment since symptoms of depression can look different depending on the person and their cultural background. Hence, using *untrained algorithms* capable of devising *personalized*

and *context-aware models* is required. As remarked in Section 2, untrained anomaly detection received much less attention in the literature.
- Finally, concerning how the results of anomaly detection algorithms are presented, the literature neglects to assess the quality of the predicted outcomes, namely the *uncertainty* of a prediction. Specifically, automated systems should support and simplify diagnostics and prognostics in health-related scenarios. Hence, trusting the system predictions is crucial to favor health professionals' adoption of automated solutions.

In this work, we propose a novel model based on Hyperbolic uncertainty for Anomaly Detection, which we dub HypAD. We leverage the current state-of-the-art anomaly detection technique [10], which detects abnormal events as those that are more arduous to reconstruct. We use hyperbolic neural networks to map the input and the reconstructed signals into a latent hyperspace in a self-supervised manner. The hyperspace associates each signal with an uncertainty score. Here, the uncertainty tells us whether the reconstruction error is significant because it is anomalous or because the model cannot reconstruct it well. When the model cannot reconstruct the signal well, we can state that the signal is regular but with high model error after training. Additionally, uncertainty is high in this case, and the *anomaly score* of HypAD is down-scaled accordingly.

The paper includes the following contributions:

1. We present a personalized and context-aware self-supervised model for behavioral anomaly detection to identify patient and environment-specific abnormalities.
2. We propose a novel prediction strategy that incorporates the uncertainty of the anomaly detector. As shown in the experimental Section 4, this considerably reduces the variance of model performances across different datasets and anomaly types, thus increasing the robustness of the system and the reliability of its predictions.
3. To favor the adoption of the proposed solution by health professionals, we equipped HypAD with explainability features (see Fig. 11). Here, we highlighted the patient-dependent and contextual features that contributed the most to detecting the anomaly at hand.
4. Our proposal has been integrated into a fully implemented end-to-end AAL system developed as part of a funded project, namely E-Linus. E-Linus aims to detect early signs of social isolation of patients in residences for the elderly and leverages an interdisciplinary consortium of computer scientists, engineers, and geriatricians.

As previously remarked, our proposed solution pays specific attention to providing explainable and reliable predictions in three different ways: (i) we use hyperbolic uncertainty to assess confidence in the predicted outcome; (ii) we use explainability mechanisms to identify and present the contextual *features* that mainly contributed to a detected abnormal behavior; (iii) we have developed a visual medical interface to help to monitor patients' behaviors and validate detected anomalies.

The workflow of our proposed solution is shown in Fig. 1, and described in the next Sections. The paper is organized as follows. Section 2 presents a state-of-the-art survey on several topics related to the one discussed here. Section 3 formalizes the anomaly detection model and discusses the approach details,[2] including the explainability method. Section 4 describes the evaluation testbeds (datasets, compared systems, and performance measures) adopted to analyze the performance of the proposed system. A discussion of experimental results and data analytics is presented in Section 5. Finally, Section 6 summarizes the features and limitations of the current study.

[2] A full implementation of the algorithm is provided for replicability purposes at https://github.com/aleflabo/HypAD.

2. Related work

To the best of our knowledge, this is the first work to combine anomaly detection with uncertainty estimation and the first to propose hyperbolic uncertainty further. Contrarily, explainability in anomaly detection has been treated in several domains, like action recognition in videos [11] and colonoscopy lesion identification [12]. Previous work relates to ours from three main perspectives, which we review here: uncertainty estimation techniques, anomaly detection in time series, and hyperbolic neural networks. Furthermore, we survey specific works related to the application domain of this paper: behavioral anomaly detection. For completeness purposes, we refer the reader to [13] for a literature review on deep learning for anomaly detection.

2.1. Uncertainty estimation techniques

Two different strategies have been proposed in the literature to model uncertainty. The *ensemble-based posterior approximation* uses several weak models to make naive predictions and combine them according to a consensus function into a more complex predictive model [14]. One of the most popular approaches to uncertainty estimation based on ensembles is Monte Carlo (MC) Dropout. It drops neurons on every layer during the training and test phases [15]. *Generative models for uncertainty modeling* use an additional latent variable to make stochastic predictions and evaluate the uncertainty of the model. Generative Adversarial Networks (GANs) [16] play a min–max game where the discriminator distinguishes between real and generated examples. GANs have state-of-the-art performances, and we build on top of that by attaching hyperspace mapping layers to estimate the uncertainty of the model. Another interesting approach to estimating uncertainty is using energy-based models [17,18]. They learn an energy function that models the input and output compatibility. Our method transcends energy-based models because the integrated hyperbolic uncertainty mechanism does not suffer from cold- or warm-start problems, which undermine the training complexity [19].

2.2. Anomaly detection in time series

We identified four methods proposed in the literature for anomaly detection in time series. **Distance-based** outlier detectors consider the distance of a point from its k-nearest neighbors, as in [20,21]. **Density-based** methods, as [22], consider the density of the point and its neighbors. Unlike traditional density-based methods, Li et al. [23] extend deep generative and state space models to achieve robust anomaly detection in case the training set is contaminated with anomalies and other noisy signals. **Prediction-based** methods [24,25] rely on loss functions (e.g., MAE or MSE) to assess the likelihood that the prediction corresponds to an anomaly or not. In [26], the authors use a prior distribution to encode and drive an end-to-end anomaly score learning (Bayesian inverse reinforcement learning). Given an agent that takes a set of sequential data as input, its normal behavior can be understood by its latent reward function. Thus a test sequence is classified as abnormal if the agent assigns a low reward to the sequence. **Reconstruction-based** methods, like [27–31], compare the input signal and the reconstructed one in the output layer, typically using autoencoders. Lu et al. [28] combine a de-noising autoencoder with a two-layered RNN to learn data representations. A similar approach uses bidirectional LSTMs to detect acoustic anomalies [29]. Zhou et al. [31] exploit a feature encoder that copes with annotated data limitations and passes the latent representation to an anomaly score generator. These methods assume anomalies are difficult to reconstruct and are lost when the signal gets mapped to the lower dimension; thus, a higher reconstruction error implies a higher anomaly score. Recently, GANs [16] have been employed to detect anomalies in time series data. Zheng et al. [32] leverage the shortcomings of GANs and modify the generator to produce matching data instances to the training data.

However, it is difficult to guarantee that the generated reference instances resemble the unknown anomalies, leading to unstable anomaly classification performances [33]. TadGAN [10] uses a cycle-consistent GAN architecture with an encoder–decoder generator and proposes several ways to compute reconstruction error and its combination with the critic outputs. We build on top of TadGAN's architecture by incorporating the hyperbolic mapping layer into the reconstructed time windows to assess the uncertainty of the detector.

2.3. Hyperbolic neural networks

Deep representation learning in hyperspaces has gained momentum after the pioneering work of hyperNNs [34] that generalizes Euclidean operations (e.g., matrix multiplications) to their counterparts in hyperspace. The authors propose analog counterparts in the hyperspace of neural network components, such as fully connected (FC) layers, multinomial logistic regression (MLR), and recurrent neural networks. Furthermore, methods like Einstein's midpoint [35] and Fréchet's mean [36] propose different ways of aggregating features in hyperspace. The work in [37] extends hyperNN and proposes Poincaré split and concatenation operations, generalizing the convolutional layer to hyperspace. The authors in [38,39] propose hyperbolic graph neural networks leveraging hyperNNs.

Thus formulated, hyperNNs have mainly been adopted to improve performance by leveraging hierarchies and uncertainty in zero-shot learning [40], re-identification [41], and action recognition [42]. Of particular interest for our domain, Suris et al. [43] have leveraged hyperNNs to model a hierarchy of actions from unlabeled videos. To the best of our knowledge, this is the first work to have applied hyperNNs for sequence modeling with the goal of anomaly detection.

2.4. Behavioral anomaly detection techniques

Abnormal activity detection in ambient assisted living (AAL) environments has recently attracted the attention of researchers [44–46]. Although the majority of research results in anomaly detection have been published in the more general context of time-series mining – see Section 2.2 – works specifically concerned with AAL, such as [47–49], highlight several critical challenges of anomaly detection in this domain. Zhang et al. [47] consider the challenging problem of learning from untrained data, where abnormal events are mixed with non-normal data. If anomalies are not rare, the model can be misled, making it hard to distinguish between standard, abnormal, and noisy data. To solve the problem, they first use Convolutional Autoencoder to characterize the spatial dependence of multi-sensor data with a Maximum Mean Discrepancy (MMD) to better distinguish between the noisy, regular, and abnormal data. Next, they use a Memory Network consisting of linear (Autoregressive Model) and non-linear predictions (Bidirectional LSTM with Attention) to capture temporal dependence from time-series data. We note that this method has been designed for contexts where anomalies are rare *instances*[3] rather than anomalous alterations in the characteristics of a specific attribute in a particular instance (e.g., a sleep of particularly long duration for a given patient or carried out at an unusual time concerning the habits of the elderly), which is the focus of our work. Deep et al. [50] found that multi-sensor-based activity recognition is the best technology to address the needs of anomaly detection in elderly care; they report the following: robustness to environmental changes, privacy-preserving, and ease of use. Multi-sensor solutions use radio, motion, contact, pressure, and door. These sensors generate signals when there is an interaction between the person and the objects in the environment, which are then associated with specific activities (for a general survey of activity recognition techniques, see [51]; for activity recognition for monitoring healthcare, see [52,53]). Therefore, multi-sensor behavioral data are multinomial since every input represents one out of several possible actions, and multivariate since actions are described by many attributes, such as start time and duration. We note that anomaly detection methods have rarely been tested in multivariate contexts, which are more challenging.

The work more closely related to ours is by Dahmen and Cook [54], who capture anomalies in semantically-labeled sequences of ADL on patients with neurodegenerative disorders among whom Parkinson's disease where anomalous events (i.e., falls) are frequent symptoms [55]. The authors propose an Indirectly Supervised Anomaly Detection (ISUDRA) to improve standard unsupervised models for anomaly detection in time series of geolocation sensoristic data. ISUDRA exploits a small number of labeled instances to direct the choice of unsupervised learning parameters. It employs Bayesian optimization to select time scales, features, base detector algorithms, and algorithm hyperparameters increase true positive and decrease false positive detection.

As reported in [56], there are two strategies to recognize behavioral changes: profiling and discriminating. In profiling, a model of "normality" is trained, and new input data is compared with the model. The behavior is considered an anomaly if it deviates from the learned data distribution during training. In the discriminating strategy, anomaly data are defined according to previously collected or manually defined (e.g., based on medical knowledge) anomalous events. Unfortunately, as reported in Section 4.1, manually annotated multi-sensor data are scarce. Although synthetic data can be generated according to some agnostic mechanism (see, e.g., [57]), anomalies in this domain are highly contextual. To overcome this difficulty, we started with real-life patient data collected in residences for older people in our work. Next, we asked clinicians to insert natural anomalies using a visual interface developed for this purpose. Since our solution is self-supervised (based on profiling patients' normal behaviors), the annotated dataset is used solely for testing purposes.

3. Methodology

This section describes the proposed method for modeling and detecting anomalous events in ADL sequences engendered by patients. First, we provide the reader with a mathematical formalization of the input model (see Section 3.1). Then, we explain our anomaly detection model (see Section 3.2).

3.1. Input modeling of multivariate time series

Without loss of generality, the ADL of a specific patient p can be represented as a series of activities, where each activity takes place at a unique timestamp. We can model a time series as a sequence of random variables x_t ordered in time $t \in \mathcal{T} \subseteq \mathbb{R}$ and dependent on each other. Notice that time series can also correspond to infinite streams of data (i.e., $\mathcal{T} = \mathbb{R}$); however, in this work, we focus only on discrete time series [58] where the ADL sequence has a limited time[4] interval $\mathcal{T} = [b(p), f(p)]$ such that $b(p)$ and $f(p)$ denote the beginning and the end of the monitoring time for patient p, respectively. A multivariate time series $\mathcal{X}$ – hereafter simply time series – is a sequence of multiple variables $x^1, \dots, x^m$. Let $\mathcal{M}^{\mathcal{T},\mathcal{X}}$ be a time-indexed table of events where $\mathcal{T}$ denotes the discrete set of timestamps and the elements of the table belong to the random variable set. As a result, a multivariate time series is a time-indexed table $\mathcal{M}^{\mathcal{T},\mathcal{X}}$ where $|\mathcal{X}| > 1$.

In our context, we have an ordered sequence of activities[5] $\mathcal{A} = \{a_1, a_2, \dots, a_{|\mathcal{A}|}\}$, where each a_i $\forall i \in [1, |\mathcal{A}|]$ is characterized by a label identifying the action and the time interval $[b(a_i), f(a_i)]$ in which it

[3] Actions performed occasionally, or ECG signals of patients affected by rare diseases.

[4] For commodity purposes, we represent timestamps as Unix time.

[5] Here, we consider only simple activities. Otherwise, composite activities such as "reading a book while sitting" have the same timestamp of execution.

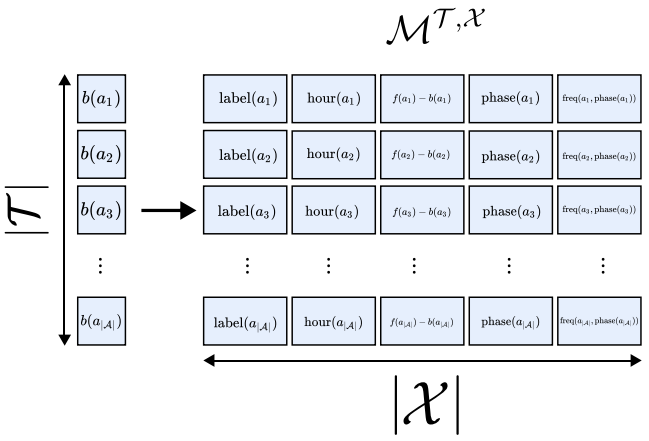


Fig. 2. Input modeling of an activity sequence to get a multivariate time series.

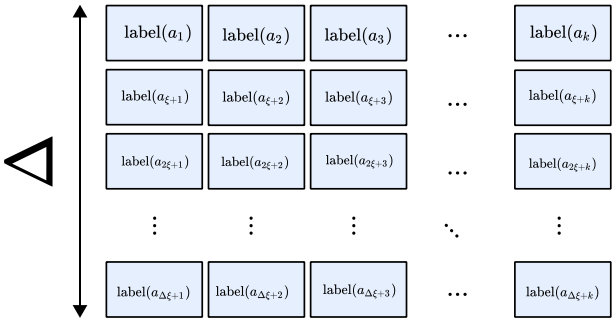


Fig. 3. Result of the time series decomposition using the sliding window approach. The image shows only the structure of the part corresponding to the activity labels in $\mathcal{M}^{\mathcal{T},\mathcal{X}}$. The same reasoning can be applied to the other features in $\mathcal{X}$.

happens. We also assume that there are no time-gaps in p's behavioral sequence (i.e., $\forall a_i, a_{i+1} \rightarrow f(a_i) = b(a_{i+1})$).

We use the following features to represent our input:

- The *activity label* identification from the event in $\mathcal{A}$. We represent the label of a_i with label(a_i).
- The *beginning time* of the considered activity - i.e., hour($b(a_i)$) where hour($*$) extracts the hours and minutes in which the input activity begins.[6]
- The *duration* of each activity $a_i \in \mathcal{A}$ - i.e., $f(a_i) - b(a_i)$.
- The *phase of the day* during which a_i is performed (i.e., phase(a_i)). This feature can be used to detect anomalous subroutines that are related to the phase of the day. In our implementation, we divide the day in four equal parts: i.e., morning, afternoon, evening, and night.
- The *frequency* of each activity a_i happening in phase(a_i). We denote this feature with freq(a_i, ρ) where ρ represents the phase where we count the frequency.

This process is described in Fig. 2 and gives as output the multivariate time series $\mathcal{M}^{\mathcal{T},\mathcal{X}}$.

As proposed in [59–63], we split the created ADL sequence into multiple subsequences according to a sliding window approach. This is crucial to detect also anomalies related to the *order* in which the activities are performed. Thus, we group the events generated throughout the monitoring interval into fixed-length event sequences where each of them can span over an arbitrary amount of time and overlap with the previous by a predefined step size ξ. In this way we transform each

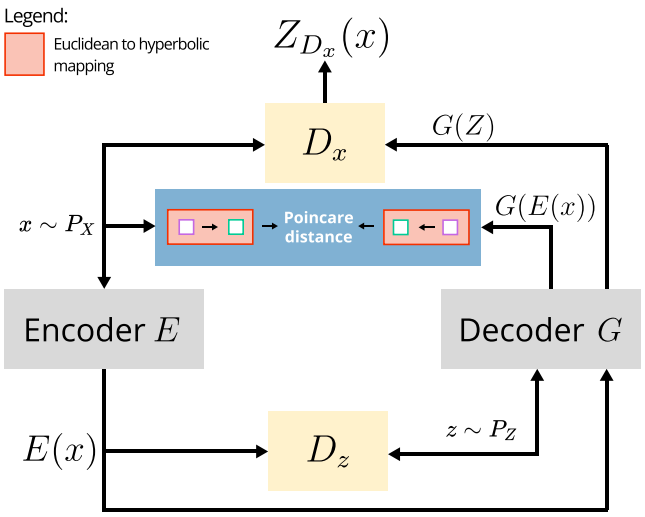


Fig. 4. Overall architecture of our proposed model HypAD. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

of the time series in $\mathcal{X}$ into sequences $\mathcal{S}_1, \ldots, \mathcal{S}_\Delta$ s.t. $\Delta = \lfloor \frac{|\mathcal{T}|-k}{\xi} \rfloor + 1$ having the same length k, where k represents the number of events to be stored in a sequence. An example of the result of this process is shown in Fig. 3, considering only the time series with activity labels for visualization purposes. Hence, the shape of the tensor given in input to our model is $(\Delta \times k \times 5)$, where the last dimension accounts for the described features above.[7] Finally, we normalize the features in [0,1], normalizing duration and frequency values w.r.t to all durations and frequencies of the same activity type in the monitoring period.

3.2. Prediction strategy

We propose a novel model for anomaly detection in time series based on hyperbolic uncertainty. HypAD is a reconstruction-based model that minimizes the reconstruction loss, given by measuring the hyperbolic distance between the input signal and its reconstruction. In hyperbolic space, errors are exponentially more significant when predictions are certain. Therefore, HypAD tends to predict either *certain correct reconstructions or uncertain, possibly mistaken reconstructions*. Fig. 4 illustrates the proposed HypAD. It integrates the machinery of hyperbolic neural networks into the reconstruction-based architecture of TadGAN [10] (see Section 2). In HypAD, the input signal x is first passed through an encoder, then followed by a decoder sub-network. The output of the decoder $G(E(x))$ as well as the original signal x are mapped to the hyperspace, shown as the dotted red edge box with red background. As in [41,43], we adopt the Poincaré ball model of hyperspaces.

Once x and $G(E(x))$ are mapped to the Poincaré ball, we use a hyperbolic feed-forward layer [34] to estimate the corresponding hyperbolic embeddings h and $\tilde{h}$. Finally, the two hyperbolic embeddings are compared using the Poincaré distance. Note that the same reconstruction error function $RE(x)$ is used at training and inference times. A key property of the Poincaré ball is that the distance between two points exponentially grows as we move away from the origin, as shown in Fig. 5. Hence, an erroneous reconstruction towards the circumference is penalized exponentially more than an erroneous reconstruction close to the center. This leads to the useful tendency of HypAD to either predict a matched reconstruction ($\tilde{h}$ and h are close by) or an unmatched

[6] We transform the beginning time into a rational number where the whole part depicts the hours in [0,23] and the decimal digits are in [0,99].

[7] Note that *order* is an implicit feature provided by what precedes and what follows a given action in the sequence.

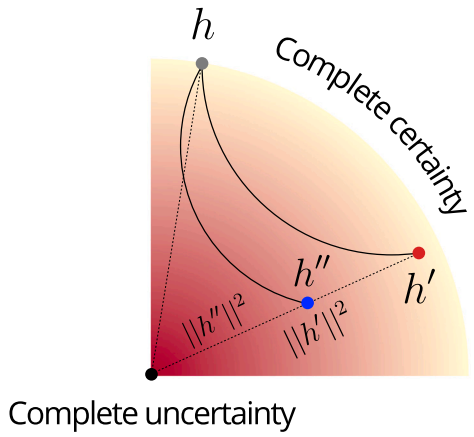


Fig. 5. In the illustration, the colored circular sector represents the hyperbolic Poincaré ball, where the radius distance of data embeddings is their degree of certainty (points on the circumference are most certain). $\boldsymbol{h}$ is the hyperbolic mapping of the input signal, which HypAD attempts to match by the reconstruction $\boldsymbol{h}'$. Thanks to hyperbolic neural networks and their exponentially larger penalization for errors at high certainty, HypAD learns to prefer signal reconstructions such as $\boldsymbol{h}''$, i.e., with the same amount of error as $\boldsymbol{h}'$ (the same angle and cosine distance) but smaller radius $\|\boldsymbol{h}''\|^2$ and thus higher uncertainty. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

reconstruction towards the origin ($\widetilde{h}$ and h are far-away, $\|h\|^2$ and $\|\widetilde{h}\|^2$ are small). Hence, the distance of the reconstruction to the origin provides a realistic estimate of the model's uncertainty, referred to as hyperbolic uncertainty, $U(X) = 1 - \|\widetilde{h}\|^2$. The smaller the distance from the origin, the more uncertain the model is. The model's reconstruction errors and critic scores are first normalized by subtracting the mean and dividing by the standard deviation to estimate the anomaly score. The normalized scores $Z_{RE}(x)$ and $Z_{D_x}(x)$ and the hyperbolic uncertainty $U(X)$ are combined to get the anomaly scores:

$$s_u(x) = Z_{RE}(x) \odot Z_{D_x}(x) \odot (1 - U(x)) \tag{1}$$

The convex integration of the model certainty reduces the score when HypAD is less confident on the reconstruction.

3.3. Explaining anomalies

Explainability is becoming a relevant topic in AI, especially in high-risk domains and cases where a person's health and well-being could be compromised. Although attention mechanisms have been amongst the first methods to explain a model's prediction, the authors in [64] have argued that they may fail to indicate the importance of the input about a prediction. More recently, Lundberg and Lee [65] introduced a new way of explaining and interpreting models' outputs using the game-theory framework SHAP (Shapely additive explanations). Antwarg et al. [66] have adapted SHAP to explain the predictions of unsupervised anomaly detectors. If it exists, the authors note that an anomaly resides in the values of the input, and the explanatory model needs to explain why this instance was not predicted (reconstructed) well. Accordingly, the proposed method focuses on the correlation between the features with high reconstruction error and those most influential in affecting it.

We decided to adapt SHAP to our proposed model HypAD because of its ability to estimate every feature's importance for a specific prediction and because its outcome is well aligned with human intuition. As a further advantage, SHAP is model-agnostic.

4. Experiments and results

This Section describes the benchmark datasets and the experimental setup to test the performance of the proposed method.

4.1. Behavioral anomaly detection benchmarks

To the best of our knowledge, very few behavioral datasets have been made available in the literature. The authors in [67] present the CASAS benchmark,[8] from which we extract the HH-set, which consists of 30 separate test beds containing data collected over several monitoring times. These datasets contain daily activities performed by patients monitored in controlled environments (apartments) with passive infrared motion and magnetic door sensors.

The eHealth Monitoring Open Data Project[9] includes two main sets of activities monitored for a single patient for one year. The project's original goal [68] is to assess the Functional Autonomy Measurement System (SMAF), a clinical rating scale to measure the functional autonomy of elderly patients. The data structure is similar to CASAS, where each instance in the sequence is an activity performed by the patient, characterized by a begin and end timestamp. The eHealth dataset has been generated synthetically. To generate realistic sequences, they use a pseudo-Markovian model where the sequence generation follows the transition probabilities under certain constraints to avoid the possible generation of less probable sequences.

4.2. The E-Linus smart living environment

Early detection of social isolation in the elderly has been addressed in a project funded by the Lazio Region (centre Italy) called E-Linus. E-Linus is an Active and Independent Living solution for elderly residences, which allows the indoor recognition of symptomatic behaviors of senile social isolation and activates home care protocols and services for professional and family caregivers, thus supporting the synergies of Integrated Home Care. In Fig. 6, we provide a high-level description of the E-Linus solution[10]:

- **The Smart Living Environment** (SLE) consists of multiple beacons[11] in the interest areas. A smartwatch[12] is used to localize the patient while monitoring their vital signs ranging from heart rate to sleep patterns. Finally, a voice recognition component is exploited to classify daily moods as anger, fear, joy, sadness or neutral emotion. This component is used under a caregiver's control during specific times of the day to limit its intrusiveness.
- The **back-end layer** processes the signals and creates ADL time series for each patient. The monitoring devices processed in this layer recognize the following activities: Sleep, WC, Hygiene, Dining Room, Recreational Room, and Garden.
- The **middle layer** performs descriptive, predictive, and prescriptive analysis of ADL sequences and other patient data, including clinical records and vital parameters. It includes the anomaly detection algorithm proposed in this paper.
- The **front-end layer** Virtual Assistant (VA) that proposes a series of engaging activities (e.g., listening to music) to sustain the patient's well-being. It also includes an interactive visual analytic interface for the caregivers [69] reporting daily activities, patient statistics, and detected anomalies. The VA is a relevant feature since visual insight may greatly facilitate the identification of health-risk scenarios, as remarked in [70].

Six volunteering elderly patients with different health conditions have been monitored for six weeks, according to the timeline established in the E-Linus project. We note that, although six patients may seem a small number, the focus here is on *personalization*: the

[8] http://casas.wsu.edu/datasets.
[9] https://sourceforge.net/projects/ehealthmonitoringproject/files.
[10] A detailed description of the E-Linus project is outside the purpose of this paper. Here, we describe the process of data collection and experimental validation of anomaly detection for detecting signs of social isolation.
[11] https://accent-systems.com/product/ibks-plus/.
[12] TicWatch Pro 3.

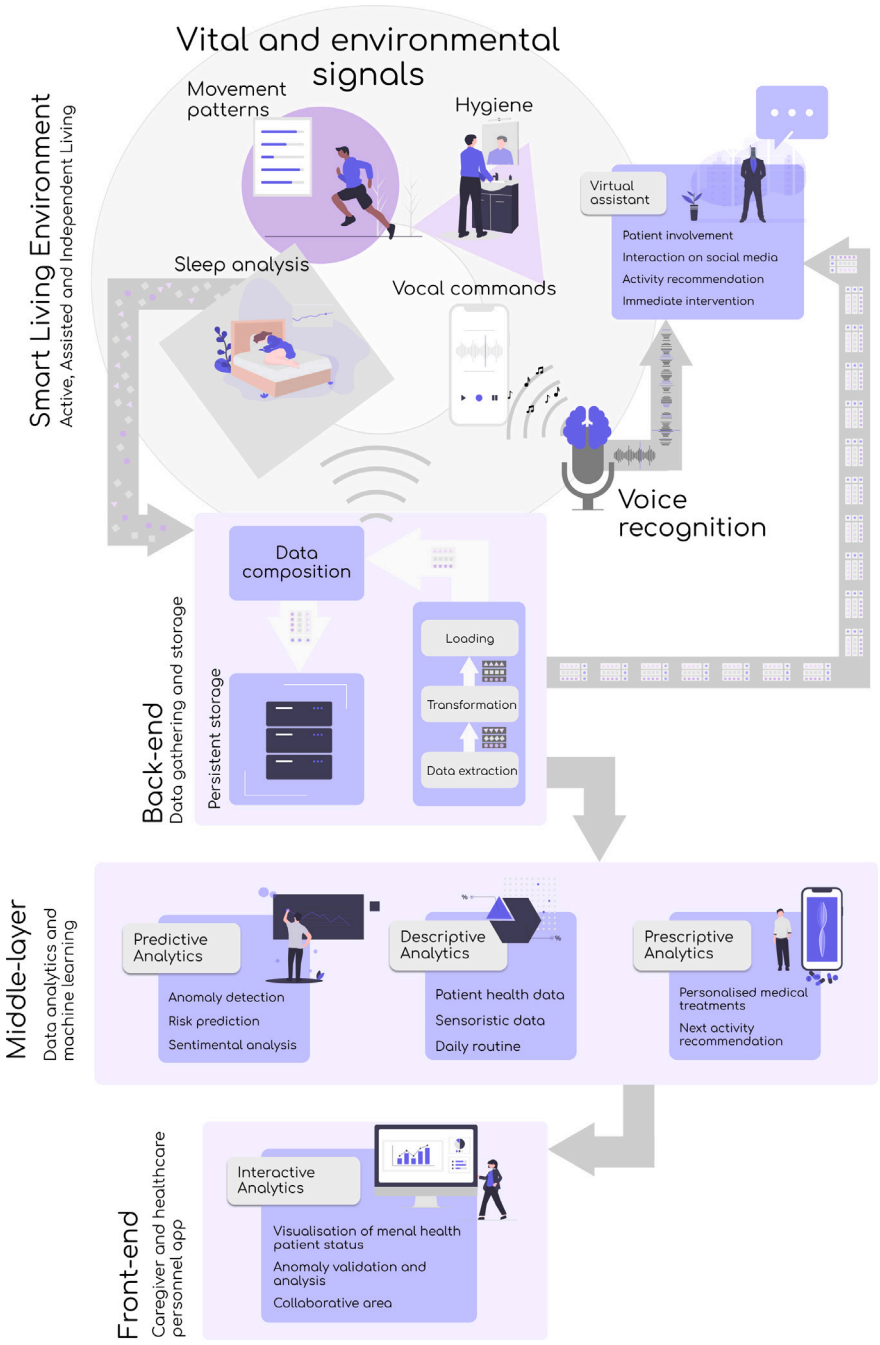


Fig. 6. Architecture and workflow of the E-Linus system.

challenge is to learn a model of normality and abnormality tailored to each patient. From this point-of-view, each patient is to be considered an independent dataset, as better argued in Section 5. Furthermore, given the relatively short monitoring period, we artificially extended these sequences over a longer period (1 year) based on small realistic perturbations of the observed routines.

4.3. Strategy to inject realistic synthetic anomalies

The datasets enlisted in Sections 4.1 and 4.2 do not contain annotations of anomalous events. Monitoring patients for long periods does not guarantee the observation of abnormal behaviors. Even when they occur, abnormal events are generally few and insufficient to test the performance of an anomaly detector. Following the approach in [71], we inject synthetic anomalies but pay specific attention to introducing realistic deviations from standard behaviors. According to [72], personal hygiene, sleep habits, and peer interactions can hide the most critical signals for identifying abnormal behavior in the elderly, especially concerning their psychological status. With the support of gerontologists, we identified and generated the following five activity-related anomalies[13]:

- **Sleep anomalies** - We increase the sleep duration per day to simulate symptoms of depression and other diseases. Additionally, we introduce several sleeping interruptions during the night to represent sleep disorders.
- **Recreation anomalies** - We analyze the trend of patients that hang out in the court/recreational area with their peers and

[13] https://github.com/dars16/routineanomaly.

Table 1
Dataset summary.

	CASAS	E-Linus	eHealth
Number of monitored activities	9	7	14
Monitoring period (avg. days per patient)	396	367	366
Avg. anomalous days in test set	35 ± 14.98	31 ± 9	32 ± 7
Avg. num. of anomalous activities	217.5 ± 113.8	144 ± 54.5	139 ± 73

increase the time they spend in their rooms while simultaneously decreasing their stay in the common areas to simulate signs of social isolation.

- **Food anomalies** - We tamper with the everyday habits of patients visiting the dining area to have breakfast, lunch, snacks, and dinner. We reduce their time in this room to simulate signs of eating disorders.
- **WC anomalies** - We augment the duration of a patient's toilet usage to catch various symptoms of several pathologies and the increased frequency of toilet visits to simulate symptoms of diabetes.
- **Hygiene anomalies** - We decrease the frequency of taking showers and having baths to represent the lack of self-care throughout the week.

Instead of programmatically modifying the sequences to inject those anomalies, we have developed a visual interface tool[14] [69] used by the caregivers to insert *artificial yet realistic* anomalies for each patient based on their daily routine. The tool allows caregivers to visually inspect each patient's usual behavior and statistics and inject four subtypes of anomalies for each monitored activity, altering their *duration, frequency, start time* and *order*.[15] Regarding WC and hygiene, we have decided to inject only *duration* and *frequency* anomalies due to the high variability of *start time* and *order* characteristics of those kinds of activities. This would lead to a highly challenging detection.

While tweaking the ADL for each patient, it is important to keep the initial period of the sequence untouched such that the models are able to learn its latent structure and the relationship between the activities therein as normal behavior. In this way, the initial monitoring period (*without injected anomalies*) can be used as the training set, while the remaining part as the test set. In collaboration with geriatricians, we have defined the train set as the sequences occurring in the first 90 days of monitoring. The subsequent days are considered as the test set and tampered with anomalous behaviors. Using the techniques described above, we have generated four anomalous test sets involving bed, recreation, and food activities, and two anomalous test sets for WC and hygiene. This is done for each of the three datasets, returning 16 different test sets for each domain.

Table 1 provides summary statistics. As previously described, patient-specific abnormal behaviors were incorporated for all datasets into the 9-month sequences of the test set. Furthermore, note that anomalies have been incorporated only in actions related to the five previously listed types (sleep, recreation, dining room, WC, hygiene).

4.4. Evaluation

Metrics: In the context of anomaly detection in behavioral sequences, anomalous instances represent a small minority, thus skewing the class distribution of the datasets. Metrics like accuracy are not informative since we can obtain perfect performances by predicting the majority class (normal behaviors). Therefore, we consider evaluation measures agnostic to the class imbalance phenomenon, like the F1 score. Furthermore, to align our work to [54], we also compute the geometric mean (G) of the product of precision (P) and recall (R) - i.e. $\sqrt{P \times R}$.

Baselines: We compare five models:

- AE [73] - We use a five-layer fully-connected autoencoder. We use dropouts [74] in each layer with probability α as a means to account for overfitting.
- ConvAE [75] - We have two layers of mono-dimensional convolutional encoding interleaved with max pooling. The decoder has a specular composition as the encoder where the de-convolution is aided by up-sampling layers. We use a dropout rate of α in all layers.
- IForest [76,77] - We rely on an isolation forest that separates anomalous instances from the normal samples. In order to isolate a data point, the algorithm recursively generates partitions on it by randomly selecting an attribute a and then randomly selecting a split value for a, between the minimum and maximum values allowed.
- ISUDRA [54] - As introduced in Section 2.4, we re-implement the algorithm, but we maintain a fixed sliding window of 30 signals to align with the strategy proposed in [43].
- LstmAE [78] - We use a two-layer stacked LSTM encoder. The hidden vector of the first LSTM layer in the encoder gets passed to the second LSTM layer. The latent representation of the encoder gets then reconstructed in reverse order from the decoder. As in AE and ConvAE, we use a dropout rate of α in each layer of the architecture.
- TadGAN [10] - We use a one-layer bidirectional LSTM for the generator E, and a two-layer bidirectional LSTM for G. For the critic D_z we use a fully connected layer, and two dense layers for D_x.

Implementation details: To favor reproducibility, we include the details of each implementation. For the autoencoder baselines, we set the number of epochs to 30, the batch size to 32, and the learning rate to 10^{-3}. For TadGAN, we set the epochs to 30, the batch size to 64, and the learning rate to 5×10^{-4} and the iteration for the critic to 5. We use Adam as the optimization function to train all the baselines. We use a Bayesian optimization for all the autoencoder-based[16] baselines. For AE, we optimize the number of neurons in the two layers of the encoder in the set $\{8, 16, 32, 63, 128\}$. We fix the number of neurons in the bottleneck layer at 4. We achieved the best performances for AE with the number of neurons set to 16, 8, 4, 8, and 16 for all five layers, respectively. For ConvAE, we optimize the number of filters of the three encoder layers in $\{2, 4, 8, 16, 32, 64, 128\}$, whereas the decoder has a specular architecture. We set the kernel size to 2 with no padding on all layers and a stride of 1. We reached the best performances for ConvAE with the number of filters set to 8 and 4 in the encoder layers, respectively. For LstmAE, we select the optimal combination of the number of hidden vector units in the set $[4, 64]$ with a step of 2. We reached the best performances with the number of hidden units equal to 16 and 8 for the two encoding layers, respectively. We used ReLU as the activation function for all these methods. We performed a Bayes optimization over the dropout rate α for all the above methods

[14] The tool is accessible accompanied by a sample daily profile extracted from one of the E-linus patients at anomalybyclick.github.io.

[15] An order anomaly happens when the usual sequence of actions is altered, for example, going to bed immediately after dinner without going to the recreation room or toilet first.

[16] Notice that we optimize only the number of neurons in the first two layers (encoder). The neuron number in the decoder is specular to that in the encoder.

Table 2
Comparison of the performances of different state-of-the-art systems on three behavioral datasets. We rank the strategies in ascending order w.r.t. G ($\mu \pm \sigma$) and F1 ($\mu \pm \sigma$). The bold-faced numbers represent the best-performing strategy among all. We calculate the mean and standard deviation across all datasets for each enlisted strategy. For completeness purposes, we include HypAD without uncertainty (i.e. HypAD w/o $U(X)$) to perform an ablation study w.r.t. TadGAN and our proposed method, HypAD, that by default incorporates the uncertainty mechanism $U(X)$.

	CASAS		E-Linus		eHealth		G ($\mu \pm \sigma$)	F1 ($\mu \pm \sigma$)
	G	F1	G	F1	G	F1		
IForest	0.281	0.273	0.368	0.353	0.270	0.262	0.306 ± 0.044	0.296 ± 0.041
LstmAE	0.244	0.220	0.446	0.430	0.388	0.366	0.359 ± 0.085	0.339 ± 0.088
ISUDRA	0.315	0.294	0.488	0.470	0.405	0.396	0.403 ± 0.071	0.387 ± 0.072
ConvAE	0.321	0.297	0.524	0.516	0.461	0.452	0.435 ± 0.085	0.422 ± 0.092
AE	0.265	0.280	0.514	0.505	**0.586**	**0.577**	0.455 ± 0.138	0.454 ± 0.126
TadGAN	0.387	0.371	0.623	0.615	0.547	0.539	0.519 ± 0.098	0.508 ± 0.102
HypAD w/o $U(X)$	**0.512**	**0.510**	0.595	0.593	0.579	0.540	0.562 ± 0.044	0.548 ± 0.042
HypAD (proposed)	**0.512**	**0.510**	**0.631**	**0.629**	0.579	0.540	**0.574 ± 0.06**	**0.560 ± 0.06**

Table 3
Performances of all methods in terms of average F1 scores for each sub-type of anomaly in the E-Linus dataset. The bold-faced numbers represent the better-performing algorithm for that particular anomaly type. The average and standard deviations are calculated across all anomaly types for each enlisted method.

	IForest	ISUDRA	AE	ConvAE	LstmAE	TadGAN	HypAD
Duration	0.342	0.520	0.624	0.594	0.572	0.651	**0.669**
Frequency	0.437	0.519	0.543	0.561	0.523	0.652	**0.652**
Start Time	0.376	0.550	0.614	**0.635**	0.386	0.575	0.596
Order	0.208	0.227	0.135	0.195	0.083	0.531	**0.575**
F1 ($\mu \pm \sigma$)	0.341 ± 0.097	0.454 ± 0.152	0.479 ± 0.232	0.496 ± 0.203	0.391 ± 0.220	0.602 ± 0.061	**0.623 ± 0.045**

in $[0, 0.95]$, and, interestingly, obtained the best performances for all with $\alpha = 0.2$. Meanwhile, we optimized the maximum number of samples $n_s \in [10^2, 10^4]$ with a step of 10^2, and the number of estimators $n_e \in [0, 10^3]$ with a step of 5×10^1 for IForest. Here, we reached the best results with $n_s = 2.6 \times 10^3$ and $n_e = 4.5 \times 10^2$.

For our proposed method HypAD, we took inspiration from an online available PyTorch implementation.[17] We incorporate a hyperbolic transformation[18] following the work of Suris et al. [43] on top of the original architecture of TadGAN. Our hyperbolic component enables us to elucidate the contribution of hyperbolic neural network to the overall boost in performances w.r.t. TadGAN, which bases its computations in Euclidean vector spaces. In other words, by tweaking TadGAN's architecture to embed vectors and exploit transformations in the hyperbolic space, we can perform exhaustive studies of the benefits of the hyperbolic network (see the discussion on Tables 2 and 3). HypAD's hyperparameters remain the same as in [10], and we use the Riemannian Adam[19] optimization function.

5. Discussion

Table 2 shows our experimental results. The shown metrics are computed for each dataset. Additionally, we average the metrics over all datasets to understand each method's overall performance. To adequately comment on the results in the Table, it is helpful to summarize the main differences between the three datasets: CASAS includes natural activity sequences of older adults freely moving in their home environment, eHealth includes artificial sequences generated as previously described, while E-Linus includes sequences obtained by artificially extending real but short sequences. Furthermore, in E-Linus, patients live in a retirement home, where their activities are regulated by a predefined routine established by the caregivers, while CASAS shows more irregular behaviors in the daily routines. Notice that, on average, our proposed method outperforms the baselines considered. Specifically, HypAD and TadGAN both show remarkably better performance in all experiments concerning the baselines and the best competitor system in this domain (ISUDRA). In CASAS, HypAD exhibits the highest increment in performance. Although the difference between HypAD and TadGAN is not striking in the other two datasets, and one case (eHealth), TadGAN performs slightly better, HypAD has a lower standard deviation w.r.t. TadGAN and all the other methods (< 0.049) across all experiments, which demonstrates its higher robustness and trustworthiness, boosted by the proposed hyperbolic uncertainty model. To explain the different increment in performance of HypAD compared to the other systems in the three datasets, we note that a synthetic dataset such as eHealth is probably one where the model can tend to estimate with greater certainty, and this nullifies the contribution of HypAD's multiplicative uncertainty term, which explains the comparatively lower performance in this domain.

To analyze the contribution of the hyperbolic transformation and the benefits of the uncertainty mechanism, we perform an ablation study (see the lower part of Table 2) according to the different components of our proposed method. Recall that TadGAN can be considered as HypAD without hyperbolic transformation and without uncertainty. In the Table, we also perform experiments by adding the hyperbolic component to TadGAN – i.e. HypAD w/o $U(X)$ – without the uncertainty and demonstrate that it performs better than TadGAN. Finally, our proposed method integrates both the hyperbolic transformation and the uncertainty mechanism that contribute to an overall amelioration of the performances. Notice that the Table highlights that the uncertainty component $U(X)$ provides no gain for CASAS and eHealth because the model is certain for all predictions – i.e. it maps the instances near the circumference of the Poincaré ball – thus not altering the final score. Contrarily, in E-Linus, the uncertainty boosts performances, uncovering that the predicted anomalous in HypAD w/o $U(X)$ are, in reality, sequences of everyday actions that are challenging to reconstruct.

The histograms of Fig. 7 show the performance (in terms of F1 and G) of all systems on each monitored activity. Again, we can see that TadGAN and HypAD have stable performance on all actions and considerably surpass the other systems. Similar observations apply to Table 3, illustrating the performance of all systems by type of anomaly.[20]

Finally, we study the effect on performances of a variable amount of abnormal behaviors. Since we are in an unsupervised scenario,

[17] https://github.com/arunppsg/TadGAN.
[18] https://github.com/cvlab-columbia/hyperfuture.
[19] https://github.com/geoopt/geoopt.

[20] Notice that the average does not precisely match with the one presented in Table 2, since the distribution of the observed phenomena is not the same: i.e. the average is not weighted by the frequency of each anomaly type in the dataset.

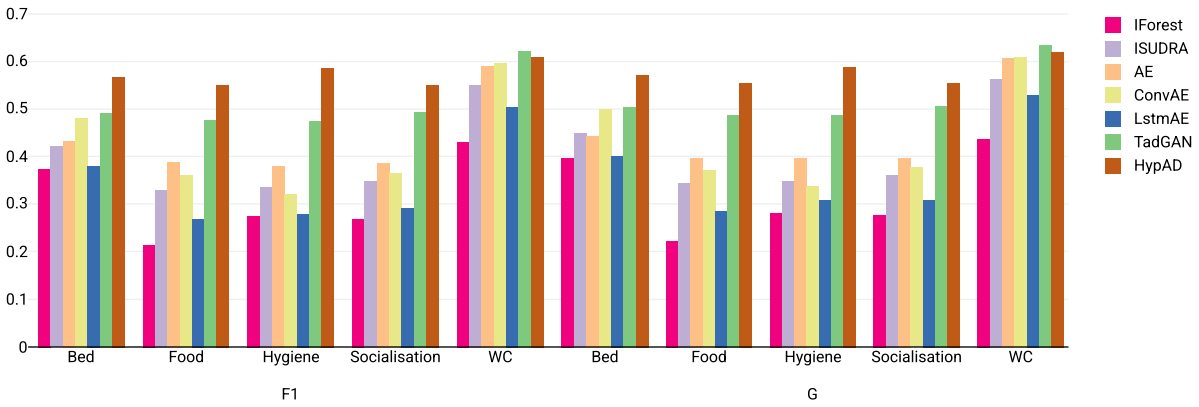


Fig. 7. Average F1 and G scores for each action in all the dataset.

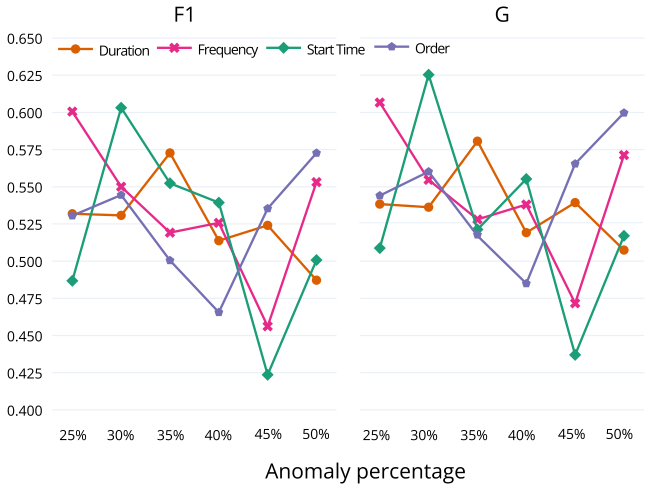


Fig. 8. The change in F1 (left) and G (right) scores for each anomaly type. Here, we change the percentage of anomalous instances in the test set and measure the performances.

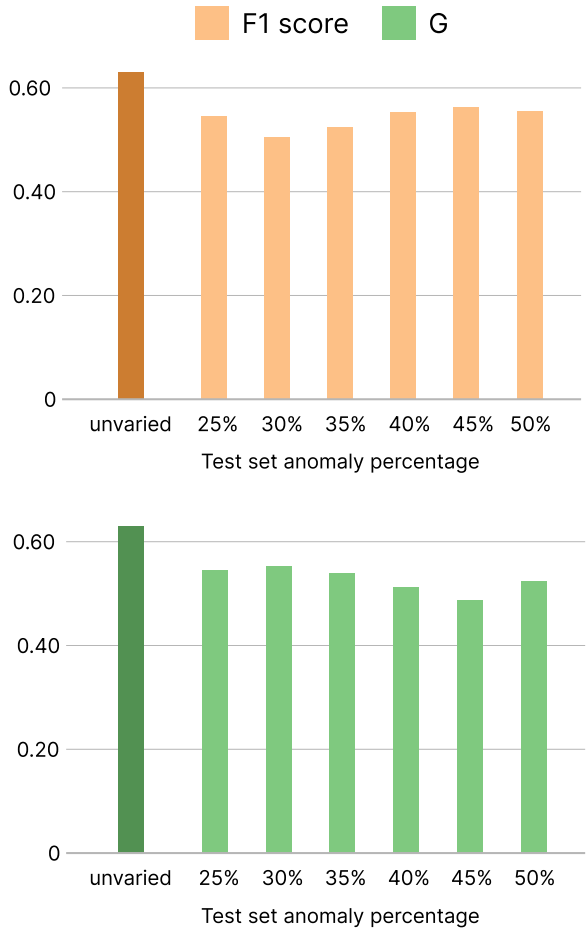
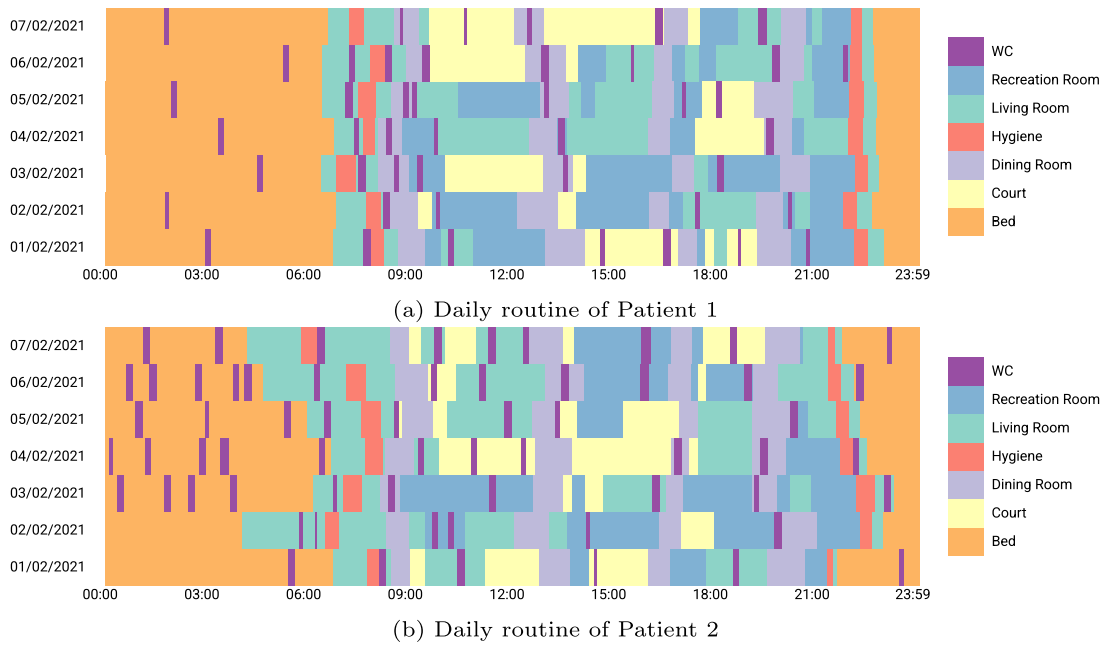


Fig. 9. HypAD's overall performances in terms of F1 and G scores when changing the percentage of anomalies in the E-Linus dataset. Notice that we compare the performances per anomaly percentage with the original ones reported in Table 2, presented with a darker color than the rest. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

we cannot vary the percentage of anomalies on the entire dataset. Therefore, we leave the training untouched for each anomaly type and study the performance change when the anomaly percentage changes in the test set. Fig. 8 illustrates the performances of HypAD on each anomaly type in E-Linus. We vary the percentage change from 25% to 50%, where 50% represents a class-balanced scenario. Clearly, 50% of anomalous instances is the maximum amount to remain in an anomaly detection task since anomalies should represent only a fraction of the dataset. Notice in Fig. 8 that the curves corresponding to the F1 scores (left) share similar trends with those presented in the G score (right). Additionally, the curves for all anomaly types have a monotonically non-increasing trend when the anomaly percentage tends to 50%. This is an expected behavior since both the discriminator and the generator might confuse normal and abnormal instances and generate anomalous instances that seem ordinary. Therefore, when the anomalies increase, HypAD reaches performances similar to the random classifier with an equal probability of outputting a 0/1. For completeness purposes, in Fig. 9, we report the overall performances of HypAD when the anomaly percentage changes from 25% to 50%. To compare the original performances of HypAD, we include the F1 and G scores as reported in Table 2 depicted with a darker color.

To complete our analysis, we also conducted, with the help of geriatricians, an in-depth study of the E-Linus cohort using the visual tools developed within the E-Linus project [69]. For example, Figs. 10(a) and 10(b) show the visual representation of a weekly routine for two of the monitored patients.[21] Although the two routines are similar, as we said, they follow a pre-determined daily program. The second patient is

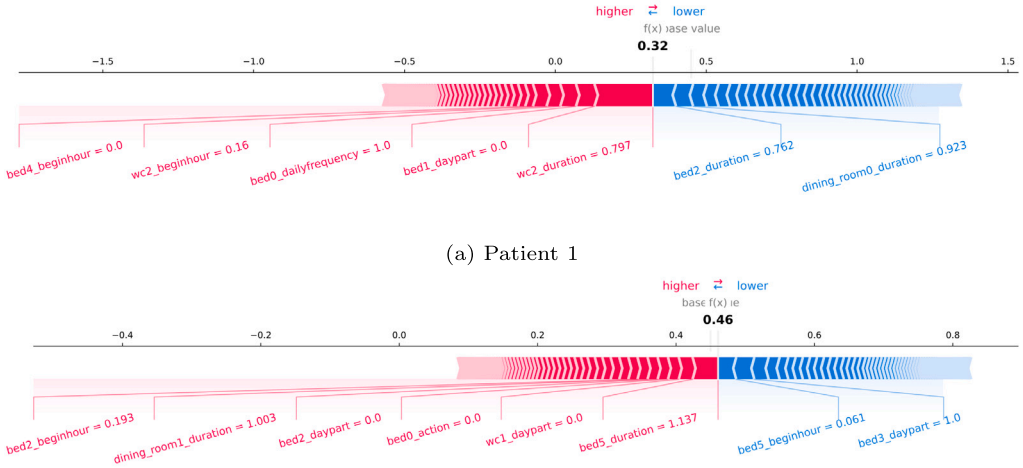
[21] Alternative types of visualization can be selected, e.g., in [70] multiple temporal axes are shown for each activity. However, we prefer collapsing the axes to get a quick glimpse of weekly and monthly routines.



(a) Daily routine of Patient 1

(b) Daily routine of Patient 2

Fig. 10. Visual representation of the weekly behavioral patterns of two particular patients in E-Linus [69]. Each row represents the ADL for a particular day. The colors represent the different activities performed in a day. Note that the daily routine of Patient 2 shows polyuria and an irregular sleeping behavior, however this irregularity represents the standard profile of Patient 2. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



(a) Patient 1

(b) Patient 2

Fig. 11. SHAP visual explanations of two detected sleeping anomalies of Patient 1 and Patient 2. The charts show the contribution of different features to the output of the anomaly detector. Blue depicts the features that play a part in the anomaly; in red, those that counterbalance it to the expected predicted value. Note that, for the same type of anomaly (bed), the explanation is quite different for the two patients. Patient 1 has very regular sleeping habits. Hence, unusual sleep durations are those that should create more alert. Instead, for Patient 2 irregular sleep duration is common. In this case, other features may determine an anomaly, like, as shown here, the day part and beginning time of the sleep (e.g., taking an unusual nap late in the morning). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

visibly less active due to his/her older age and, furthermore, is affected by polyuria and irregular sleeping behavior.

Caregivers can also visualize statistics on each specific activity and monitor parameters. Doctors have also found particularly helpful the visual explanation of anomalies generated with SHAP. For example, Figs. 11(a) and 11(b) visually show the contribution, based on their Shapley value, of the various features to a detected anomaly in a patient's sleep activity. The charts, which refer to the sleeping patterns of the two patients whose daily routine is illustrated in Fig. 10, show that some features contribute to the anomaly (in blue), and others offset it to the base value (in red). In this way, caregivers can quickly diagnose and validate the system prediction of abnormal behavior and are also supported in identifying the main signals that, for each patient, should create an alert. For example, Fig. 11(a) (referring to a sleep anomaly detected for Patient 1 of Fig. 10(a) shows that the total duration of the sleep activity and the duration of the subsequent action (breakfast in the dining room) contributed the most on reconstructing an abnormal sleep event for Patient 1. This patient generally has a regular sleep, so the sleep duration is the most relevant feature that, if altered, should create an alert. Differently from Patient 1, Patient 2 exhibits more irregular sleeping habits Fig. 10(b), therefore, the sleep duration is not expected to contribute to anomalies since it is usually variable. For example, Fig. 11(b) shows that the unusual day part in which the patient has taken a nap on a given day (specifically, mid-morning) has

contributed the most to classifying that particular sleep event as an anomaly. In general, explanations help doctors identify the characteristics that contribute most to determining an unusual behavior for each patient and should therefore be monitored with greater attention.

6. Conclusion

Sensor networks and wearable devices encourage assisted living solutions that help older people be independent and support patient care while preserving their privacy. In this context, implementing algorithmic solutions to detect early signs of possible pathologies based on abnormal signals is particularly challenging. The solution proposed in this paper, named HypAD, has several relevant and, in some cases, unique features. First, anomaly prediction depends on the temporal and spatial context, on the habits of each specific patient, and is grounded on medical knowledge. Second, our proposal is a novel formulation that combines the reconstruction error (how much a specific element diverges from the normality) and the uncertainty of the model about the reconstruction. The uncertainty component considerably increases the explainability of HypAD predictions, as demonstrated by the minimal variance of the system's performance across different monitoring environments, actions, and anomaly types. The experiments show that HypAD performs comparatively better in challenging domains where behaviors are unpredictable. Here, the contribution of the uncertain estimation to the system's predictions helps the most.

The entire workflow – from the representation of input behaviors to the reporting and explanation of anomalous events – can be monitored using visual analytic tools for the caregivers to gain insight, verify and possibly correct the system's predictions, and consolidate their trust in the system. In particular, caregivers have found the visual explanation of anomalies helpful in identifying specific traits that characterize the expected behavior of every single patient and that, if changed, should create an alert.

Finally, the proposed model has, however, several limitations. First, we do not address drift anomalies [79], which are slow but persistent behavioral changes that none of the systems in this paper are equipped to detect. HypAD does not perform continual learning which is essential in adapting its latent knowledge when a drift is signaled. In our ongoing studies, we are considering the problem of drift anomalies and how the hyperbolic model can address the prediction of these anomalies. A second limitation is a lack of more extensive, real datasets that could bolster our results and experiments. In our work, we coped with this limitation by artificially extending and manipulating sensor data captured from actual patients in a limited period. Although artificial anomalies have been injected by doctors (based on realistic assumptions) through a dedicated interface, long-term and real-life patient data would certainly allow a more accurate estimation of the pros and cons of the defined solution. Regarding this, a new pilot experiment is planned in the coming months in the context of a European project.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has been funded by Regione Lazio, Italy, Project E-Linus POR A0376E0005, within the program POR FESR Lazio 2014–2020.

References

[1] Cotterell N, Buffel T, Phillipson C. Preventing social isolation in older people. Maturitas 2018;113.
[2] Michelozzi P, De'Donato F, Scortichini M, De Sario M, Noccioli F, Rossi P, Davoli M. Mortality impacts of the coronavirus disease (COVID-19) outbreak by sex and age: rapid mortality surveillance system, Italy, 1 February to 18 2020. Eurosurveillance 2020;25:2000620.
[3] De Pue S, Gillebert C, Dierckx E, Vanderhasselt M, De Raedt R, Bussche E. The impact of the COVID-19 pandemic on wellbeing and cognitive functioning of older adults. Sci Rep 2021;11:1–11.
[4] Stanton R, To Q, Khalesi S, Williams S, Alley S, Thwaite T, Fenning A, Vandelanotte C. Depression, anxiety and stress during COVID-19: associations with changes in physical activity, sleep, tobacco and alcohol use in Australian adults. Int J Environ Res Public Health 2020;17:4065.
[5] American Psychiatric Association Diagnostic and statistical manual of mental disorders: DSM-5. 2013, Autor.
[6] Zhang H, Zhang Y, Zhong B, Lei Q, Yang L, Du J, Chen D. A comprehensive survey of vision-based human action recognition methods. Sensors 2019;19.
[7] Pirzada P, Wilde A, Doherty G, Harris-Birtill D. Ethics and acceptance of smart homes for older adults. Inform Health Soc Care 2021;1–28.
[8] Ruihua M, Meng Z, Nan C, Panqi L, Hua G, Sijia L, Jing S, Ke Z, Yunlong T, Shuping T, Fude Y, Li T, Zhiren W. Differences in facial expression recognition between unipolar and bipolar depression. Front Psychol 2021;12.
[9] Guo W, Yang H, Liu Z, Xu Y, Hu B. Deep neural networks for depression recognition based on 2D and 3D facial expressions under emotional stimulus tasks. Front Neurosci 2021;15:342.
[10] Geiger A, Liu D, Alnegheimish S, Cuesta-Infante A, Veeramachaneni K. TadGAN: Time series anomaly detection using generative adversarial networks. In: 2020 IEEE int. conf. on big data. 2020, p. 33–43.
[11] Szymanowicz S, Charles J, Cipolla R. Discrete neural representations for explainable anomaly detection. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. 2022, p. 148–56.
[12] Pang G, Ding C, Shen C, Hengel A. Explainable deep few-shot anomaly detection with deviation networks. 2021, CoRR. abs/2108.00462.
[13] Pang G, Shen C, Cao L, Hengel A. Deep learning for anomaly detection: A review. ACM Comput Surv 2021;54:1–38.
[14] Dietterich T. Ensemble methods in machine learning. In: Int. workshop on multiple classifier systems. 2000, p. 1–15.
[15] Gal Y, Ghahramani Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: Int. conf. on machine learning. 2016, p. 1050–9.
[16] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y. Generative adversarial nets. In: Advances in neural information processing systems, Vol. 27. 2014.
[17] Du Y, Mordatch I. Implicit generation and modeling with energy based models. Adv Neural Inf Process Syst 2019;32:3608–18.
[18] Salakhutdinov R, Hinton G. Deep boltzmann machines. In: Artificial intelligence and statistics. 2009, p. 448–55.
[19] Xie J, Lu Y, Gao R, Zhu S, Wu Y. Cooperative training of descriptor and generator networks. IEEE Trans Pattern Anal Mach Intell 2018;42:27–45.
[20] Angiulli F, Pizzuti C. Fast outlier detection in high dimensional spaces. In: European conf. on principles of data mining and knowledge discovery. 2002, p. 15–27.
[21] Ghoting A, Parthasarathy S, Otey M. Fast mining of distance-based outliers in high-dimensional datasets. Data Min Knowl Discov 2008;16:349–64.
[22] Breunig M, Kriegel H, Ng R, Sander J. LOF: identifying density-based local outliers. ACM Sigmod Rec 2000;29:93–104.
[23] Li L, Yan J, Wen Q, Jin Y, Yang X. Learning robust deep state space for unsupervised anomaly detection in contaminated time-series. IEEE Trans Knowl Data Eng 2022.
[24] Hundman K, Constantinou V, Laporte C, Colwell I, Söderström T. Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding. In: KDD. 2018, p. 387–95.
[25] Ahmad S, Lavin A, Purdy S, Agha Z. Unsupervised real-time anomaly detection for streaming data. Neurocomputing 2017;262.
[26] Oh M, Iyengar G. Sequential anomaly detection using inverse reinforcement learning. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. 2019, p. 1480–90.
[27] An J, Cho S. Variational autoencoder based anomaly detection usin reconstruction probability. In: Special lecture on IE, Vol. 2. 2015.
[28] Lu W, Cheng Y, Xiao C, Chang S, Huang S, Liang B, Huang T. Unsupervised sequential outlier detection with deep architectures. IEEE Trans Image Process 2017;26:4321–30.
[29] Marchi E, Vesperini F, Weninger F, Eyben F, Squartini S, Schuller B. Non-linear prediction with LSTM recurrent neural networks for acoustic novelty detection. In: 2015 international joint conference on neural networks. IJCNN, 2015, p. 1–7.
[30] Zhou C, Paffenroth R. Anomaly detection with robust deep autoencoders. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining. 2017, p. 665–74.

[31] Zhou Y, Song X, Zhang Y, Liu F, Zhu C, Liu L. Feature encoding with autoencoders for weakly supervised anomaly detection. IEEE Trans Neural Netw Learn Syst 2021.
[32] Zheng P, Yuan S, Wu X, Li J, Lu A. One-class adversarial nets for fraud detection. In: Proceedings of the AAAI conference on artificial intelligence, Vol. 33. 2019, p. 1286–93.
[33] Zaheer M, Lee J, Astrid M, Lee S. Old is gold: Redefining the adversarially learned one-class classifier training paradigm. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 14183–93.
[34] Ganea O, Becigneul G, Hofmann T. Hyperbolic neural networks. In: Advances in neural information processing systems, Vol. 31. 2018.
[35] Gulcehre C, Denil M, Malinowski M, Razavi A, Pascanu R, Hermann K, Battaglia P, Bapst V, Raposo D, Santoro A, Freitas N. Hyperbolic attention networks. In: International conference on learning representations. 2019.
[36] Lou A, Katsman I, Jiang Q, Belongie S, Lim S, Sa C. Differentiating through the Fréchet mean. In: Proceedings of the 37th international conference on machine learning, ICML 2020, 13-18 July 2020, Virtual Event, Vol. 119. 2020, p. 6393–403.
[37] Shimizu R, Mukuta Y, Harada T. Hyperbolic neural networks++. In: Int. conf. on learning representations. 2021.
[38] Liu Q, Nickel M, Kiela D. Hyperbolic graph neural networks. Adv Neural Inf Process Syst 2019;32.
[39] Chami I, Ying Z, Ré C, Leskovec J. Hyperbolic graph convolutional neural networks. Adv Neural Inf Process Syst 2019;32:4868–79.
[40] Liu S, Chen J, Pan L, Ngo C, Chua T, Jiang Y. Hyperbolic visual embedding learning for zero-shot recognition. In: IEEE/CVF conference on computer vision and pattern recognition. CVPR, 2020.
[41] Khrulkov V, Mirvakhabova L, Ustinova E, Oseledets I, Lempitsky V. Hyperbolic image embeddings. In: IEEE/CVF conference on computer vision and pattern recognition. CVPR, 2020.
[42] Long T, Mettes P, Shen H, Snoek C. Searching for actions on the hyperbole. In: 2020 IEEE/CVF conference on computer vision and pattern recognition. CVPR, 2020, p. 1138–47.
[43] Surís D, Liu R, Vondrick C. Learning the predictability of the future. In: Proceedings of the IEEE conference on computer vision and pattern recognition. CVPR, 2021.
[44] Akbulut F, Ikitimur B, Akan A. Wearable sensor-based evaluation of psychosocial stress in patients with metabolic syndrome. Artif Intell Med 2020;104:101824.
[45] Arifoglu D, Bouchachia A. Detection of abnormal behaviour for dementia sufferers using convolutional neural networks. Artif Intell Med 2019;94:88–95.
[46] Bilbao A, Almeida A, Ipiña D. Promotion of active ageing combining sensor and social network data. J Biomed Inform 2016;64:108–15.
[47] Zhang Y, Chen Y, Wang J, Pan Z. Unsupervised deep anomaly detection for multi-sensor time-series signals. IEEE Trans Knowl Data Eng 2021;1.
[48] Cook A, Mısırlı G, Fan Z. Anomaly detection for IoT time-series data: A survey. IEEE IoT J 2019;7:6481–94.
[49] Li X, Pang T, Liu W, Wang T. Fall detection for elderly person care using convolutional neural networks. In: 2017 10th int. congress on image and signal processing, biomedical engineering and informatics (CISP-BMEI). 2017, p. 1–6.
[50] Deep S, Zheng X, Karmakar C, Yu D, Hamey L, Jin J. A survey on anomalous behavior detection for elderly care using dense-sensing networks. IEEE Comm Surv Tutor 2020;22:352–70.
[51] Wang J, Chen Y, Hao S, Peng X, Hu L. Deep learning for sensor-based activity recognition: A survey. Pattern Recognit Lett 2017;119.
[52] Can Y, Arnrich B, Ersoy C. Stress detection in daily life scenarios using smart phones and wearable sensors: A survey. J Biomed Inform 2019;92:103139.
[53] Qi J, Yang P, Waraich A, Deng Z, Zhao Y, Yang Y. Examining sensor-based physical activity recognition and monitoring for healthcare using Internet of Things: A systematic review. J Biomed Inform 2018;87:138–53.
[54] Dahmen J, Cook D. Indirectly supervised anomaly detection of clinically meaningful health events from smart home data. ACM Trans Intell Syst Technol (TIST) 2021;12:1–18.
[55] Bloem B, Hausdorff J, Visser J, Giladi N. Falls and freezing of gait in Parkinson's disease: a review of two interconnected, episodic phenomena. Mov Disord: Off J Mov Disord Soc 2004;19:871–84.
[56] Bakar U, Ghayvat H, Hasanm S, Mukhopadhyay S. Activity and anomaly detection in smart home: A survey. 2016.
[57] Mshali H, Lemlouma T, Magoni D. Context-aware adaptive framework for e-health monitoring. In: 2015 IEEE int. conf. on data science and data intensive systems. 2015, p. 276–83.
[58] Brockwell P, Davis R, Calder M. Introduction to time series and forecasting. Springer; 2002.
[59] Ranjan K, Tripathy D, Prusty B, Jena D. An improved sliding window prediction-based outlier detection and correction for volatile time-series. Int J Numer Model: Electron Netw Devices Fields 2021;34:e2816.
[60] Chu C. Time series segmentation: A sliding window approach. Inform Sci 1995;85:147–73.
[61] Frank R, Davey N, Hunt S. Time series prediction and neural networks. J Intell Robot Syst 2001;31:91–103.
[62] Laguna J, Olaya A, Borrajo D. A dynamic sliding window approach for activity recognition. In: International conference on user modeling, adaptation, and personalization. 2011, p. 219–30.
[63] Yu Y, Zhu Y, Li S, Wan D. Time series outlier detection based on sliding window prediction. Math Probl Eng 2014;2014.
[64] Serrano S, Smith N. Is attention interpretable? In: Proc. of the 57th conf. of the association for computational linguistics, 2019, Volume 1: Long Papers. 2019, p. 2931–51.
[65] Lundberg S, Lee S. A unified approach to interpreting model predictions. In: Proc. of the 31st international conf. on neural information processing systems. 2017, p. 4768–77.
[66] Antwarg L, Miller R, Shapira B, Rokach L. Explaining anomalies detected by autoencoders using Shapley additive explanations. Expert Syst Appl 2021;186:115736.
[67] Cook D, Crandall A, Thomas B, Krishnan N. CASAS: A smart home in a box. Computer 2012;46:62–9.
[68] Mshali H, Lemlouma T, Magoni D. Context-aware adaptive framework for e-health monitoring. In: 2015 IEEE international conference on data science and data intensive systems. 2015, p. 276–83.
[69] Podo L, Velardi P. AnomalyByClick: An interactive visualization tool for monitoring activities of daily living and anomaly annotation. In: Proceedings of the 2022 international conference on advanced visual interfaces. 2022.
[70] Juarez J, Ochotorena J, Campos M, Combi C. Spatiotemporal data visualisation for homecare monitoring of elderly people. Artif Intell Med 2015;65:97–111.
[71] Zhang C, Song D, Chen Y, Feng X, Lumezanu C, Cheng W, Ni J, Zong B, Chen H, Chawla N. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In: Proc. of the AAAI conf. on AI, Vol. 33. 2019, p. 1409–16.
[72] American Psychiatric Association A, et al. Diagnostic and statistical manual of mental disorders. Washington, DC: American Psychiatric Association; 1980.
[73] Baldi P. Autoencoders, unsupervised learning, and deep architectures. In: Proc. of ICML workshop on unsupervised and transfer learning. 2012, p. 37–49.
[74] Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 2014;15:1929–58.
[75] Maggipinto M, Masiero C, Beghi A, Susto G. A convolutional autoencoder approach for feature extraction in virtual metrology. Procedia Manuf 2018;17:126–33.
[76] Liu F, Ting K, Zhou Z. Isolation forest. In: 2008 eighth IEEE international conference on data mining. 2008, p. 413–22.
[77] Li C, Guo L, Gao H, Li Y. Similarity-measured isolation forest: Anomaly detection method for machine monitoring data. IEEE Trans Instrum Meas 2021;70:1–12.
[78] Sagheer A, Kotb M. Unsupervised pre-training of a deep LSTM-based stacked autoencoder for multivariate time series forecasting problems. Sci Rep 2019;9:1–16.
[79] Gemaque R, Costa A, Giusti R, Dos Santos E. An overview of unsupervised drift detection methods. Wiley Interdiscip Rev: Data Min Knowl Discov 2020;10:e1381.