Expert Systems With Applications 87 (2017) 280–290

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Detecting unseen falls from wearable devices using channel-wise ensemble of autoencoders

Shehroz S. Khan[a,b,*], Babak Taati[a]

[a] Toronto Rehabilitation Institute, University Health Network, Toronto, Canada
[b] University of Toronto, Canada

ARTICLE INFO

Article history:
Received 30 March 2017
Revised 2 June 2017
Accepted 7 June 2017
Available online 15 June 2017

Keywords:
Fall detection
One-class classification
Autoencoder
Anomaly detection

ABSTRACT

A fall is an abnormal activity that occurs rarely, so it is hard to collect real data for falls. It is, therefore, difficult to use supervised learning methods to automatically detect falls. Another challenge in automatically detecting falls is the choice of engineered features. In this paper, we formulate fall detection as an anomaly detection problem and propose to use an ensemble of autoencoders to learn features from different channels of wearable sensor data trained only on normal activities. We show that the traditional approach of choosing a threshold as the maximum of the reconstruction error on the training normal data is not the right way to identify unseen falls. We propose two methods for automatic tightening of reconstruction error from only the normal activities for better identification of unseen falls. We present our results on two activity recognition datasets and show the efficacy of our proposed method against traditional autoencoder models and two standard one-class classification methods.

1. Introduction

Falls are a major cause of both fatal and non-fatal injury and a hindrance in living independently. Each year an estimated 424, 000 individuals die from falls globally and 37.3 million falls require medical attention (World Health Organization, 2016). Experiencing a fall may lead to a fear of falling (Igual, Medrano, & Plaza, 2013), which in turn can result in lack of mobility, less productivity and reduced quality of life. There exist several commercial wearable devices to detect falls (Pannurat, Thiemjarus, & Nantajeewarawat, 2014); most of them use accelerometers to capture motion information. They normally come with an alarm button to manually contact a caregiver if the fall is not detected by the device. However, most of the devices for detecting falls produce many false alarms (El-Bendary, Tan, Pivot, & Lam, 2013). Automatic detection of falls is long sought; hence, machine learning techniques are needed to automatically detect falls based on sensor data. However, a fall is a rare event that does not happen frequently (Stone & Skubic, 2015); therefore, during the training phase, there may be very few or no fall samples. Standard supervised classification techniques may not be suitable in this type of skewed data scenario (Khan, Karg, Kulić, & Hoey, 2014). Another issue regarding the use of machine learning methods in fall detection is the choice of features. Traditional activity recognition and fall detection methods extract a variety of domain specific features from raw sensor readings to build classification models (Khan, 2016; Ravi, Dandekar, Mysore, & Littman, 2005). It is very difficult to ascertain the number or types of engineered features, specially in the absence of fall specific training data to build generalizable models.

To handle the problems of lack of training data from real falls and the difficulty in engineering appropriate features, we explore the use of Autoencoders (AE) (Japkowicz, Myers, & Gluck, 1995) that are trained only on normal activities. AEs can learn generic features from the raw sensor readings and can be used to identify unseen falls as abnormal activities during testing based on a threshold on the reconstruction error. We present two ensembles approaches of AE that train on the raw data of the normal activities from different channels of accelerometer and gyroscope separately and the results of each AE is combined to arrive upon a final decision. Typically, while using AE, the maximum of reconstruction error on the training set is considered as the threshold to identify an activity as abnormal (Dau, Ciesielski, & Song, 2014). However, we experimentally show that such threshold may not be appropriate for detecting falls due to noisy sensor data. We present two threshold tightening techniques to remove few outliers from the normal data. Then, either a new threshold is derived using inter-quartile range or by training a new AE on the training data with outliers removed. We show result on two activity recognition datasets that

* Corresponding author at: Toronto Rehabilitation Institute, University Health Network, Toronto, Canada.

E-mail addresses: shehroz.khan@utoronto.ca (S.S. Khan), babak.taati@uhn.ca (B. Taati).

http://dx.doi.org/10.1016/j.eswa.2017.06.011
0957-4174/

contain different normal activities along with falls from wearable sensors.

The rest of the paper is organized as follows. In the next section, we present a brief introduction to AEs. Section 3 reviews the literature on detecting falls as anomaly, by using AE and on the use of AE in general outlier detection tasks. We present the proposed channel-wise ensemble of AE and two threshold tightening approaches using reconstruction error in Section 5. Experimental analysis and results are discussed in Section 6, followed by conclusions and future work in Section 7.

2. Brief introduction to autoencoders

An AE is an unsupervised multi-layer neural network that learns compact representation of the input data (Scholz & Vigário, 2002). An AE tries to learn an identity function such that its outputs are similar to its inputs. However, by putting constraint on the network, such as limiting the number of hidden neurons, it can discover compact representations of the data that can be used as features for other supervised or unsupervised learning tasks. An AE is often trained by using the backpropagation algorithm and consists of an encoder and decoder part. If there is one hidden layer, an AE takes the input $\mathbf{x} \in \mathbb{R}^d$ and maps it onto $\mathbf{h} \in \mathbb{R}^p$, s.t.

$$\mathbf{h} = f(W\mathbf{x} + b) \tag{1}$$

where **W** is a weight matrix and **b** is a bias term and $f(.)$ is a mapping function. This step is referred to as encoding or learning latent representation, after which **h** is mapped back to reconstruct **y** of the same shape as **x**, i.e.

$$\mathbf{y} = g(\mathbf{W}'\mathbf{h} + \mathbf{b}') \tag{2}$$

This step is referred to as decoding or reconstructing the input back from latent representation. An AE can be used to minimize the squared reconstruction error, L i.e.,

$$L(\mathbf{x},\mathbf{y}) = \| \mathbf{x}\text{-}\mathbf{y} \|^2 \tag{3}$$

AE can learn compact and useful features if $p < d$; however, it can still discover interesting structures if $p > d$. This can be achieved by imposing a sparsity constraint on the hidden units, s.t. neurons are inactive most of the time or the average activation of each hidden neuron is close to zero. To achieve sparsity, an additional sparsity parameter is added to the objective function. Multiple layers of AEs can be stacked on top of each other to learn hierarchical features from the raw data. They are called Stacked AE (SAE). During encoding of a SAE, the output of first hidden layer serves as the input to the second layer, which will learn second level hierarchical features and so on. For decoding a SAE, the output of the last hidden layer is reconstructed at the second last hidden layer, and so on until the original input is reconstructed.

3. Related work

AEs can be used both in supervised and unsupervised settings for identifying falls. In a supervised classification setting, AE is used to learn representative features from both the normal and fall activities. This step can be followed by a standard machine learning classifier trained on these compressed features (Li, Shi, Ding, & Liu, 2014b) or by a deep network (Jokanovic, Amin, & Ahmad, 2016). In the unsupervised mode or One-Class Classification (OCC) (Khan & Madden, 2014) setting, only data for normal activities is present during training the AE. In these situations, an AE is used to learn representative features from the raw sensor data of normal activities. This step is followed by either employing (i) a discriminative model by using one-class classifiers or (ii) a generative model with appropriate threshold based on reconstruction error, to detect falls and normal activities. The present paper follows the unsupervised AE approach with a generative model and aimed at finding an appropriate threshold to identify unseen falls.

Machine learning techniques are extensively used for handling fall detection problem (Özdemir & Barshan, 2014); it has also been formulated as an anomaly detection problem due to the fact that falls occur rarely (Khan & Hoey, 2017). Popescu and Mahnot (2009) present a fall detection technique that uses acoustic signals of normal activities for training and detecting fall sounds from it. They train One-class SVM (OSVM), one-class nearest neighbour approach (OCNN) and One-class Gaussian Mixture Model to train models on normal acoustic signals and find that OSVM performs the best in detecting falls. However, it is outperformed by its supervised counterpart. Khan, Yu, Feng, Wang, and Chambers (2015) propose an unsupervised acoustic fall detection system with interference suppression that makes use of the features extracted from the normal sound samples, and constructs an OSVM model to distinguish falls from non-falls. They show that in comparison to Popescu and Mahnot (2009), their interference suppression technique makes the fall detection system less sensitive to interferences by using only two microphones. Principi, Droghini, Squartini, Olivetti, and Piazza (2016) present a floor acoustic sensor that can automatically discriminates the sounds produced by falls of distinct objects. They show that this sensor can capture fall signals with high signal-to-noise ratio with respect to an aerial microphone by filtering out high frequency components. Tran, Le et al. (2014) propose to use image and audio to tackle the problem of abnormal events detection, such as, falling, lying motionless, etc. They introduce audio and video based event detection systems that resulted in high sensitivity and low false alarm rate in two setup environments. Medrano, Igual, Plaza, and Castro (2014) propose to identify falls using a smartphone as a novelty from the normal activities and find that OCNN performs better than OSVM but is outperformed by supervised SVM. Micucci, Mobilio, Napoletano, and Tisato (2015) evaluate several OCC methods for fall detection using data from smartphone and show that in most of the cases, OCNN performs better or similar to the supervised SVM and KNN. Zhou, Wang, Chen, Chen, and Zhao (2012) present a method to detect falls using transitions between the activities as a cue to model falls. They train supervised classification methods using normal activities collected from a mobile device, then extract transitions among these activities, and use them to train an OSVM. They show that this method performs better than an OSVM trained with only normal activities. Khan et al. (2014) present 'X-Factor' HMM approaches that are similar to normal HMMs, but have inflated output covariances that can be used as alternative models to estimate the parameters of unseen falls. Their results show high detection rates for falls on two activity recognition datasets.

A lot of work has been done in evaluating the feasibility of learning generic representations through AEs for general activity recognition and fall detection tasks. Plötz, Hammerla, and Olivier (2011) explore the potential of discovering universal features for context-aware application using wearable sensors. They present several feature learning approaches using PCA and AE and show their superior performance in comparison to standard features across a range of activity recognition applications. Budiman, Fanany, and Basaruddin (2014) use SAEs and marginalized SAE to infer generic features in conjunction with neural networks and Extreme Learning Machines as the supervised classifiers to perform pose-based action recognition. Li, Shi, Ding, and Liu (2014a) compare SAE, Denoising AE and PCA for unsupervised feature learning in activity recognition using smartphone sensors. They show that traditional features perform worse than the generic features inferred through AEs. Jokanovic et al. (2016) use a SAE to learn generic lower dimensional features and use softmax regression classifier to identify falls using radar signals. Other re-