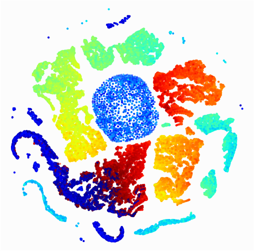
Unsupervised Anomaly Detection in Time Series Data using Deep Learning

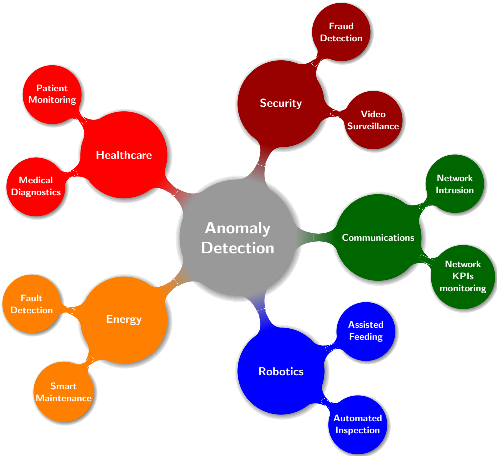
Integrated Master in Electrical and Computer Engineering



João Pereira

23rd November, 2018

Anomaly detection is about finding patterns in data that do not conform to *expected* or *normal* behaviour.



Main Challenges

- Most data in the world are **unlabelled**

anomaly labels

$$\text{Dataset } \mathcal{D} = \left\{ \left(\mathbf{x}^{(i)}, \mathbf{y}^{*(i)} \right) \right\}_{i=1}^{N}$$

- Annotating large datasets is difficult, time-consuming and expensive

- Time series have temporal structure/dependencies

$$\mathbf{x} = \left(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T \right) \quad , \qquad \mathbf{x}_t \in \mathbb{R}^{d_\mathbf{x}}$$

Objectives

We would like:

- Unsupervised: no need for anomaly labels;
- Suitable for sequential data (e.g., time series);
- General;
- Scalable & efficient, allowing real-time detection.

Objectives

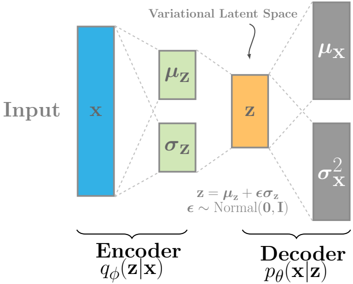
We would like:

- Unsupervised: no need for anomaly labels;
- Suitable for sequential data (e.g., time series);
- General;
- Scalable & efficient, allowing real-time detection.

How to design such a model?

The Principle in a Nutshell

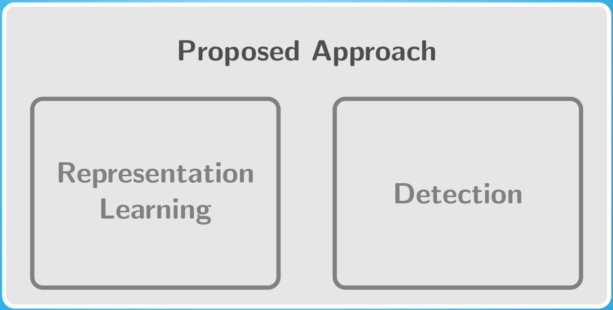
- Train a **Variational Autoencoder**[1,2] to reconstruct input data with mostly normal patterns;

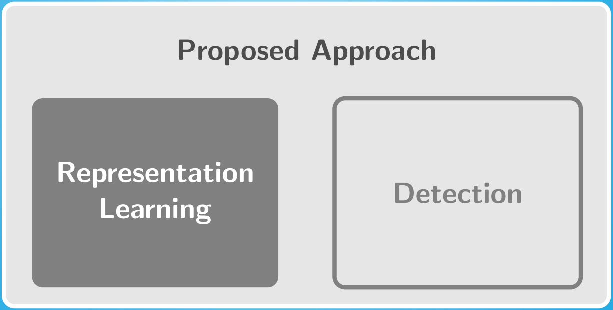


- At test time, it reconstructs well *normal* data, while it fails to reconstruct *anomalous* data;
- The quality of the reconstructions and the representations are used to compute anomaly scores.

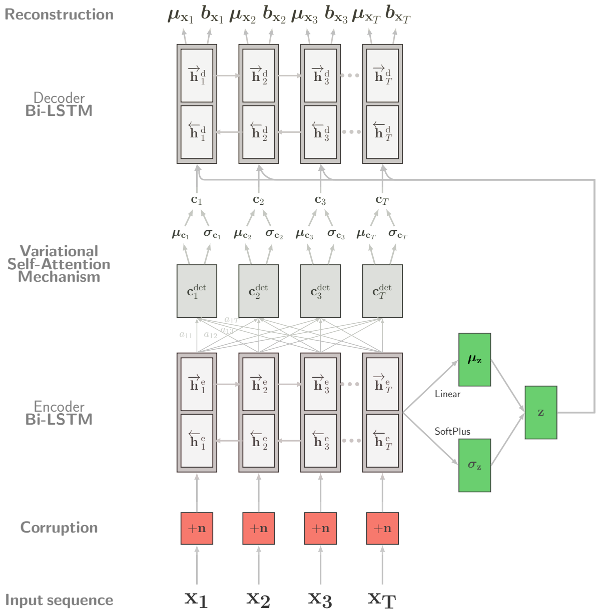
[1] Kingma & Welling, **Auto-Encoding Variational Bayes**, ICLR'14

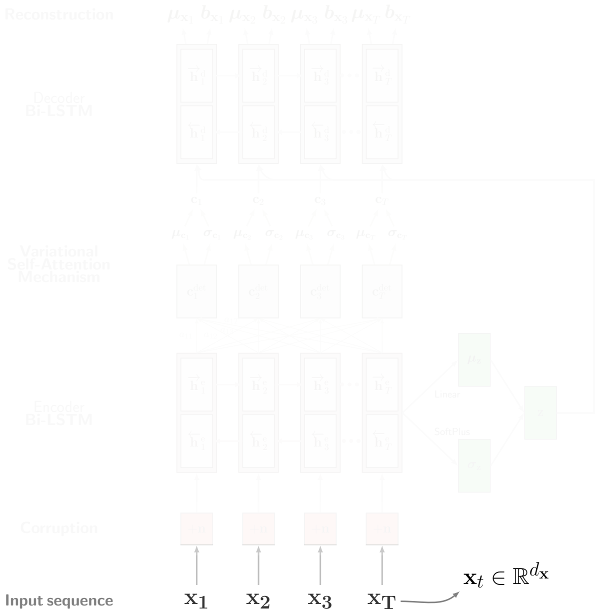
[2] Rezende *et al.*, **Stochastic Backpropagation and Approximate Inference in Deep Generative Models**, ICML'14


Proposed Approach
Representation Learning
Detection


Proposed Approach
Representation Learning
Detection

Representation Learning




Reconstruction
Decoder
Bi-LSTM
Variational
Self-Attention
Mechanism
Encoder
Bi-LSTM
Linear
SoftPlus
Corruption
Input sequence
$\mathbf{x}_1$
$\mathbf{x}_2$
$\mathbf{x}_3$
$\mathbf{x}_T$
$\mathbf{x}_t \in \mathbb{R}^{d_\mathbf{x}}$

Denoising Autoencoding Criterion

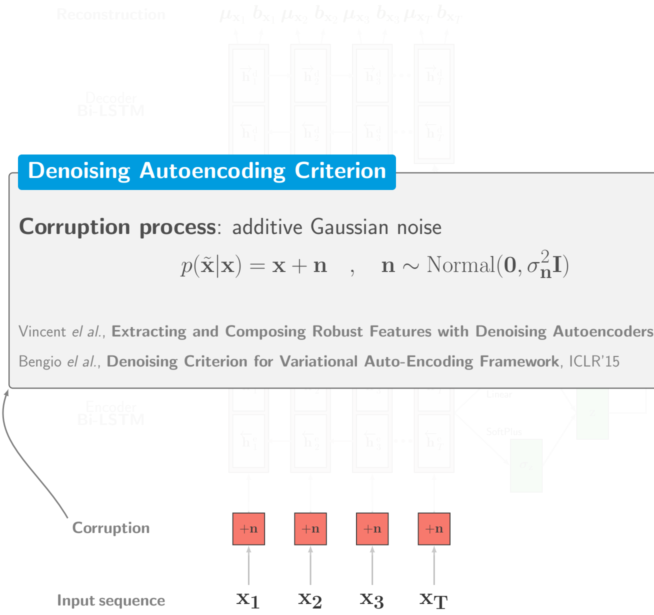
Corruption process: additive Gaussian noise

$$p(\tilde{\mathbf{x}}|\mathbf{x}) = \mathbf{x} + \mathbf{n} \quad , \quad \mathbf{n} \sim \mathrm{Normal}(\mathbf{0}, \sigma_{\mathbf{n}}^2 \mathbf{I})$$

♣

Vincent *el al.*, **Extracting and Composing Robust Features with Denoising Autoencoders**, ICML'08

Bengio *el al.*, **Denoising Criterion for Variational Auto-Encoding Framework**, ICLR'15



Learning temporal dependencies

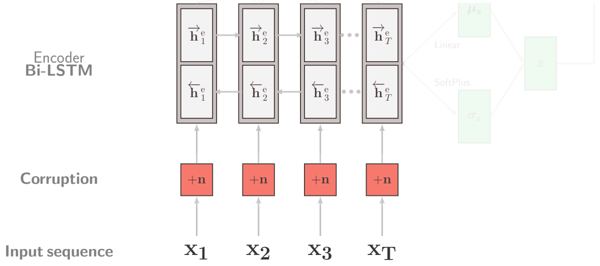
Bidirectional Long-Short Term Memory network

$$\mathbf{h}_t = \left[\overrightarrow{\mathbf{h}}_t; \overleftarrow{\mathbf{h}}_t\right]$$

- 256 units, 128 in each direction
- Sparse regularization, $\Omega(\mathbf{z}) = \lambda \sum_{i=1}^{d_{\mathbf{z}}} |z_i|$

Hochreiter *el al.*, **Long-Short Term Memory**, Neural Computation'97

Graves *el al.*, **Bidirectional LSTM Networks for Improved Phoneme Classification and Recognition**, ICANN'05



Variational Latent Space

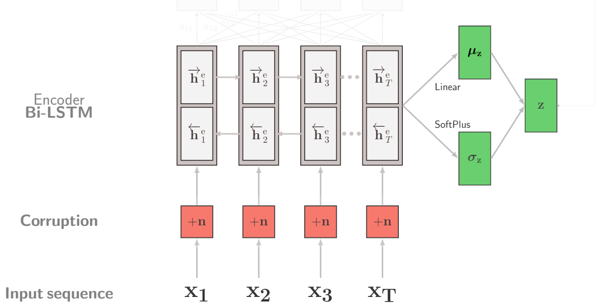
Variational parameters derived using neural networks

$$(\boldsymbol{\mu}_{\mathbf{z}}, \boldsymbol{\sigma}_{\mathbf{z}}) = \mathrm{Encoder}(\mathbf{x})$$

Sample from the approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$

$$\mathbf{z} = \boldsymbol{\mu}_{\mathbf{z}} + \boldsymbol{\sigma}_{\mathbf{z}} \odot \boldsymbol{\epsilon} \qquad \boldsymbol{\epsilon} \sim \mathrm{Normal}(\mathbf{0}, \mathbf{I})$$

Kingma & Welling, **Auto-Encoding Variational Bayes**, ICLR, 2014

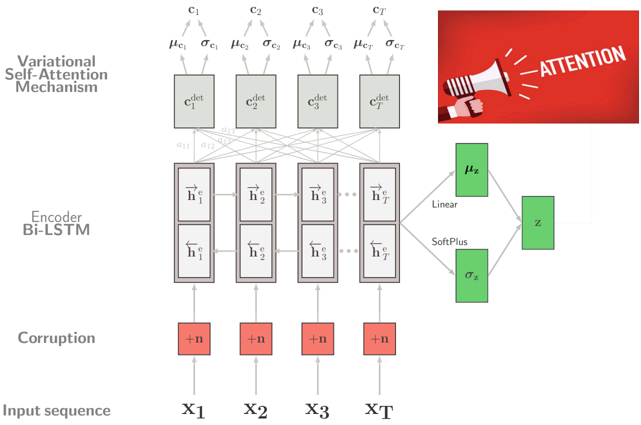


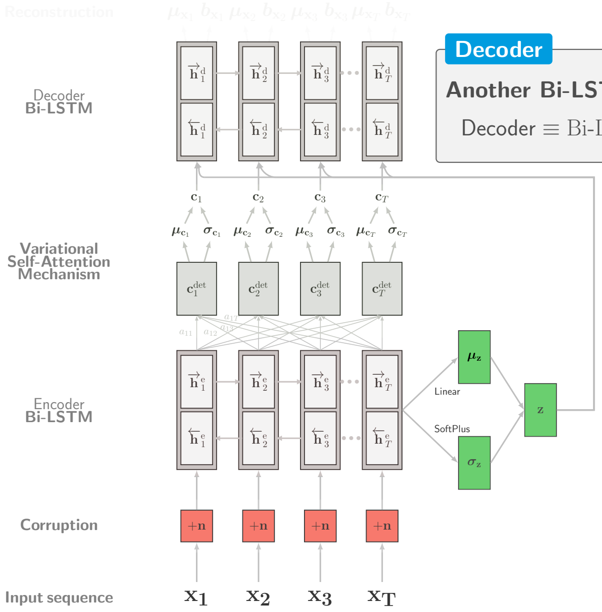
Introducing Variational Self-Attention

A combination of self-attention and variational inference.

$$\mathbf{c}_t^{\text{det}} = \sum_{j=1}^{T} a_{tj}\mathbf{h}_j \qquad (\boldsymbol{\mu}_{\mathbf{c}_t}, \boldsymbol{\sigma}_{\mathbf{c}_t}) = \text{NN}(\mathbf{c}_t^{\text{det}}), \quad \mathbf{c}_t \sim \text{Normal}(\boldsymbol{\mu}_{\mathbf{c}_t}, \boldsymbol{\sigma}_{\mathbf{c}_t}^2\mathbf{I})$$

Vaswani *et al.*, **Attention is All You Need**, NIPS'17



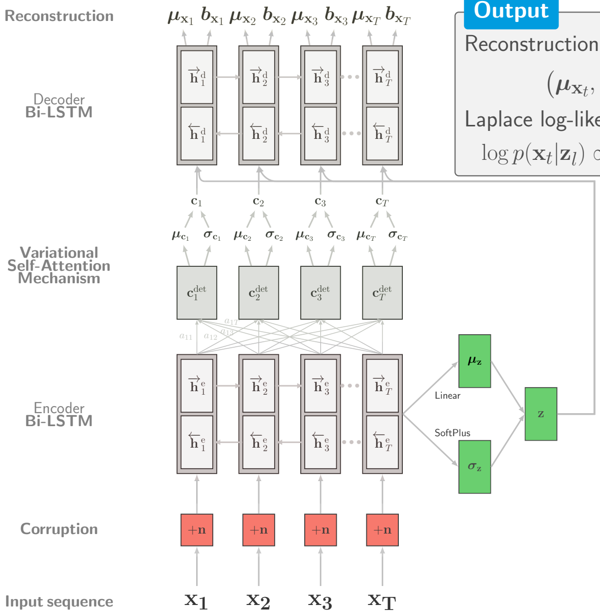


Decoder

Another Bi-LSTM.

$$\text{Decoder} \equiv \text{Bi-LSTM}([\mathbf{z}; \mathbf{c}])$$

Representation Learning



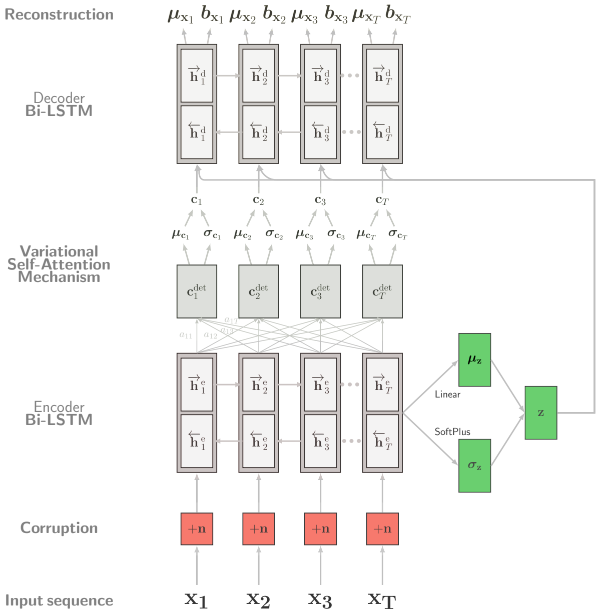
Output

Reconstruction parameters:

$$\left(\boldsymbol{\mu}_{\mathbf{x}_t}, \boldsymbol{b}_{\mathbf{x}_t}\right)_{t=1}^{T}$$

Laplace log-likelihood:

$$\log p(\mathbf{x}_t|\mathbf{z}_l) \propto \|\mathbf{x}_t - \boldsymbol{\mu}_{\mathbf{x}_t}\|_1$$


Reconstruction
μ_{x_1} b_{x_1} μ_{x_2} b_{x_2} μ_{x_3} b_{x_3} μ_{x_T} b_{x_T}
Decoder
Bi-LSTM
$\overrightarrow{h}^d_1$ $\overrightarrow{h}^d_2$ $\overrightarrow{h}^d_3$ $\overrightarrow{h}^d_T$
$\overleftarrow{h}^d_1$ $\overleftarrow{h}^d_2$ $\overleftarrow{h}^d_3$ $\overleftarrow{h}^d_T$
c_1 c_2 c_3 c_T
μ_{c_1} σ_{c_1} μ_{c_2} σ_{c_2} μ_{c_3} σ_{c_3} μ_{c_T} σ_{c_T}
Variational
Self-Attention
Mechanism
c^{det}_1 c^{det}_2 c^{det}_3 c^{det}_T
a_{11} a_{12} a_{13} a_{17}
Encoder
Bi-LSTM
$\overrightarrow{h}^e_1$ $\overrightarrow{h}^e_2$ $\overrightarrow{h}^e_3$ $\overrightarrow{h}^e_T$
$\overleftarrow{h}^e_1$ $\overleftarrow{h}^e_2$ $\overleftarrow{h}^e_3$ $\overleftarrow{h}^e_T$
Linear
SoftPlus
μ_z
σ_z
z
Corruption
+n
Input sequence
x_1 x_2 x_3 x_T

Loss Function

Loss Function

$$\mathcal{L}(\theta, \phi; \mathbf{x}^{(n)}) = -\mathbb{E}_{\mathbf{z} \sim \tilde{q}_\phi(\mathbf{z}|\mathbf{x}^{(n)}),\ \mathbf{c}_t \sim \tilde{q}^a_\phi(\mathbf{c}_t|\mathbf{x}^{(n)})} \Big[\log p_\theta\big(\mathbf{x}^{(n)}|\mathbf{z}, \mathbf{c}\big) \Big]$$

$$+ \lambda_{\mathrm{KL}} \left[\mathcal{D}_{\mathrm{KL}} \Big(\tilde{q}_\phi(\mathbf{z}|\mathbf{x}^{(n)}) \| p_\theta(\mathbf{z}) \Big) + \eta \sum_{t=1}^{T} \mathcal{D}_{\mathrm{KL}} \Big(\tilde{q}^a_\phi(\mathbf{c}_t|\mathbf{x}^{(n)}) \| p_\theta(\mathbf{c}_t) \Big) \right]$$

$\mathcal{D}_{\mathrm{KL}}$ denotes the Kullback-Leibler Divergence

Loss Function

$$\mathcal{L}(\theta, \phi; \mathbf{x}^{(n)}) = \overbrace{-\mathbb{E}_{\mathbf{z}\sim\tilde{q}_\phi(\mathbf{z}|\mathbf{x}^{(n)}),\ \mathbf{c}_t\sim\tilde{q}^a_\phi(\mathbf{c}_t|\mathbf{x}^{(n)})}\left[\log p_\theta\left(\mathbf{x}^{(n)}|\mathbf{z}, \mathbf{c}\right)\right]}^{\text{Reconstruction Term}}$$

$$+ \lambda_{\mathrm{KL}}\left[\underbrace{\mathcal{D}_{\mathrm{KL}}\left(\tilde{q}_\phi(\mathbf{z}|\mathbf{x}^{(n)})\|p_\theta(\mathbf{z})\right)}_{\text{Latent Space KL loss}} + \eta \underbrace{\sum_{t=1}^{T} \mathcal{D}_{\mathrm{KL}}\left(\tilde{q}^a_\phi(\mathbf{c}_t|\mathbf{x}^{(n)})\|p_\theta(\mathbf{c}_t)\right)}_{\text{Attention KL loss}}\right]$$

$\mathcal{D}_{\mathrm{KL}}$ denotes the Kullback-Leibler Divergence

Optimization & Regularization

- About 270k parameters to optimize
- *AMS-Grad* optimizer[3]
- Xavier weight initialization[4]
- Denoising autoencoding criterion[5]
- Sparse regularization in the encoder Bi-LSTM[6]
- KL cost annealing[7]
- Gradient clipping[8]

Training executed on a single GPU (NVIDIA GTX 1080 TI)

---

[3] Reddi, Kale & Kumar, **On the Convergence of Adam and Beyond**, ICLR'18

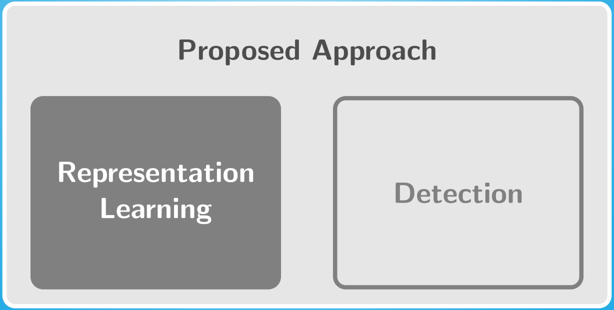
[4] Bengio *et al.*, **Understanding the Difficulty of Training Deep Feedforward Neural Networks**, AISTATS'10

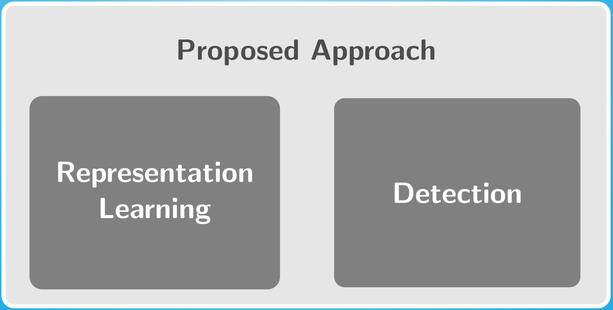
[5] Bengio *et al.*, **Denoising Criterion for Variational Auto-Encoding Framework**, AAAI'17

[6] Arpit *et al.*, **Why Regularized Auto-Encoders Learn Sparse Representation?**, ICML'16

[7] Bowman, Vinyals *et al.*, **Generating Sentences from a Continuous Space**, SIGNLL'16

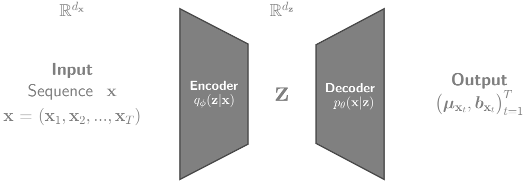
[8] Bengio *et al.*, **On the Difficulty of Training Recurrent Neural Networks**, ICML'13


Proposed Approach
Representation Learning
Detection


Proposed Approach
Representation Learning
Detection

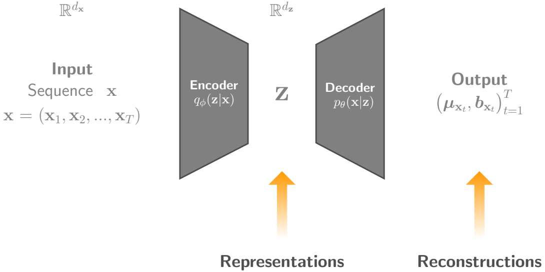
One model, two detection strategies

The model provides two products: **representations** in the $\mathbf{z}$-space and the **reconstruction parameters** in the $\mathbf{x}$-space



One model, two detection strategies

The model provides two products: **representations** in the $\mathbf{z}$-space and the **reconstruction parameters** in the $\mathbf{x}$-space

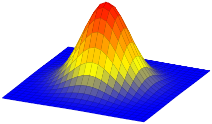


- Reconstruction-based detection
- Latent space-based detection

$$\text{Reconstruction Error} = \mathbb{E}_{\mathbf{z}_l \sim q_\phi(\mathbf{z}|\mathbf{x})}\left[\left\|\mathbf{x} - \mathbb{E}[p_\theta(\mathbf{x}|\mathbf{z}_l)]\right\|_1\right]$$

$$\text{Reconstruction Probability} = \mathbb{E}_{\mathbf{z}_l \sim q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p(\mathbf{x}|\mathbf{z}_l)\right]$$

$q_\phi(\mathbf{z}|\mathbf{x})$

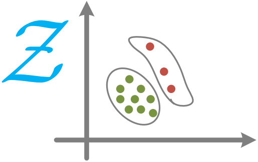


$\mathbf{z}_l \sim \text{Normal}(\boldsymbol{\mu}_\mathbf{z}, \boldsymbol{\sigma}^2_\mathbf{z}\mathbf{I})$

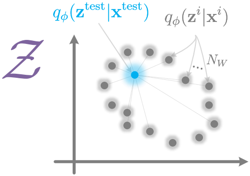
Latent Space Detection

Based on the representations in the z-space.

- **Clustering**



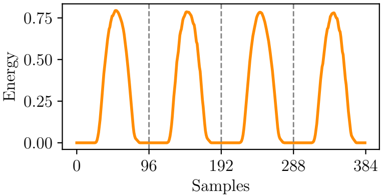
- **Wasserstein Metric (W)**



$$\text{score}(\mathbf{z}^{\text{test}}) = \text{median}\{W(\mathbf{z}^{\text{test}}, \mathbf{z}^i)^2\}_{i=1}^{N_W}$$

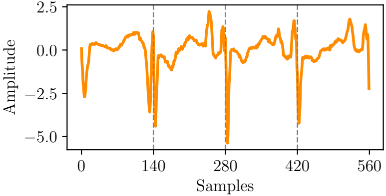
Experiments & Results

Solar PV Generation

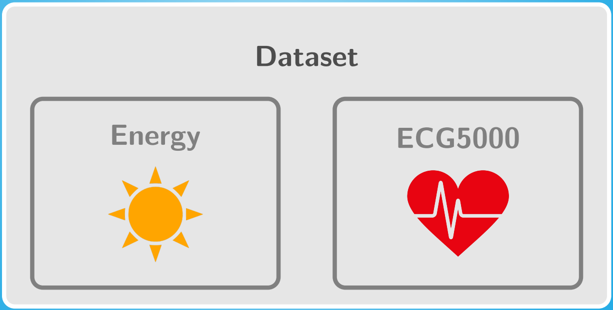


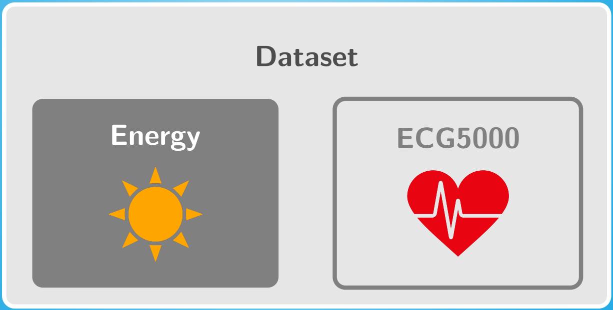
- Provided by c|side
- Recorded every 15min (96 samples/day)
- Daily seasonality
- Unlabelled

Electrocardiogram



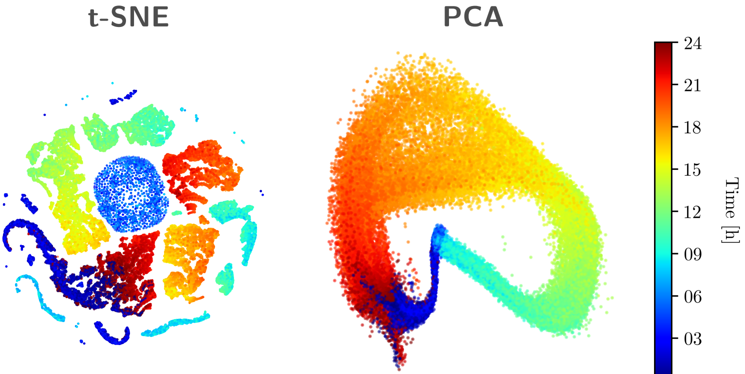
- Available in the UCR Time Series Classification Archive ECG5000 [Keogh *et al.*, 2015]
- One heartbeat ≈ 140 samples
- 5000 sequences
- Labelled, 5 classes annotated


Dataset
Energy
ECG5000

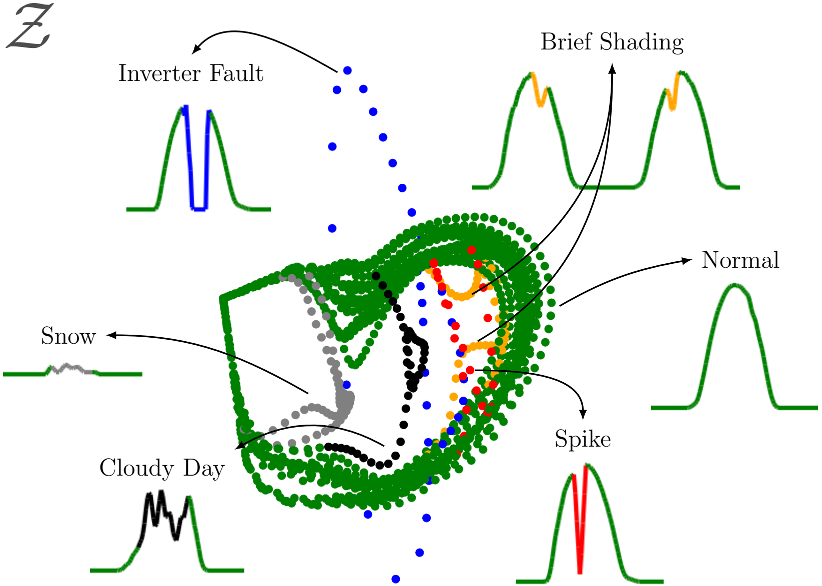

Dataset
Energy
ECG5000

$\mathbf{z}$-space in 2D ($\mathcal{X}_{\text{train}}^{\text{normal}}$)

$T = 12$ (< 96)
$d_{\mathbf{z}} = 3$
online mode



Finding Anomalies in the Latent Space

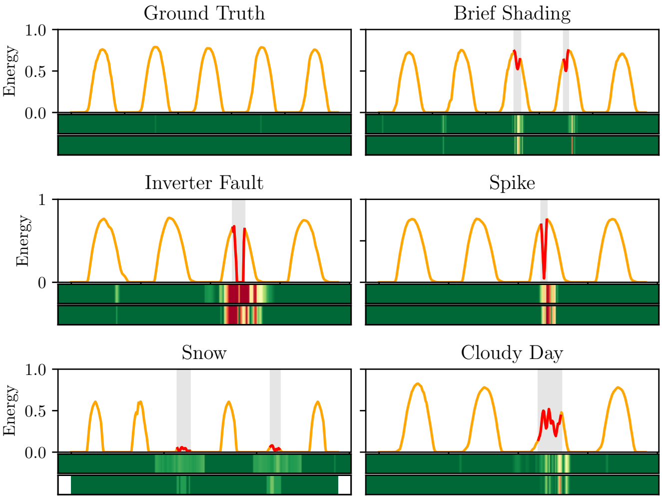


Reconstruction Error

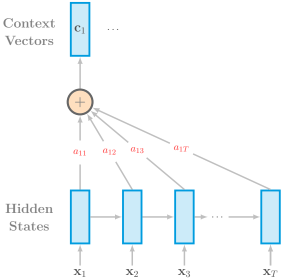
(top bar)

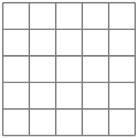
Reconstruction Probability

(bottom bar)

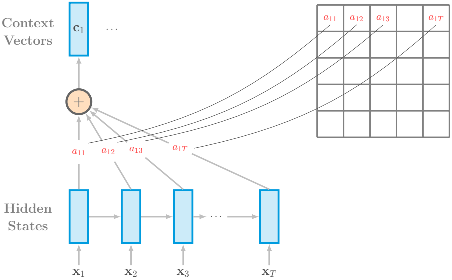


Attention Map

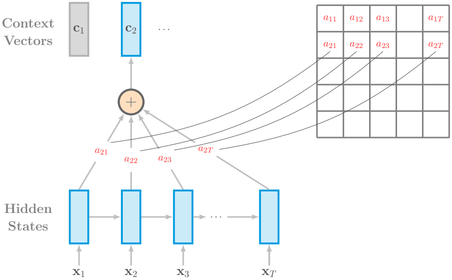




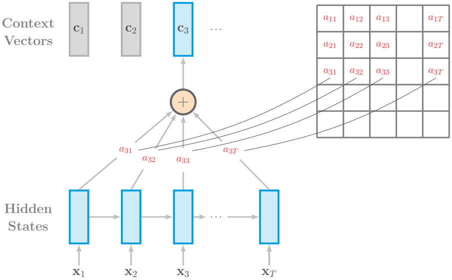
Attention Map



Attention Map

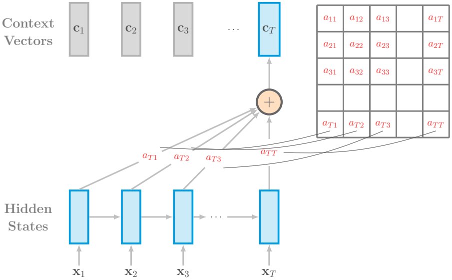


Attention Map



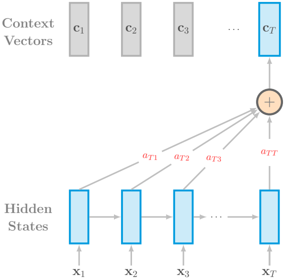
Attention Visualization

Attention Map



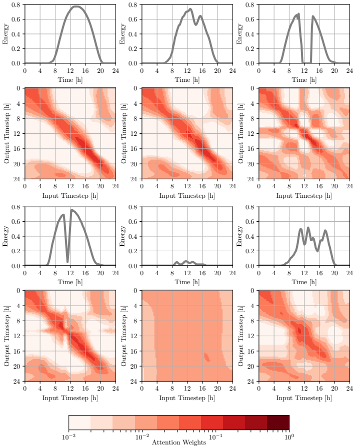
Examples

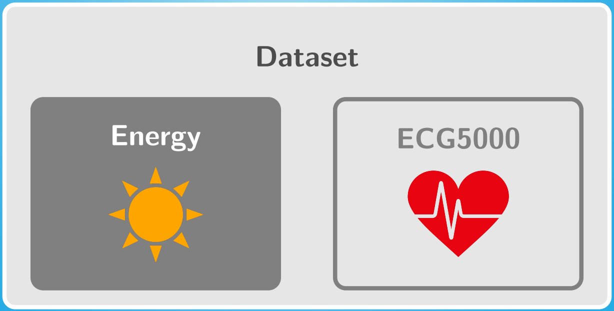
Attention Map

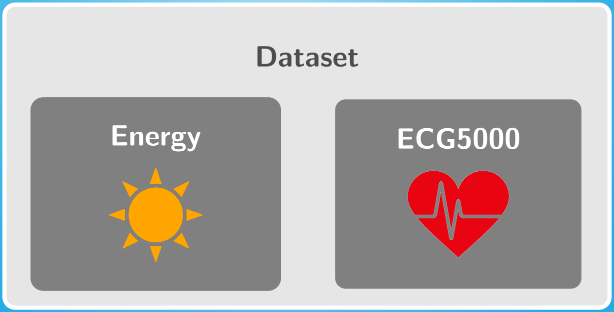


a_{11}	a_{12}	a_{13}		a_{1T}
a_{21}	a_{22}	a_{23}		a_{2T}
a_{31}	a_{32}	a_{33}		a_{3T}
a_{T1}	a_{T2}	a_{T3}		a_{TT}

Examples

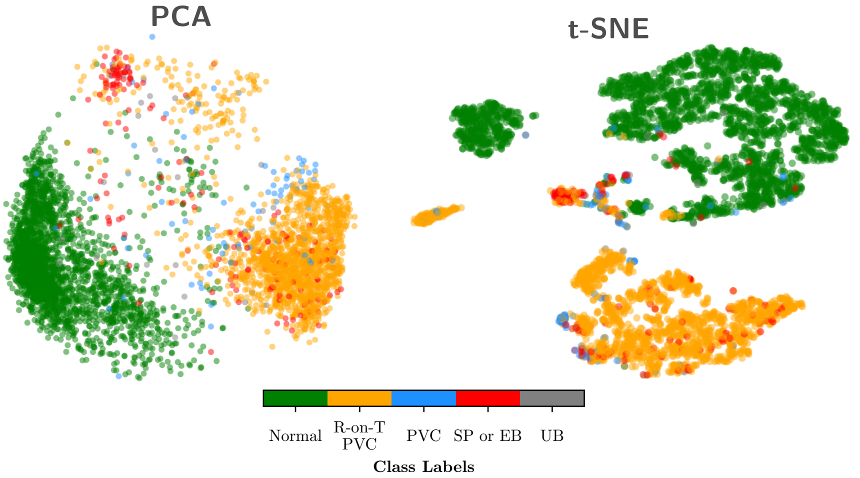



Dataset
Energy
ECG5000


Dataset
Energy
ECG5000

Each datapoint $\rightarrow$ a sequence of length T

$T = 140$ (one heartbeat)
$d_{\mathbf{z}} = 5$



Metric	Hierarchical	Spectral	k-Means++	Wasserstein	SVM
AUC	0.9569	0.9591	0.9591	**0.9819**	0.9836
Accuracy	0.9554	0.9581	**0.9596**	0.9510	0.9843
Precision	**0.9585**	0.9470	0.9544	0.9469	0.9847
Recall	0.9463	0.9516	**0.9538**	0.9465	0.9843
F_1-score	0.9465	0.9474	**0.9522**	0.9461	0.9844

Source	S/U	Model	AUC	Acc	F_1
Proposed	S	VRAE+SVM	**0.9836**	**0.9843**	**0.9844**
	U	VRAE+Clust/W	**0.9819**	**0.9596**	**0.9522**
Lei *et al.*, 2017	S	SPIRAL-XGB	0.9100	-	-
Karim *et al.*, 2017	S	F-t ALSTM-FCN	-	0.9496	-
Malhotra *et al.*, 2017	S	SAE-C	-	0.9340	-
Liu *et al.*, 2018	U	oFCMdd	-	-	0.8084

Metric	Hierarchical	Spectral	k-Means++	Wasserstein	SVM
	Unsupervised				Supervised
AUC	0.9569	0.9591	0.9591	**0.9819**	0.9836
Accuracy	0.9554	0.9581	**0.9596**	0.9510	0.9843
Precision	**0.9585**	0.9470	0.9544	0.9469	0.9847
Recall	0.9463	0.9516	**0.9538**	0.9465	0.9843
F_1-score	0.9465	0.9474	**0.9522**	0.9461	0.9844

Source	S/U	Model	AUC	Acc	F_1
Proposed	S	VRAE+SVM	**0.9836**	**0.9843**	**0.9844**
	U	VRAE+Clust/W	**0.9819**	**0.9596**	**0.9522**
Lei *et al.*, 2017	S	SPIRAL-XGB	0.9100	-	-
Karim *et al.*, 2017	S	F-t ALSTM-FCN	-	0.9496	-
Malhotra *et al.*, 2017	S	SAE-C	-	0.9340	-
Liu *et al.*, 2018	U	oFCMdd	-	-	0.8084

Unsupervised | Supervised

Metric	Hierarchical	Spectral	k-Means++	Wasserstein	SVM
AUC	0.9569	0.9591	0.9591	**0.9819**	0.9836
Accuracy	0.9554	0.9581	**0.9596**	0.9510	0.9843
Precision	**0.9585**	0.9470	0.9544	0.9469	0.9847
Recall	0.9463	0.9516	**0.9538**	0.9465	0.9843
F_1-score	0.9465	0.9474	**0.9522**	0.9461	0.9844

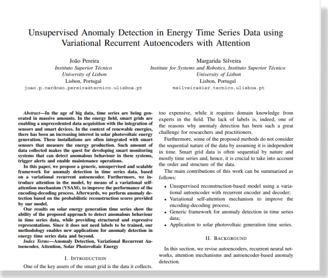
Source	S/U	Model	AUC	Acc	F_1
Proposed	S	VRAE+SVM	**0.9836**	**0.9843**	**0.9844**
	U	VRAE+Clust/W	**0.9819**	**0.9596**	**0.9522**
Lei *et al.*, 2017	S	SPIRAL-XGB	0.9100	-	-
Karim *et al.*, 2017	S	F-t ALSTM-FCN	-	0.9496	-
Malhotra *et al.*, 2017	S	SAE-C	-	0.9340	-
Liu *et al.*, 2018	U	oFCMdd	-	-	0.8084

Conclusions & Future Work

The proposed approach is:

- Effective on detecting anomalies in time series data;
- Suitable for both **univariate and multivariate** data;
- **Efficient**: inference and anomaly scores computation is fast;
- Works with other kinds of sequential data (e.g., **text**, **videos**);
- Extensible to a **multi-class framework** that allows discrimination between anomalies.

Publications



Unsupervised Anomaly Detection in Energy Time Series Data using Variational Recurrent Autoencoders with Attention

João Pereira
Instituto Superior Técnico
University of Lisbon
Lisbon, Portugal
joao.p.cardoso.pereira@tecnico.ulisboa.pt

Margarida Silveira
Institute for Systems and Robotics, Instituto Superior Técnico
University of Lisbon
Lisbon, Portugal
msilveira@isr.tecnico.ulisboa.pt

***Abstract*—In the age of big data, time series are being generated in massive amounts. In the energy field, smart grids are enabling a unprecedented data acquisition with the integration of sensors and smart devices. In the context of renewable energies, there has been an increasing interest in solar photovoltaic energy generation. These installations are often integrated with smart sensors that measure the energy production. Such amount of data collected makes the quest for developing smart monitoring systems that can detect anomalous behaviour in these systems, trigger alerts and enable maintenance operations.**

In this paper, we propose a generic, unsupervised and scalable framework for anomaly detection in time series data, based on a variational recurrent autoencoder. Furthermore, we introduce attention in the model, by means of a variational self-attention mechanism (VSAM), to improve the performance of the encoding-decoding process. Afterwards, we perform anomaly detection based on the probabilistic reconstruction scores provided by our model.

Our results on solar energy generation time series show the ability of the proposed approach to detect anomalous behaviour in time series data, while providing structured and expressive representations. Since it does not need labels to be trained, our methodology enables new applications for anomaly detection in energy time series data and beyond.

***Index Terms*—Anomaly Detection, Variational Recurrent Autoencoder, Attention, Solar Photovoltaic Energy**

I. INTRODUCTION

One of the key assets of the smart grid is the data it collects.

too expensive, while it requires domain knowledge from experts in the field. The lack of labels is, indeed, one of the reasons why anomaly detection has been such a great challenge for researchers and practitioners.

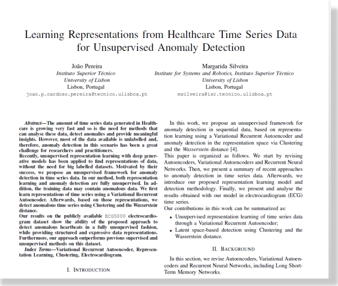
Furthermore, some of the proposed methods do not consider the sequential nature of the data by assuming it is independent in time. Smart grid data is often sequential by nature and mostly time series and, hence, it is crucial to take into account the order and structure of the data.

The main contributions of this work can be summarized as follows:

- Unsupervised reconstruction-based model using a variational autoencoder with recurrent encoder and decoder;
- Variational self-attention mechanism to improve the encoding-decoding process;
- Generic framework for anomaly detection in time series data;
- Application to solar photovoltaic generation time series.

II. BACKGROUND

In this section, we revise autoencoders, recurrent neural networks, attention mechanisms and autoencoder-based anomaly detection.



Learning Representations from Healthcare Time Series Data for Unsupervised Anomaly Detection

João Pereira
Instituto Superior Técnico
University of Lisbon
Lisbon, Portugal
joao.p.cardoso.pereira@tecnico.ulisboa.pt

Margarida Silveira
Institute for Systems and Robotics, Instituto Superior Técnico
University of Lisbon
Lisbon, Portugal
msilveira@isr.tecnico.ulisboa.pt

***Abstract*—The amount of time series data generated in Healthcare is growing very fast and so is the need for methods that can analyse these data, detect anomalies and provide meaningful insights. However, most of the data available is unlabelled and, therefore, anomaly detection in this scenario has been a great challenge for researchers and practitioners.**
Recently, unsupervised representation learning with deep generative models has been applied to find representations of data, without the need for big labelled datasets. Motivated by their success, we propose an unsupervised framework for anomaly detection in time series data. In our method, both representation learning and anomaly detection are fully unsupervised. In addition, the training data may contain anomalous data. We first learn representations of time series using a Variational Recurrent Autoencoder. Afterwards, based on those representations, we detect anomalous time series using Clustering and the *Wasserstein* distance.
Our results on the publicly available `ECG5000` electrocardiogram dataset show the ability of the proposed approach to detect anomalous heartbeats in a fully unsupervised fashion, while providing structured and expressive data representations. Furthermore, our approach outperforms previous supervised and unsupervised methods on this dataset.

***Index Terms*—Variational Recurrent Autoencoder, Representation Learning, Clustering, Electrocardiogram.**

I. INTRODUCTION

In this work, we propose an unsupervised framework for anomaly detection in sequential data, based on representation learning using a Variational Recurrent Autoencoder and anomaly detection in the representation space via Clustering and the *Wasserstein* distance [4].
This paper is organized as follows. We start by revising Autoencoders, Variational Autoencoders and Recurrent Neural Networks. Then, we present a summary of recent approaches to anomaly detection in time series data. Afterwards, we introduce our proposed representation learning model and detection methodology. Finally, we present and analyse the results obtained with our model in electrocardiogram (ECG) time series.
Our contributions in this work can be summarized as:

- Unsupervised representation learning of time series data through a Variational Recurrent Autoencoder;
- Latent space-based detection using Clustering and the Wasserstein distance.

II. BACKGROUND

In this section, we revise Autoencoders, Variational Autoencoders and Recurrent Neural Networks, including Long Short-Term Memory Networks.

Accepted for Oral Presentation at the 17th IEEE International Conference on Machine Learning and Applications

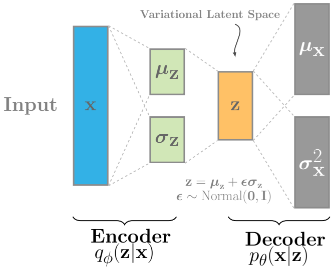
Submitted to the IEEE International Conference on Big Data and Smart Computing

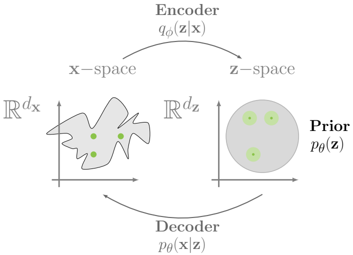
Thank you for your attention!

Appendix

The Variational Autoencoder

- Deep generative model rooted in Bayesian inference





$$p_\theta(\mathbf{x}) = \int_{\mathbf{z}} p_\theta(\mathbf{z}) p_\theta(\mathbf{x}|\mathbf{z}) d\mathbf{z} \qquad p_\theta(\mathbf{z}|\mathbf{x}) = \frac{p_\theta(\mathbf{z}) p_\theta(\mathbf{x}|\mathbf{z})}{p_\theta(\mathbf{x})}$$

- **Objective:** Maximize the Evidence Lower Bound (ELBO)

$$\log p_\theta(\mathbf{x}) \geq \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - \mathcal{D}_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z})\right)}_{:=\mathcal{L}_{\mathrm{ELBO}}(\theta,\phi;\ \mathbf{x})}$$

Kingma & Welling, **Auto-Encoding Variational Bayes**, ICLR'14

Rezende *et al.*, **Stochastic Backpropagation and Approximate Inference in Deep Generative Models**, ICML'14

Derivation of the Evidence Lower Bound (ELBO)

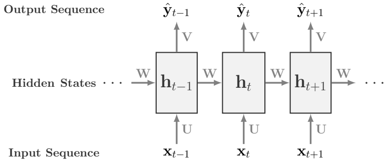
$$
\begin{aligned}
\log p_\theta(\mathbf{x}) &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\big[\log p_\theta(\mathbf{x})\big] \\
&= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log \frac{p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z})}{p_\theta(\mathbf{z}|\mathbf{x})}\right] \\
&= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log \frac{p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z})}{p_\theta(\mathbf{z}|\mathbf{x})}\frac{q_\phi(\mathbf{z}|\mathbf{x})}{q_\phi(\mathbf{z}|\mathbf{x})}\right] \\
&= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\big[\log p_\theta(\mathbf{x}|\mathbf{z})\big] - \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z})}\right] + \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z}|\mathbf{x})}\right] \\
&= \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\big[\log p_\theta(\mathbf{x}|\mathbf{z})\big] - \mathcal{D}_{\mathrm{KL}}\big(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})\big)}_{=\mathcal{L}_{\mathrm{ELBO}}(\theta,\phi;\mathbf{x})} + \underbrace{\mathcal{D}_{\mathrm{KL}}\big(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}|\mathbf{x})\big)}_{\geq 0}
\end{aligned}
$$

What if data are not i.i.d. in time?

(e.g., time series, text, videos)

RNNs capture the temporal dependencies of the data

- Real-valued hidden state $\mathbf{h}_t$
- Feedback connection
- Parameters shared across timesteps

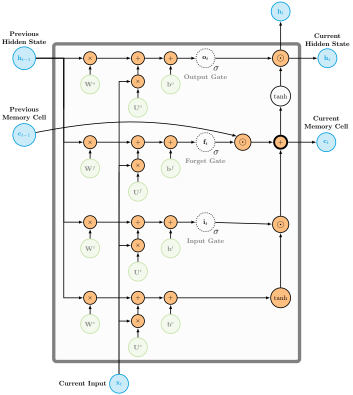


$$\mathbf{h}_t = f(\mathbf{U}\mathbf{x}_t + \mathbf{W}\mathbf{h}_{t-1})$$

f is typically a $\tanh$ or $\mathrm{sigmoid}$

Long Short-Term Memory Network

- Proposed to solve the vanishing gradient problem
- New cell and three gates



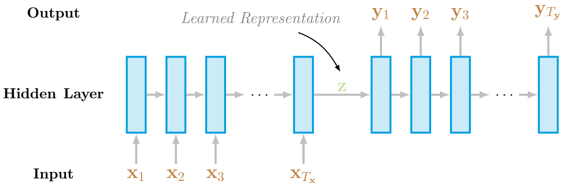
- Updates:

$$\mathbf{i}_t = \sigma(\mathbf{W}_i \mathbf{h}_{t-1} + \mathbf{U}_i \mathbf{x}_t + \mathbf{b}_i)$$
$$\mathbf{f}_t = \sigma(\mathbf{W}_f \mathbf{h}_{t-1} + \mathbf{U}_f \mathbf{x}_t + \mathbf{b}_f)$$
$$\mathbf{o}_t = \sigma(\mathbf{W}_o \mathbf{h}_{t-1} + \mathbf{U}_o \mathbf{x}_t + \mathbf{b}_o)$$
$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tanh(\mathbf{W}_c \mathbf{h}_{t-1} + \mathbf{U}_c \mathbf{x}_t + \mathbf{b}_c)$$
$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t)$$

Hochreiter *el al.*, **Long-Short Term Memory**, Neural Computation'97

Graves *el al.*, **Bidirectional LSTM Networks for Improved Phoneme Classification and Recognition**, ICANN'05

Map sequences to sequences is useful.



Sutskever *el al.*, **Sequence to Sequence Learning with Neural Networks**, NIPS'14

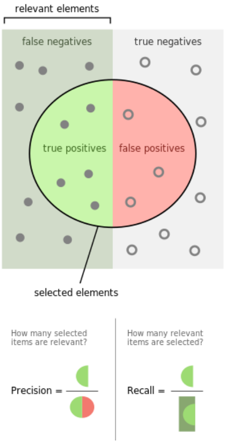
Cho *el al.*, **Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation**, NIPS'14

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F}_1\text{−score} = 2\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$



Legend:

TP : True Positives

TN : True Negatives

FP : False Positives

FN : False Negatives

Classification Metrics

AUC: Area under the Receiver Operating Characteristic Curve (average precision over recalls)