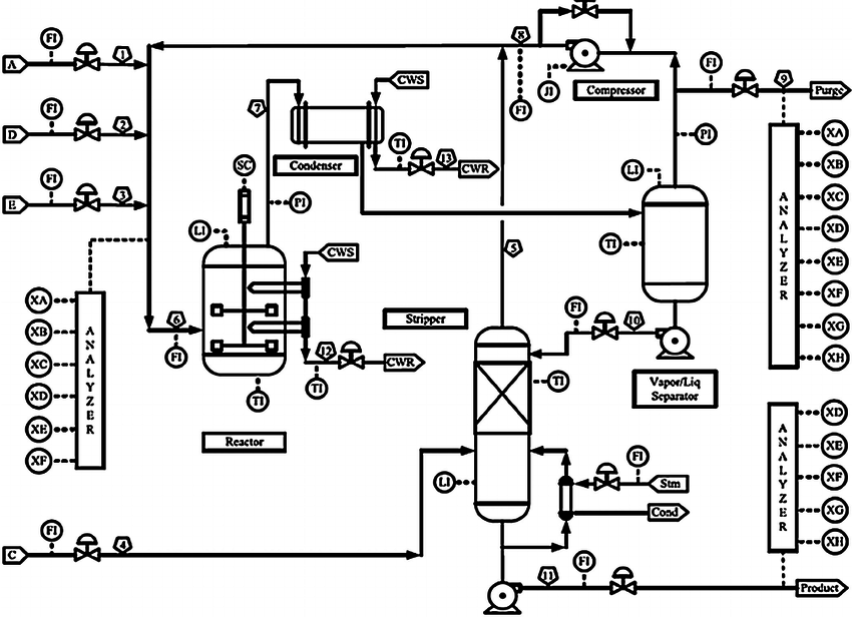
# Introduction

## Context

The machine learning (ML) area has undergone radical change in the last several years. Academia and industry have invested billions of dollars in developing massively complex machine learning algorithms. Deep learning (DL), which deploys multiple layers in the network, achieved great success in image classification (*Ref. ImageNet Classification with Deep Convolutional Neural Networks*), natural language processing (*Ref. Very Deep Transformers for Neural Machine Translation*), etc. One fundamental difference between deep learning and traditional machine learning algorithms is the scalability of data volume. In other words, deep learning models can continuously improve performance given more data and become a prioritized method for big data processing. One application scenario of big data processing is anomaly detection in industrial production.

The industrial processes have gradually become extremely complex to fulfill the requirement of high production efficiency and quality. More and more monitor and controllable variables are introduced, which pose challenges for system control. Early detection of anomalies and implementation of correspondence measures are critical for safety and effective operation. For anomaly detection, as in (*Ref. A survey of the application of basic data-driven and model-based methods in process monitoring and fault diagnosis*) discussed, the methods are generally classified into two types: data-driven and model-based. The latter usually requires expert-level knowledge and experience to build the sophisticated model, which is difficult to achieve in reality. Compared to the model-based method, the data-driven method such as machine learning relies on analyzing large amounts of data, which provides a practical solution for various industrial productions. Due to the feasibility and practicability, machine learning has become an essential analysis method in the industrial process.

This thesis investigates machine learning methods for anomaly detection in Tennessee Eastman Process (TEP). The TEP proposed in (*Ref. A PLANT-WIDE INDUSTRIAL PROCESS PROBLEM*) is an industrial chemical process model and can be used as a benchmark model to evaluate the control system's effectiveness. The TEP mainly consists of five process units: a reactor where an exothermic reaction occurs, a product condenser, a vapor-liquid separator, a compressor and a stripper. The process diagram is shown in (*fig*). There are also simplified TEP versions as (*Ref. Model predictive control of a continuous, nonlinear, two-phase reactor*) proposed.

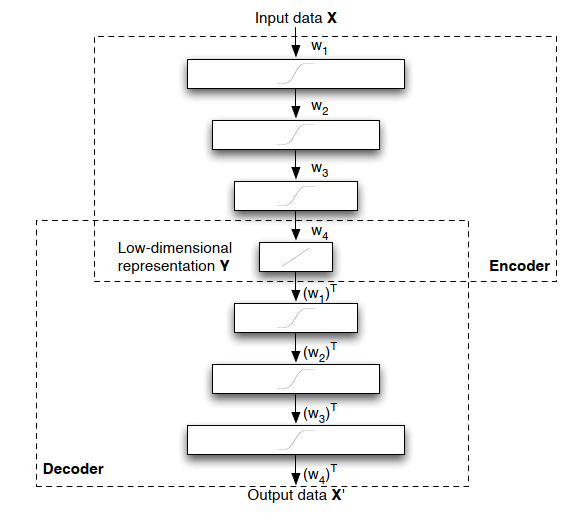


(*Ref. A PLANT-WIDE INDUSTRIAL PROCESS PROBLEM*)

In general, TEP is a nonlinear unstable system with multi-input/outputs variables and characterized by fast and slow dynamic behaviors (*Ref. Process in Modelica Process in Modelica Process in Modelica*). Due to this reason, the data generated in TEP has the following characteristics:

* **high dimensionality,**
* **nonlinearity,**
* **non-Gaussian distribution.**

High-dimensional data pose a severe challenge for machine learning algorithms, and some algorithms may even break down in practice. In (*Ref. A Few Useful Things to Know about Machine Learning*), they gave two descriptive explanations for the *curse of dimensionality*. The first explanation is, the influence of noise accumulates as dimensionality increases, i.e., the noise from irrelated features may suppress relevant features' influence. The second explanation is that the similarity between data points increases as dimension expands. For example, when using distance as criteria to decide if data points are neighbors, the number of neighbors of data points increase as the dimensionality increases. To mitigate the impact of high dimensionality, feature selection (*Ref. Feature Engineering and Selection*) and feature extraction (dimension reduction) are proposed (*Ref. Fault Detection of the Tennessee Eastman Process Using Improved PCA and Neural Classifier*). The concept of feature selection is to select the most relevant features based on correlation metrics and expect to improve the machine learning model's performance (*Ref. Feature Selection for Fault Detection Systems: Application to the Tennessee Eastman Process*). The feature extraction represents the high dimensional data into the lower dimension and extracts the hidden structures (*Ref. A Tutorial on Principal Component Analysis*). As a nonlinear feature extraction method, autoencoder (AE) use the encoder part to represent the input data into a low dimension in the middle hidden layers (*bottleneck),* then uses the decoder part to map the data back in high dimensional space (*Ref. Dimensionality Reduction: A Comparative Review*), as (*fig*) shows. For autoencoders, it is trained to copy the input to its output. One key point of this thesis is to explore the application of autoencoder in TEP's anomaly detection.



(*Ref. Dimensionality Reduction: A Comparative Review*)

??? To be added

## Problem Statement

The main goal of this research is to evaluate the feasibility of deep learning, especially the autoencoder, in TEP's anomaly detection. Analyzing the performance of different machine learning models can help enhance the understanding of anomaly detection and provide the possibility for further improvement. As the literature research implies, there is no detailed work comparing different anomaly detection methods for TEP. This thesis seeks to fill the gaps.

## Research Questions

This thesis's main contribution is to provide a comprehensive comparison between the traditional machine learning method and the deep learning method for TEP anomaly detection. Then analyze the reasons for the difference and implement an improved deep learning model for anomaly detection. More specifically, this research is to answer the following questions:

* What is the current status of research in TEP's anomaly detection, and what machine learning algorithms are already applied?
* Develop evaluation metrics and compare the performance of different algorithms using the TEP dataset.
* Analyze the reasons for the performance difference. Based on the evaluation results, implement a deep learning method.

## Organization (may change)

Chapter 1 introduces the topic and provides an overview of the problem. Chapter 2 reviews the background and prior works about machine learning algorithms. Chapter 3 describes the structure and function of machine learning models for the experiment in detail. Chapter 4 includes the setting of the experiment, the adjustment of model parameters and result analysis. Chapter 5 summarizes the works and points out the future research directions.

# Background Theory

This chapter provides the necessary background and theoretical knowledge to understand the methodology in chapter 4. It first introduces the basic concepts of machine learning, then gives a broad overview of traditional machine learning and deep learning. Finally, it reviews the prior work and compares the performance of different machine learning methods.

## Machine learning concept

It is not a trivial task to give a precise definition of machine learning. If a machine can realize self-adjustment and improve the performance given more external information, we might say it learns (*Ref. INTRODUCTION TO MACHINE LEARNING - Nils J. Nilsson*). Machine learning is one of the fast-evolving areas in both academia and industry in the last decades. As an important branch of machine learning, deep learning has achieved huge success in pattern recognition and translation (*Ref. The Unreasonable Effectiveness of Deep Learning in Artificial Intelligence*). This is due to several reasons: 1) deep learning methods can realize sustained performance improvement when given more data exceeding the threshold of older learning algorithms (*fig below*); 2) deep learning can reconstruct the unknown structure and is based on the distribution of inputs to find appropriate representation (*Ref. Deep Learning of Representations for Unsupervised and Transfer Learning*).



(*Ref.* <https://www.slideshare.net/ExtractConf>)

Machine learning algorithms can be classified into different groups based on the criteria. By learning style (*Ref. A Survey on Machine Learning: Concept, Algorithms and Applications*), it could be categorized into:

* Supervised learning: Each input data has a corresponded label, e.g., Positive/Negative. The machine learning algorithm (classifier) can find the mapping between input data and labels by training. Further, the trained classifier can predict the label of data.
* Unsupervised learning: The input data has no label, unlike in supervised learning. Machine learning algorithms' task is to find the typical input data pattern and cluster it into groups.
* Semi-supervised learning: This can be seen as a hybrid version of supervised learning and unsupervised learning because the input data is only partially labeled. The task is to predict the label of the data and execute the clustering.
* Reinforcement learning: The algorithms are trained to choose behavior in a specific environment to obtain the most reward.

The machine learning algorithms for TEP's anomaly detection can also be classified into traditional- and deep learning algorithms in view of prior work.

* Traditional algorithms: Principal Component Analysis (PCA) (*Ref. Fault Detection of the Tennessee Eastman Process Using Improved PCA and Neural Classifier*), Support Vector Machine (SVM) (*Ref. Study on Support Vector Machine-Based Fault Detection in Tennessee Eastman Process*), Clustering (*Ref. Unsupervised Feature Selection Based on Fuzzy Clustering for Fault Detection of the Tennessee Eastman Process*), Random Forest Tree (*Ref. A Dynamic Nonlinear Process Fault Diagnosis Method Using Canonical Rotation Forest*), ???adding more
* Deep learning algorithms: Long Short-Term Memory Recurrent Neural Network(*Ref. Fault Detection and Diagnosis in a Chemical Process using Long Short-Term Memory Recurrent Neural Network*), Stacked Sparse Autoencoders (*Ref. Mutual Information−Dynamic Stacked Sparse Autoencoders for Fault Detection*), Variational Autoencoders (*Ref. Comparison of Semi-supervised Deep Neural Networks for Anomaly Detection in Industrial Processes*), ???adding more

Certain traditional algorithms are primarily based on statistical knowledge or assumption and have some limitations. For example, PCA assumes that the process data are linear correlated and in Gaussian distribution (*Ref. A Tutorial on Principal Component Analysis*). Another limitation of PCA is the information loss during the projection of high-dimensional data into low-dimensional data (*Ref. A NONLINEAR SUPPORT VECTOR MACHINE BASED FEATURE SELECTION APPROACH FOR FAULT DETECTION AND DIAGNOSIS: APPLICATION TO THE TENNESSEE EASTMAN PROCESS*). SVM solves the nonlinear and non-Gaussian distribution issues by transforming the data in a high dimension space with a kernel function. Then SVM uses the optimal hyperplane to separate the data and makes the classification (*Ref. A NONLINEAR SUPPORT VECTOR MACHINE BASED FEATURE SELECTION APPROACH FOR FAULT DETECTION AND DIAGNOSIS: APPLICATION TO THE TENNESSEE EASTMAN PROCESS*). But the SVM classifier's performance depends strongly on the kernel function and is prone to overfitting (*Ref. Overcome Support Vector Machine Diagnosis Overfitting*). Compared to traditional algorithms, deep learning algorithms composed of the artificial neural network have many advantages and provide a universal solution for anomaly detection (*Ref. Approximation with Artificial Neural Networks*).

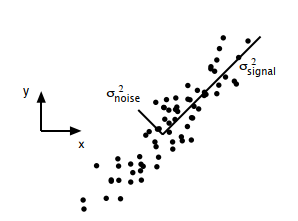
## Traditional algorithms

The traditional machine learning algorithms in this thesis refers to the algorithms in which no artificial neural network is involved. The purpose of the distinction is to investigate the impact of structural complexity, i.e., if adding more layers in deep learning can improve performance in anomaly detection.

### Principal component analysis

As a feature extraction (dimension reduction) method, the principal component analysis's goal is to represent the data in such a way that only the most *meaningful* features remain. More specifically, the representation of data is achieved by projecting the data on a new orthonormal basis. The assumption for the projecting of data is the linearity of data, i.e., the data are the linear combination of the basis vectors. For example, we want to transform the 2-dimensional data points into 1-dimensional data points. Signal-to-noise ratio (SNR) is used as a measure to evaluate the meaningfulness of data points in an axis direction.

The axis is defined as the signal (meaningful) direction because the data points have the largest variances in that direction. The axis is defined as the noise (meaningless) direction in which the data points have the smallest variance. Those two axes are orthogonal. As the (*figure*) shows, a meaningful way of representing the data is to project the data in axis direction and ignore the projection of data in axis. In this way, we get the 1-dimensional representation of the data points.

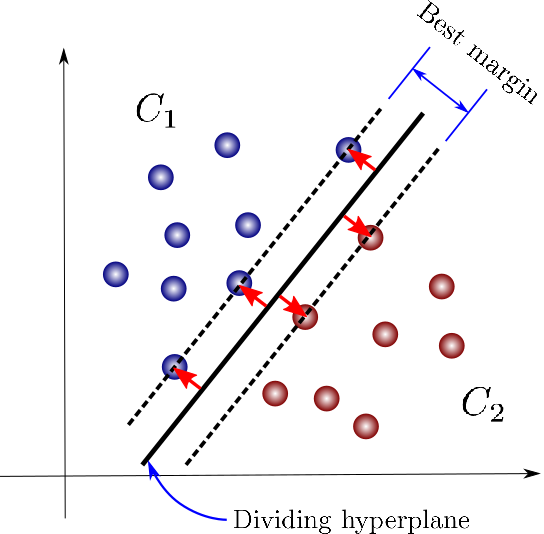


(*Ref. A Tutorial on Principal Component Analysis*)

Although PCA is a non-parametric approach and easy to implement, it has several limitations: 1) the reduced representation can cause information loss, 2) PCA can not be used for nonlinear data without further modification, 3) orthogonality of basis may not hold in some cases (*Ref. A Tutorial on Principal Component Analysis*).

### Support vector machine

Support vector machine is a classification model which can be used to separate the data by transforming it into high dimensional space with the kernel technique. As the (*figure*) shows, certain data points are chosen as the support vector, and the area between the support vectors is called margin. SVM solves the classification problem by finding a hyperplane that can separate the data best and realize the maximal margin.



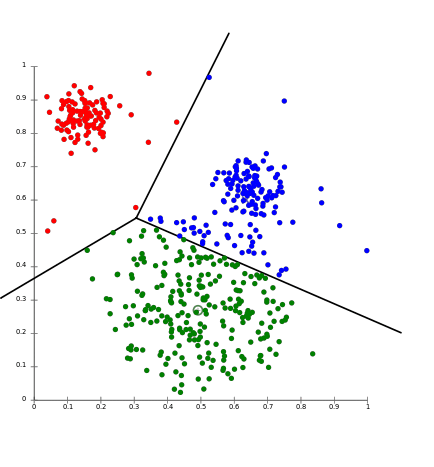
(*Ref. https://towardsdatascience.com/support-vector-machines-for-classification-fc7c1565e3*)

One fact about the classification task is that mapping the data into another space may significantly improve the performance. SVM takes advantage of this fact and solves the nonlinear separable issues using the kernel, transforming the data point into high- or infinite-dimensional space.

As a classification method, SVM works effectively in high-dimensional space. But it doesn't provide a probability estimate. Instead, it gives the classification result directly. Besides, the selection of kernels has a crucial impact on classification performance. Finally, from practical experience, the computation cost for SVM is much hight than other classifiers such as the random forest tree (*Ref. An introduction to support vector machines*).

### Clustering

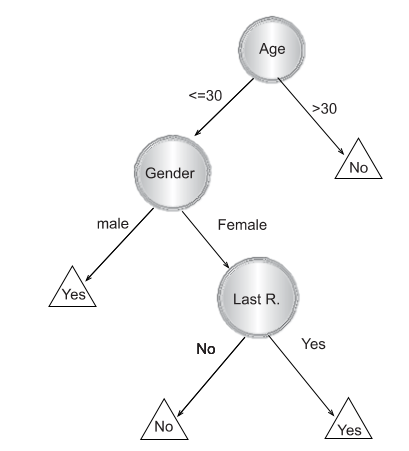
Clustering is a kind of unsupervised learning, and the goal is to assign data points into groups. A representative clustering algorithm is the K-means algorithm. K-means algorithm works by iterating between two steps until it reaches the termination criteria(*Ref. Top 10 algorithms in data mining*). In the initialization phase, several centroids need to be chosen first. In step 1, data points are assigned to different groups based on the distance to the centroids. Then in step 2, the centroids are recalculated based on the data assigned to its group. K-means iterated between those two steps until the algorithm converges. As (*fig*) shows, three centroids are chosen before the execution. After reaching the termination condition, all the data points are assigned into three groups. K-means algorithm is very sensitive to the initialization setting. Different centroids chosen at the initialization phase may lead to different clustering results. For TEP anomaly detection, clustering algorithms can be used for feature selection and improve detection performance.



(*Ref.* [*https://aws.amazon.com/cn/blogs/machine-learning/k-means-clustering-with-amazon-sagemaker/*](https://aws.amazon.com/cn/blogs/machine-learning/k-means-clustering-with-amazon-sagemaker/))

### Decision tree

As a predictive model, the decision tree can be used for both classification and regression tasks. In this thesis, the decision tree is mainly used as a classifier to identify TEP abnormalities. The decision tree works by partition the instance space recursively and decides each node. As (*fig*) shows, a decision tree is consists of nodes and edges. The nodes without outgoing edges are called leaves or terminal nodes, and the nodes without incoming edges are called the root. Classification begins from the root node, after which each node splits the instance space into sub-spaces using a specific evaluation function until it reaches the leaves. The leaves represent the classification result, and edges correspond with certain sub-spaces.



(*Ref. Data Mining with Decision Trees: Theory and Applications, chapter 1*)

The decision tree is an easy to explain machine learning algorithms because of the similarity to the human decision process. But it is not very robust and has high variance because a minor change in the training data can lead to a different decision tree construction. To further improve the decision tree's performance, the random forest tree is proposed (*Ref. Random Forests Leo Breiman 2001*).

### Random forest tree

Random forest tree is also a predictive model based on the decision tree for classification or regression. Since the decision tree has poor generalization performance, (*Ref. The Random Subspace Method for Constructing Decision Forests*), (*Ref. Random Forests Leo Breiman 2001*) proposed random sub-space selection and bagging techniques, which leads to the invention of the random forest tree. The main idea of bagging is to construct many decision trees and make predictions using the *majority vote* (*Ref. An Introduction To Statistical Learning with Applications in R, chapter 8*). If the decision trees are not correlated after training, the majority vote of trees can decrease the variance. Random sub-space selection solves the problem between overfitting and high accuracy by selecting a random subset of features when splitting each decision tree's sub-space.



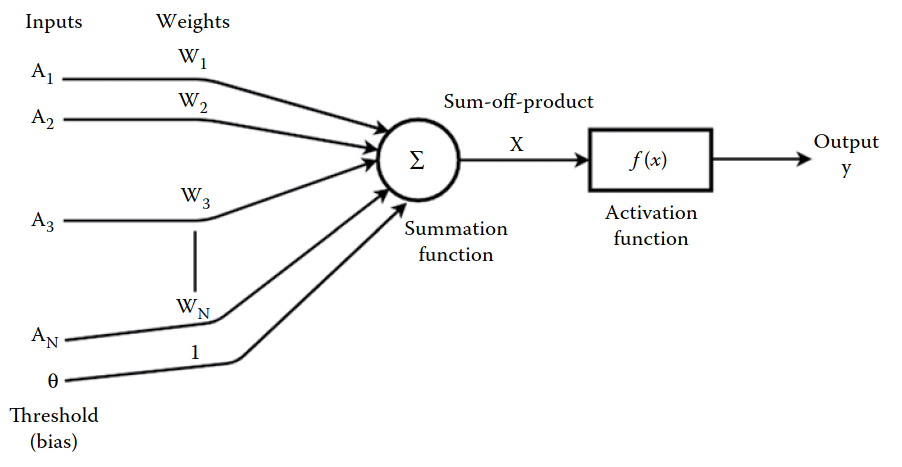
(*Ref. https://community.tibco.com/wiki/random-forest-template-tibco-spotfire*)

As (*fig*) shows, a random forest tree makes the classification based on the majority vote of Tree-1, Tree-2 and Tree-3. Since each tree is trained uncorrelated, the majority vote can help to provide better generalization and accuracy.

## Deep learning

An artificial neural network (ANN) is inspired by studies on biological neurons and built with connected units, as (*fig*) shows. The input signals flow through the weights section, and the signal strength is adjusted according to the weight value. Then the summation function accumulates all the weighted signals and transmits them to an activation function. The bias signal has the function as an offset. A commonly used activation function is sigmoid function ranging from -1 to 1 and is defined by:

Stacking and connecting such units in different directions regarding the signal flows can build various artificial neural networks. (*Ref. Multilayer Feedforward Networks are Universal Approximators*) proved that an artificial neural network could be used as a universal approximator if enough units are in the middle layer. Based on the artificial neural network, deep learning uses multiple layers to solve complicated problems without the human's formal specification of knowledge. Besides, the representation of input data has a strong influence on machine learning performance. Deep learning solved the representation problem by dividing it into simple subproblems, i.e., the complexity of representations at different layers is different. This hierarchy working principle helps deep learning learn more abstract representations and achieve great success (*Ref. Deep Learning by Ian Goodfellow, Yoshua Bengio, Aaron Courville*).

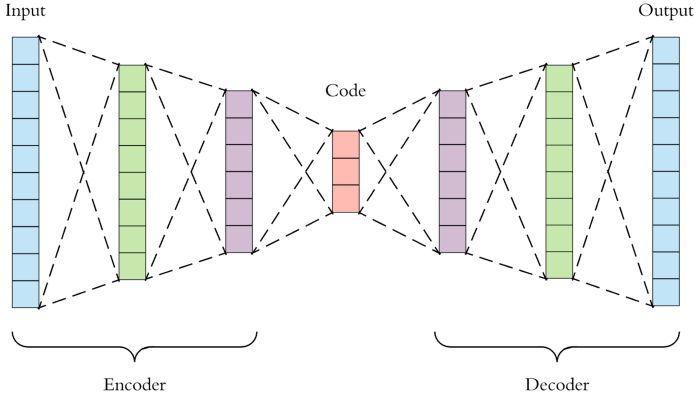


(*Ref. Machine Learning and Iot A Biological Perspective by Shampa Sen, Leonid Datta, Sayak Mitra*)

In the deep neural network history, hard to train is the main obstacle for the development. When the deep neural network is trained with gradient-based learning methods by backpropagation, the gradient may become unstable and lead to gradient exploding or vanish problems (*Ref. Neural Networks and Deep Learning - Michael Nielsen*). (*Ref. Gradient-based learning applied to document recognition*), (*Ref. Greedy Layer-Wise Training of Deep Networks*) and (*Ref. A Fast Learning Algorithm for Deep Belief Nets*) proposed several new approaches such as greedy layer-wise training and global training to solve that problem. Applying activation such as ReLU (Ref. Deep Sparse Rectifier Neural Networks) or residual networks (*Ref. Deep Residual Learning for Image Recognition*) can also ease the training problem. The event that promotes deep learning development is the work (*Ref. ImageNet Classiﬁcation with Deep Convolutional Neural Networks*) which achieves a new score for the image classification task. Besides image classification, deep learning also makes significant progress in speech recognition (*Ref. Deep Speech 2- End-to-End Speech Recognition in English and Mandarin*) and natural language processing (*Ref. Nematus: a Toolkit for Neural Machine Translation*).

### Autoencoder

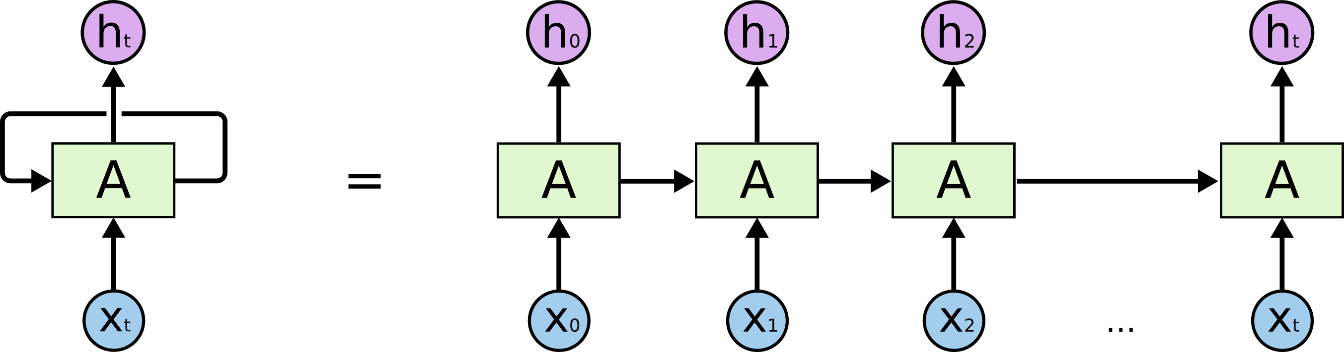
An autoencoder is a type of artificial neural network which learns to output the original input data. As (*fig*) shows, the input is first compressed in by the encoder until the *Code* hidden layer, then expended by the decoder. The *bottleneck* design forces the autoencoder unable to copy the input precisely. The reason is that the *Code* hidden layer helps to learn the useful representation of input data (*Ref. Deep Learning by Ian Goodfellow, Yoshua Bengio, Aaron Courville*). Those representations can be further used for machine learning tasks as classification.



(*Ref. https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798*)

### Recurrent neural network

A recurrent neural network (RNN) is a class of artificial neural networks that can learn the mapping between input sequences and output sequences with the internal state. As (*fig*) shows, RNN uses loops to preserve previous computation's state information and process the input of next time sequence. One observation of the structure of RNN is that it should be able to build a connection between previous information and the current task. However, RNN can exhibit forgetting behavior which limits its representation capability when dealing with long-range sequences (*Ref. Local Feedback Multi-Layered Networks*). Besides, RNN suffers from training problems when long-term memory is involved in the task(*Ref. Learning Long-Term Dependencies with Gradient Descent is Difficult*), (*Ref. The Utility Driven Dynamic Error Propagation Network*). To mitigate the drawbacks of RNN, (*Ref. LONG SHORT-TERM MEMORY*) introduced Long Short-Term Memory



(*Ref. http://colah.github.io/posts/2015-08-Understanding-LSTMs/*)

### Generative Adversary Network

??? To be added