

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



# Deep Learning Approach for Raman Spectroscopy

*M.H. Wathsala N. Jinadasa, Amila C. Kahawalage,  
Maths Halstensen, Nils-Olav Skeie and Klaus-Joachim Jens*

## Abstract

Raman spectroscopy is a widely used technique for organic and inorganic chemical material identification. Throughout the last century, major improvements in lasers, spectrometers, detectors, and holographic optical components have uplifted Raman spectroscopy as an effective device for a variety of different applications including fundamental chemical and material research, medical diagnostics, bio-science, in-situ process monitoring and planetary investigations. Undoubtedly, mathematical data analysis has been playing a vital role to speed up the migration of Raman spectroscopy to explore different applications. It supports researchers to customize spectral interpretation and overcome the limitations of the physical components in the Raman instrument. However, large, and complex datasets, interferences from instrumentation noise and sample properties which mask the true features of samples still make Raman spectroscopy as a challenging tool. Deep learning is a powerful machine learning strategy to build exploratory and predictive models from large raw datasets and has gained more attention in chemical research over recent years. This chapter demonstrates the application of deep learning techniques for Raman signal-extraction, feature-learning and modelling complex relationships as a support to researchers to overcome the challenges in Raman based chemical analysis.

**Keywords:** machine learning, deep learning, neural networks, fluorescence, data analysis

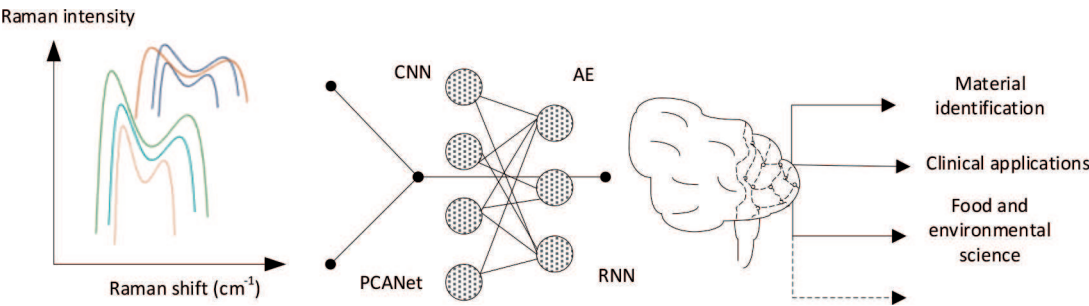
## 1. Introduction

Spectroscopy is an ubiquitous method in natural sciences and engineering for e.g. characterization of materials, molecules or mechanisms, kinetics and thermodynamics of chemical reactions. It is the study of the interaction between electromagnetic radiation and molecules/particles which involves either absorption, emission, or scattering. In Raman spectroscopy, it is the interaction of light with matter which is generating the Raman effect. This effect is the scattering of incoming radiation leading to a change of wavelength or frequency. A Raman spectrum is composed of peaks which show the intensity and wavelength of the Raman scattered light which is due to radiation interaction with individual chemical bond vibrations. These peaks are used to detect, identify, and quantify information about atoms and molecules. Raman spectroscopy is a prominent choice among other spectroscopic techniques, particularly in chemical systems containing

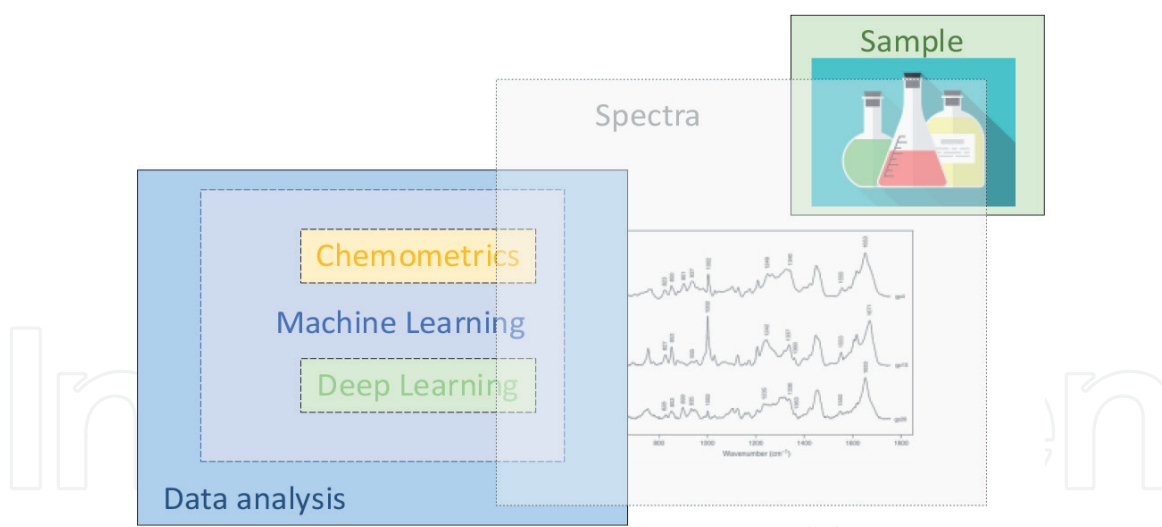
water and/or polar solvents [1]. Weak Raman scattering of water enables in-situ analysis in aqueous chemical systems and in vitro and in vivo analysis in human and sensitive biological systems. While many analytical techniques require sample preparation (such as grinding, glass formation, or tablet pressing) before measurement, Raman analysis can be made on ‘as received’ samples. A measurement can be made within few seconds in a non-destructive, non-contact manner and therefore samples can be retained for other analysis if necessary. Raman scattering of light by molecules was first predicted using classical quantum theory by Smekal in 1923 [2] and experimentally observed by Raman and Krishnan in 1928 [3, 4]. After a century of first ever discovery of Raman fundamentals, today, different types of Raman spectroscopies have been developed such as time-resolved Raman spectroscopy, high pressure Raman spectroscopy, matrix-isolation Raman spectroscopy, Surface-Enhanced Raman Spectroscopy, Raman microscopy and Raman Imaging spectrometry [5]. Throughout the last century, major improvements in lasers, spectrometers, detectors, and holographic optical components have yielded Raman spectroscopy as a dominant tool for molecular verification in a wide range of scientific disciplines.

A primary role of scientists is the extraction of new knowledge from experimental data. Spectroscopic techniques produce profiles containing a high amount of data. It can take significant time and effort to read, interpret and model these data. Cozzolino [6] mentions that the three critical pillars that support the development and implementation of vibrational spectroscopy including Raman, are as the sample (e.g., sampling, methodology), the spectra and the mathematics (e.g., spectral analysis, algorithms, pre-processing, data interpretation, etc.). Thus, data analysis becomes the only flexible option that can be adjusted to assess data extracted from a specific application (i.e. sample) using a given spectroscopic method. Spectroscopic techniques are only as powerful as the information that can be extracted from the resulting spectral data. Simultaneous development of spectroscopy hardware components and data analysis throughout the last five decades, made a radical change for the propagation of spectroscopic techniques in different fields. In case of Raman spectroscopy, instead of having a spectrometer whose volume fills up an entire room including a group of scientists manually reading the spectra, today we have miniature spectroscopic analyzers supported by a computer and software which automatically read, treat, interpret, and summarize measurements within seconds (**Figures 1 and 2**).

The objective of this chapter is to show the role of data analysis to raise and expand the awareness of Raman spectroscopy. The chapter reviews key deep learning strategies under machine learning perspectives, that have been already applied in different Raman applications emphasizing how these strategies have contributed to solving Raman spectroscopic data challenges. The objective is to strengthen the role of data analysis to uplift the capability and standard of Raman spectroscopy.



**Figure 1.**  
*Raman spectroscopy combined with machine learning methods in different applications.*



**Figure 2.**  
*Integration between sample, spectra, and data analysis to develop Raman applications.*

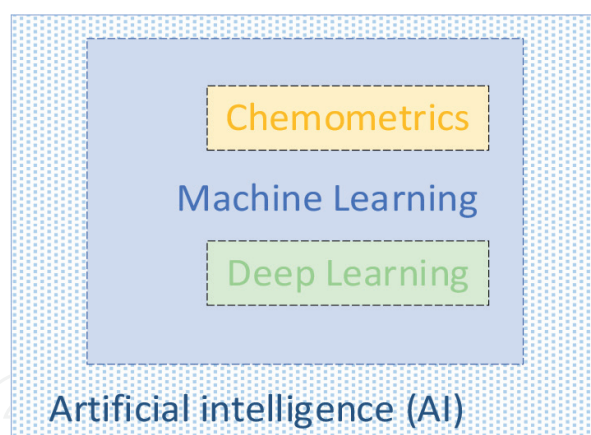
“Deep learning” is a subset of machine learning in artificial intelligence. Many spectroscopists have a background in chemometrics and statistics for chemical analysis, but so far only a few are taking advantage of the potential provided by machine learning. There are many synergies and common concepts applied in between the areas of machine learning and spectroscopy which enhances productive inter-communication. The chapter provides comprehensions by showing how deep learning algorithms increase analytical insight into Raman spectra.

## 2. Deep learning

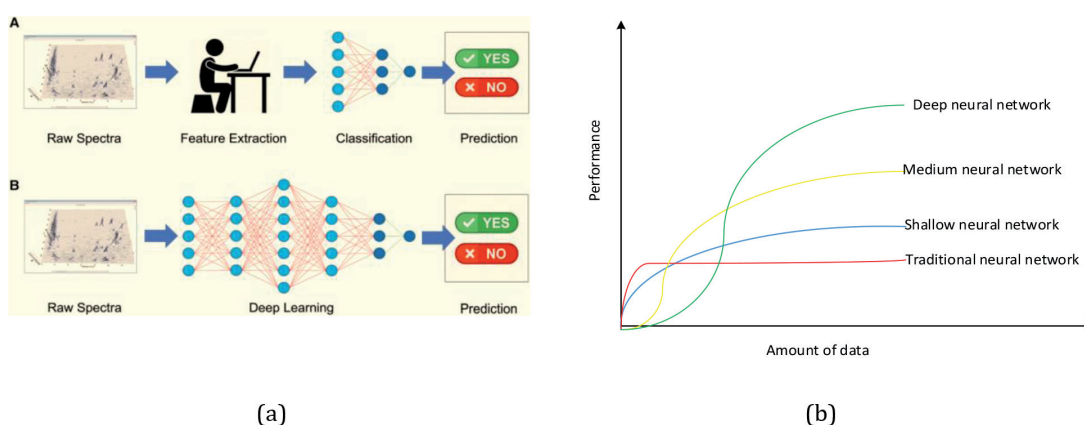
In a simplest way, deep learning can be introduced as a method which teaches computers to do a task. Very often this task is difficult to carry out by human brain due to limited brain capacities and limited time. It is a subset of Machine learning (ML) which is further a subset of artificial intelligence (AI).

Integrating data, information, machines, sensors, and software is a component of transforming conventional ways of human-oriented methods into more digitalized roots. It can level up efficiency and performance of an individual system and its related components by giving more deep insights. Artificial intelligence and machine learning persist to support this transformation.

Artificial intelligence (AI) compasses the science and engineering of making intelligent machines specially computer programs. Machine learning (ML) which is a subset of artificial intelligence, uses algorithms to optimize a certain task by using examples or experience and support AI to learn with explicit programming. Deep learning (DL) is a sub class of machine learning algorithms which consist of learning methods based on artificial neural networks (ANNs). **Figure 3** shows the interconnection of AI, ML, DL and chemometrics. ML algorithms that are not deep learning are referred as shallow learning. A simple explanation to understand the difference between a shallow learning and deep learning algorithm is shown in **Figure 4(a)**. It shows that in shallow learning, feature (useful patterns) extraction and classification are performed in two different stages. For instance, a general practice of a chemist who obtains a vibrational spectrum of an unknown chemical sample, starts with mapping individual peaks. Typically, this is performed by combining the knowledge of chemical vibrational modes and sample. There can be peaks that are not originated from sample chemical properties such as instrument noise or stray lights. The chemist will only utilize the peaks that reveal required



**Figure 3.**  
Artificial intelligence, machine learning, deep learning and chemometrics.



**Figure 4.**  
(a) Comparison between machine learning and deep learning [7]; (b) performance vs. amount of data for traditional ML and DL algorithms.

information about the sample. This process is called feature extraction in ML language. Next, the chemist will proceed to the next step of analysis such as regression or classification. Conversely, in deep learning, feature extraction and subsequent analysis are performed automatically inside the single boundary of DL algorithm.

## 2.1 Chemometrics

For scientists utilizing spectroscopy, chemometrics is a very familiar term linked to data analysis. Chemometrics was established at the beginning of the 1970s by Svante Wold, Bruce L. Kowalski, and D.L. Massart [8, 9]. It is a chemical discipline that uses mathematical, statistical, and other methods employing formal logic to design or select optimal measurement procedures and experiments, and to provide maximum relevant chemical information by analyzing chemical data [10]. Throughout the past 50 years chemometrics revolutionized in the field of spectroscopy through the applications of multivariate calibration, (re)activity modeling, pattern recognition, classification, discriminant analysis, and multivariate process modeling and monitoring [11]. Wold and Sjöström [11] point out two strong trends where the future success of chemometrics remains; 1). Ability of chemometrics to handle the number of 'objects' observations, cases, or samples which is fairly small, and tends to become even smaller with time and 2). Ability of chemometrics to handle big data sets and those which continuously updated with more addition of data in the future. Data sets often remain smaller when experimentation is demanding more resources like time, personnel, laboratory space, instrumentation, chemicals, solvents

and hence is becoming more and more expensive. Big data sets are generated when more samples are measured or several experimental runs are propagated over time for examples in combinatorial chemistry and process monitoring. Vogt [8] explains that maintaining chemometrics as an active and widely recognized research field, requires opening new research areas for chemometricians and without the power of parallel computation, many new and exciting avenues will remain unfeasible. For instance, limiting chemometrics to linear methodologies imposes restrictions because many chemical systems are nonlinear. Chemometrics has its main territory in analytical and measurement science, however fundamentally it can also be considered as a subset of machine learning. The understanding of chemical systems, and the respective underlying behavior, mechanisms, and dynamics, can be facilitated by the development of descriptive, interpretative, and predictive models. Common examples of chemometric techniques which develop such models are principal component analysis (PCA), partial least squares (PLS), linear discriminant analysis (LDA) and support vector machine (SVM). Studies showing the possibility of combining routine chemometrics methods with machine learning algorithms influence to break the stagnancy of chemometrics tools in the chemical laboratory.

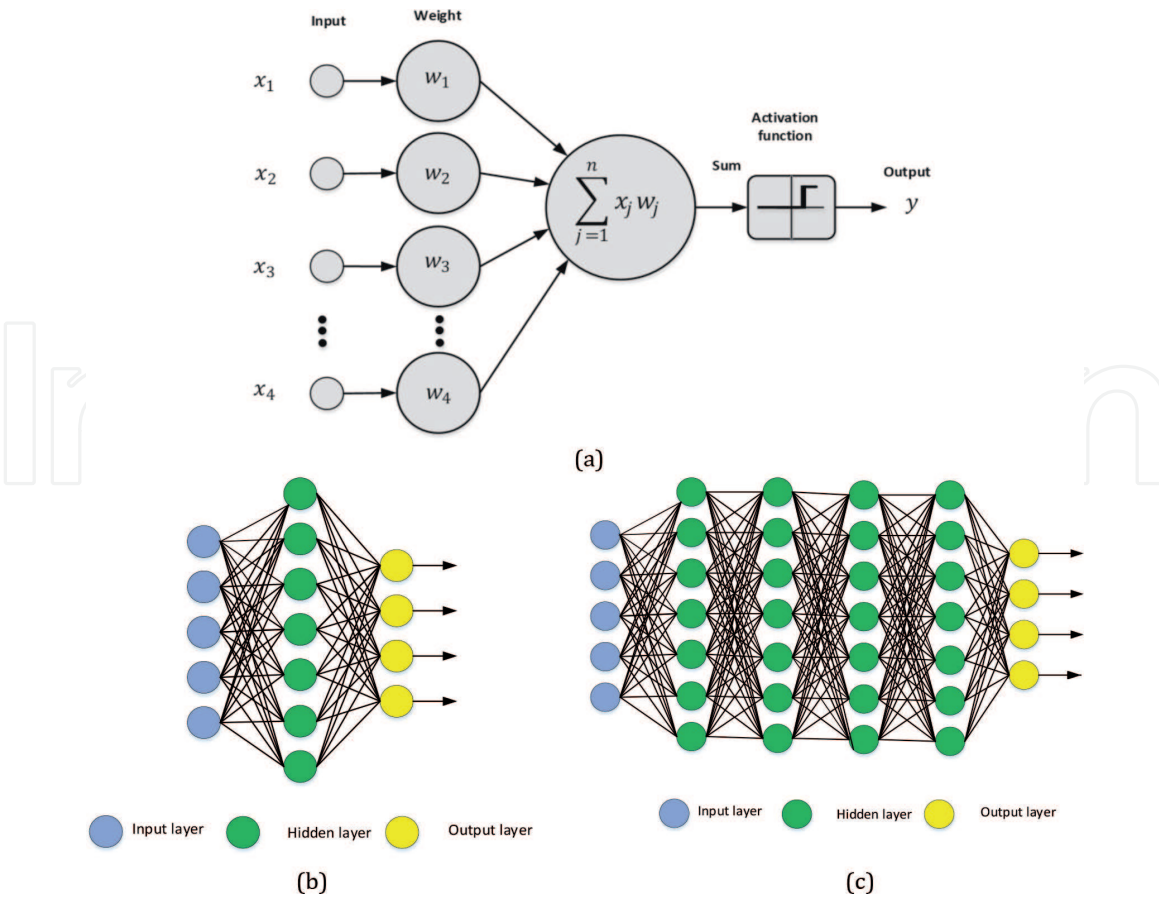
In spectral data analysis, the amount of data plays a decisive role. DL algorithms give better performance for big data sets and as more data are being added. On the other hand, performance of a system which is analyzed by human brain or conventional machine learning algorithms, is limited after a certain size and scale of data. **Figure 4(b)** shows the performance curve for traditional machine learning algorithms and deep learning algorithms. Performance curve for traditional algorithms is saturated after a certain number of data because they are based on handcrafted rules. Creating many rules manually is an erroneous task. For instance, linear regression and random forests (which are traditional ML), tend to plateau at large data volumes. On the contrary, deep learning uses more than one level of non-linear feature transformation and therefore the performance keeps increasing with added data.

Shallow machine learning methods, such as shallow neural networks [12], support vector machines [13, 14], or kernel methods [15], have been applied to Raman spectroscopy with higher success, for instance, for the prediction of the physical, chemical, or biological properties of systems. More complex models and deep machine learning methods become useful as more data becomes available and more complex problems are experienced. It allows users to make decisions as data are collected, without human-in-the-loop processing [16]. Different type of data can be input to a DL algorithm such as sound, text, images, time series and video. Raman spectroscopy generates time series data such as in resonance Raman and image data such as in Raman image microscope. DL can be applied for machine perception including classification, clustering, and predictions and also a preferred choice for unstructured data like images where manual feature extraction are difficult.

## 2.2 Neural networks

Neural networks (NN) make up the backbone of deep learning algorithms and therefore, it is important to understand common terms in a neural network such as layers, weights and activation functions. **Figure 5(a)** shows a representation of an artificial neuron. The first layers are called input layers which passes incoming data ( $x_1, x_2, x_3, \dots, x_n$ ) into other layers. Output layer is the last layer of neurons that produces given outputs ( $y$ ) for the program.

All layers in between are called hidden layers. Weights ( $w_1, w_2, w_3, \dots, w_n$ ) are the parameters within a neural network that transforms input data within the network's hidden layers. A layer is the highest-level building block in deep learning and is a

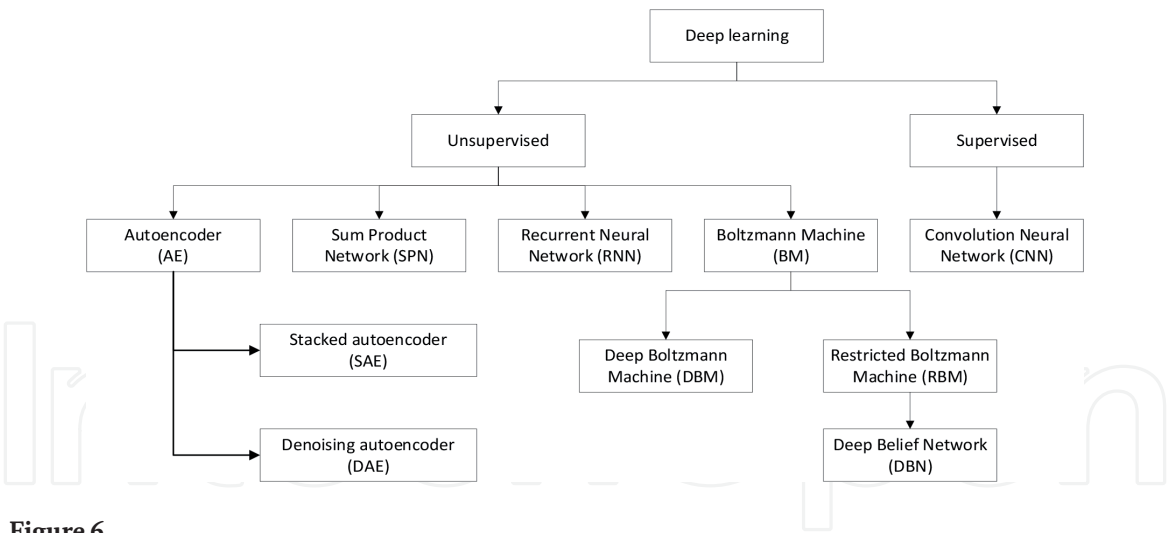


**Figure 5.** Schematic representation of an artificial neuron (a), and a simple neural network displaying three basic elements: Input, hidden and output layers (b), and a deep neural network showing at least two hidden layers, or nodes (c).

container that usually receives weighted input, transforms it with a set of mostly non-linear functions and then passes these values as output to the next layer. An activation function takes in weighted data ( $x_j w_j$  - matrix multiplication between input data and weights) and outputs a non-linear transformation of the data [17]. In generally, an activation function is a function that is added into an artificial neural network to help the network to learn complex patterns in the data. The most important feature in an activation function is its ability to add non-linearity into the network. Activation functions are applied after every layer in deep neural networks and they should be computationally inexpensive to be calculated. Sigmoid, Softmax, Tanh and ReLU are examples for activation function. **Figure 5(b)** shows a simple neural network which has a one hidden layer and **Figure 5(c)** shows a deep neural network which has at least two hidden layers. The neural network calculation is performed through the connections, which contain the input data, the pre-assigned weights, and the paths defined by the activation function. If the result is far from expected, the weights of the connections are recalibrated, and the analysis continues, until the outcome is as accurate as possible. Examples for neural networks are perceptron, feed forward neural network, multilayer perceptron, convolutional neural network, radial basis functional neural network, recurrent neural network, LSTM – long short-term memory, sequence to sequence models, modular neural network [18].

### 2.3 Deep learning algorithms

The objective of this chapter is to give an understanding about the possibilities of deep learnings in the field of Raman spectroscopy. Not many publications



**Figure 6.**  
 Unsupervised and supervised algorithms commonly used in deep learning applications for Raman spectra.

can be found since the connection between Raman data and machine learning is still under the development stage. **Figure 6** shows some algorithms which have been applied for previous Raman data which the reader will find in the rest of the chapter. They are categorized under supervised, unsupervised and hybrid learning methods [19]. In supervised learning algorithms, we try to model relationships and dependencies between the target prediction output and the input features. The goal is to predict the output values for new data based on those relationships which it learned from the previous data sets. Therefore supervised algorithms are task driven. Supervised learning carries out tasks like regression and classification. A very common example for a supervised deep learning method is convolution neural network.

Unsupervised learning is a machine learning technique in which models are not supervised using training dataset. Instead, models itself find the hidden patterns and insights from the given unlabeled data. It can be compared to learning which takes place in the human brain while learning new things. It allows users to perform more complex processing tasks compared to supervised learning and is called a data driven approach. Dimensionality reduction, clustering and association are some tasks than an unsupervised machine learning platform can deliver. The ability to apply deep learning algorithms for unsupervised learning tasks is an important benefit because in big data sets unlabeled data are more abundant than the labeled data. Autoencoder, sum product network, recurrent neural network and Boltzmann machine can be considered as unsupervised deep learning algorithms. Supervised learning algorithms seek to answer the questions like “Based on the Raman fingerprint of this new sample I have just collected, which class in my database does it (most likely) belong to?” and/or “What is the level of purity this substance has?”. Meanwhile unsupervised learning algorithms seek to answer the questions like “How similar to one another are these samples based on their Raman fingerprints?”

### 3. Can deep learning contribute to the development of Raman spectroscopy?

Raman scattering use a technique to interrogate chemical samples in question in a fast and non-destructive way. However, it is a weak scattering and therefore not always give straightforward results. As highlighted in Section 1 three success pillars

of a spectral data analysis depends on “the type of sample to be measured”, “the quality of the spectra” and “the choice of data analysis method”. If any of these pillars fails, the final result will be weak in sensitivity, repeatability and reproducibility. Since we are interested about deep learning methods in this chapter, let's focus on issues related to spectral analysis and merge the contribution of deep learning to overcome those issues. Given below are four challenges that researchers have been experiencing when they analyze Raman fingerprints.

i. Assigning correct vibrational modes

Multicomponent chemical samples can contain vibrational peaks which look similar in shape and distribution over the Raman wavelength region. For instance, biological samples are composed of biochemicals such as lipids, proteins, nucleic acids, and carbohydrates. All the vibrations from these biochemicals are manifested in the Raman spectra of a biological sample making them convoluted and complex. Specially, for a fresh researcher, these spectra may appear very similar if analyzed by an untrained eye. Researchers working with Raman spectra of cells, tissues and bacteria also encounter the same problem. There are also incidents that different Raman spectrometers exhibit a small magnitude of change of Raman shift for the same component. This change can also be a significant problem if the spectra is crowded with several closely packed peaks.

ii. Analyte is influenced by the background

Weak Raman-active samples can be only analyzed if there is high spectral resolution, low spectral background, and high sensitivity. The relative intensities of the Raman bands of analytes change with solvents and are correlated with the absorption peak shift [20]. Occurrence of peaks from the matrix is true in many biomolecular Raman applications. For instance, paraffin fixed tissue may show a similar peak to a C–H stretch. Differentiating the actual spectra from the matrix therefore, becomes an equally important part before analysis.

iii. Fluorescence problem

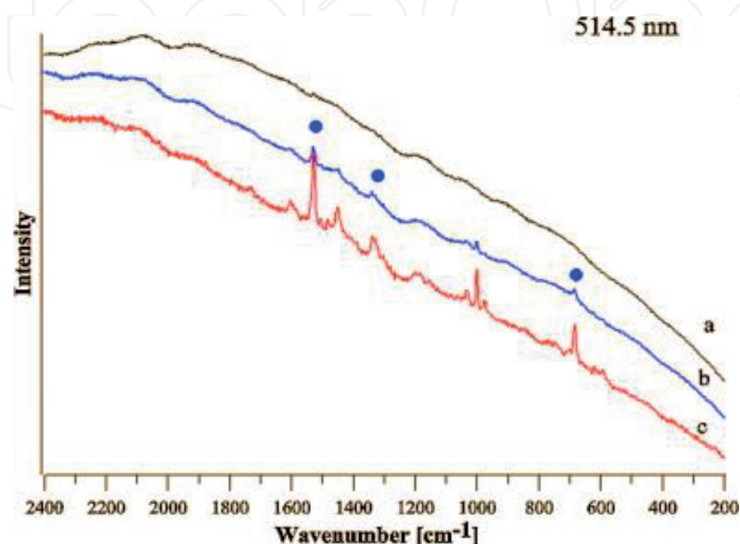
One of the greatest challenge in Raman spectroscopy is that it is influenced by the turbidity, color, and fluorescence of the sample [21]. In spite of obvious advantages of Raman spectroscopy, the strong fluorescence background has so far restricted its use in many otherwise potential applications, for example, in the agricultural, food and oil industries, security control and crime investigations, for example. Marquardt [22] mentions that Raman biotech applications are currently is the most challenging because of the complex biological matrices and the associated fluorescence. Raman spectra are typically masked by a strong fluorescence background in most potential applications. This type of fluorescence intensity is normally several orders of magnitude larger than the Raman scattering signal, especially in biological samples. This is due to the fact that the probability of Raman scattering (cross-section) is much lower than that of fluorescence [23]. A strong fluorescence background gives rise to two problems. Firstly, it becomes the dominant element in the photon shot noise and thus detracts from the SNR (signal-to-noise ratio), and secondly, even if the Raman bands are narrow and the fluorescence has quite a smooth, featureless spectrum,

errors in the mathematical estimation and removal (background subtraction) of the fluorescence increase with increasing fluorescence levels and result in increasing errors in both material identification and concentration measurement applications [23]. Fluorescence can be dealt with a variety of techniques such as the utilization of confocal configuration, photobleaching and the deployment of laser excitation at longer wavelengths. These techniques could be generally grouped into time domain, frequency-domain, wavelength-domain, and computational methods [24]. **Figure 7** shows three Raman spectra obtained from 514.5 nm laser, where the fluorescence effect of the original spectrum, *a*, could be reduced by increasing the irradiation effect as shown by spectra *b* and *c*. However, a key constraint to consider is the interaction between the laser wavelength and the sample, due to the effects of phototoxicity [25].

Computational methods can play a significant role for unmolding chemical Raman spectra from fluorescence spectra. Examples for such methods are polynomial fitting wavelet transform, and derivatives. Wei, Chen [24] describe pros and cons of polynomial fitting and derivative of Raman spectra. They mention that the optimal choice of order for polynomial fitting varies and the performance depends on the user's experience. The derivative of a measured Raman spectrum will eliminate the background components irrespective of their magnitudes and thus enhance the sharp Raman signal. However, high-frequency noises are often amplified by this method as well and the spectrum can be distorted because of the derivative process.

#### iv. Selection of optimum data processing technique

Understanding the system under study and making an informed judgment based on the experiments and correlating it with the available data is crucial for scientists. Selecting the correct signal processing method is a contributing factor towards understanding of the system. Improving the existing data analysis methods in Raman spectroscopy is a leading challenge. Preprocessing methods are very important to reduce inherent disturbances of a Raman spectrum such as baseline variation. Currently spectroscopists are limited to traditional chemometrics based preprocessing methods.



**Figure 7.** Three Raman spectra of metallic paint showing the effect of fluorescence [7] (*a*) in normal measurement conditions (only fluorescence visible), (*b*) after irradiation for (*b*) 30 min and (*c*) 60 min at 50% laser power.

Models are calibrated using a fewer number of datasets, even in the situations where it is possible to use fairly a large calibration dataset. When the models are used for future large data sets these methods are limited in accuracy wise. For example, the instrument gives poor results when unknown interferences come with larger datasets such as spikes, cosmic rays and often require for re-calibration of the models time to time.

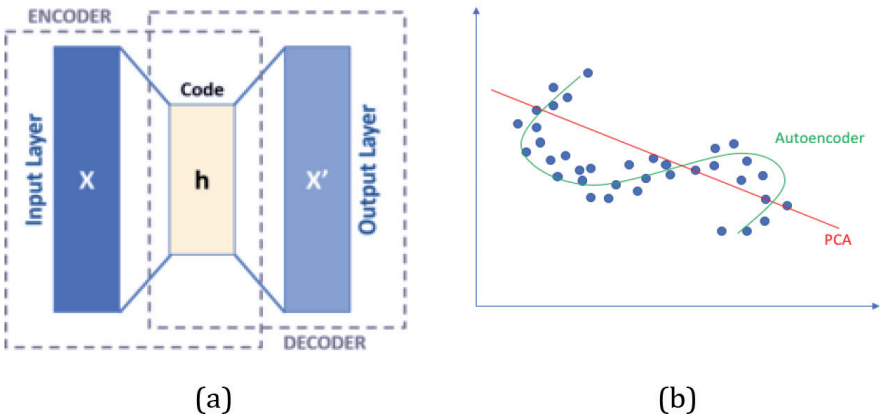
4. Deep learning algorithms in Raman applications

In this section, four deep learning algorithms and their derivations for different applications of Raman spectroscopy are described to provide an understanding of deployment of these methods as a means of strengthening computational methods and data analysis methods for Raman spectra.

4.1 Autoencoder

An autoencoder (AE) is an unsupervised type of artificial neural network used to learn efficient data coding. It consists of an encoder-decoder architecture as shown in **Figure 8**. Encoder consists of input data  $x$  while decoder includes output data  $x'$ .  $h$  usually referred to as code, latent variables, or latent representation and combines encoder and decoder. The aim of an autoencoder is to learn a representation (encoding) for a set of data, by training the network to ignore signal noise. Along with the reduction side of encoder, a reconstructing side is learned, where the autoencoder tries to generate from the reduced encoding a representation as close as possible to its original input  $x$ . This is done by training the AE to minimize the squared reconstruction errors  $|x - x'|^2$ . PCA is a linear transformation while auto-encoders are capable of modeling complex non linear functions (refer **Figure 8(b)**). PCA is faster and computationally cheaper than autoencoders. A single layered autoencoder with a linear activation function is very similar to PCA. The autoencoder weights are not equal to the principal components, and are generally not orthogonal, yet the principal components may be recovered from them using the singular value decomposition [26].

Advantages of autoencoder in Raman spectroscopy span in different areas such as dimensionality reduction, information retrieval, image processing and anomaly detection. Scientists have experimented several kinds of autoencoders such as convolution AE, denoising AE, sparse AE because they have different advantages. For



**Figure 8.** Explanation of (a) autoencoder (AE) system and (b) simple demonstration of linear dimensionality reduction by PCA and non-linear dimensionality reduction by AE.

instance, sparse AE prevents overfitting. Convolutional AE is generally applied in the task of image reconstruction. If the network is trained on corrupted versions of the inputs with the goal of improving the robustness to noise, it is called a denoising autoencoder [27].

#### *4.1.1 Anomaly detection without actually testing samples using an autoencoder network*

Anomaly (outlier) detection has been an important research topic in data mining and machine learning while it also provides practical benefits in many real-world applications. Outlier detection has been used in spectroscopic data to detect and remove anomalous observations (if required). Most of the process analytical instruments implemented in industrial plants can also be converted to perform outlier detection in addition to their main task; for instance to detect a fault on a factory production line by constantly monitoring specific features of the products and comparing the real-time data with either the features of normal products or those for faults. Outliers arise due to mechanical faults, changes in system behavior, fraudulent behavior, human error, instrument error or simply through natural deviations in populations [28]. Modeling anomalies are not easy in real datasets as they appear irregularly and not often. Since abnormal data points appear rarely it is very costly to collect those data from real world [29]. Hodge and Austin [28] shows a survey on different techniques for outlier detection in machine learning. They highlighted that correct distribution model, correct attribute types, scalability, speed, any incremental capabilities to allow new exemplars to be stored and the modeling accuracy must be considered when selecting a suitable algorithm for outlier detection. In machine learning, multiclass or multinomial classification is the problem of classifying instances into one of three or more classes (classifying instances into one of two classes is called binary classification). In one-class classification which is also referred as class-modeling, whether a sample is compatible or not with the characteristics of a single class of interest is considered. The study by Hofer-Schmitz [30], presents an one-class anomaly detector based on autoencoder for Raman spectra for a biological application, where it's very costly to collect spectra of the outlier class. They use two chemical data sets with 10,000 samples and over 2000 samples for their evaluation. Bio-chemical approach to identify and characterize outliers takes months and therefore they measured normal class and trained one-class model using Autoencoder network to learn the normal classes' characteristics by minimizing the reconstruction error (score) with respect to the given loss function, similar to the learnt components of PCA. When using the learnt encodings to reconstruct irregular spectra, a sample's reconstruction was considered as anomaly if it exceeds a standard deviation threshold.

#### *4.1.2 Sample classification using an autoencoder network*

Houston [31] used six classification algorithms to identify whether a set of chemical samples contain chlorinated solvents or not, based on their Raman spectra. Dataset included 230 Raman spectra of solvents and solvent mixtures. An additional dataset comprising 24 Raman spectra of carbohydrates was compiled for use as examples of possible outlier data. k-Nearest Neighbors (kNN), Support Vector Machine, Decision Tree, Fully Connected Neural Network (FCNN), Gaussian Naïve Bayes, Locally Connected Neural Network (LCNN) were the algorithms used. The ability of the autoencoder models to correctly identify negative outliers were further demonstrated. Their results showed that a two-step process, combining an outlier detector and LCNN binary classifier, have better performance.

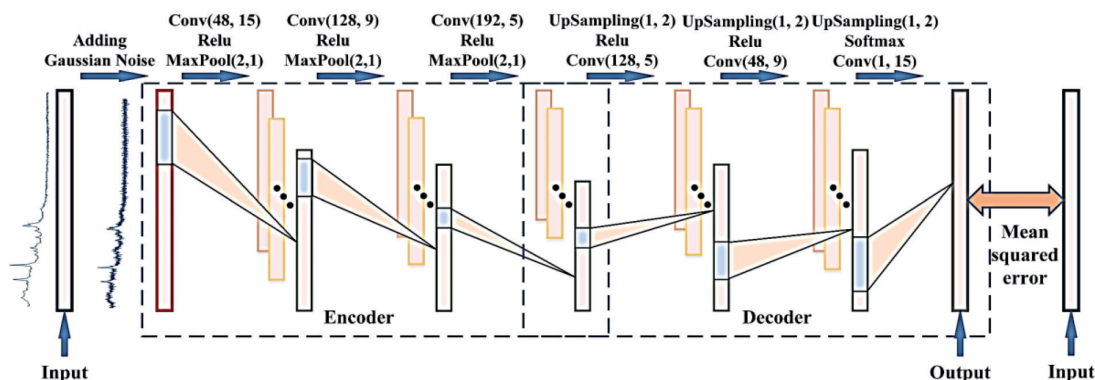
LCNN is quite the same as the Convolutional layer explained in Section 4.2. But has one (important) difference. In LCNN, there is a locally connected layer going from the inputs to the first hidden layer. In the Convolutional layer the filter is common among all output neurons. In Locally-Connected Layer, each neuron has its own filter. This type of layer let the network to learn different types of feature for different regions of the input, but if there is less number of data, it can also generate over-fitting.

#### 4.1.3 Increasing signal-to-noise ratio (SNR) by convolutional denoising autoencoder (CDAE)

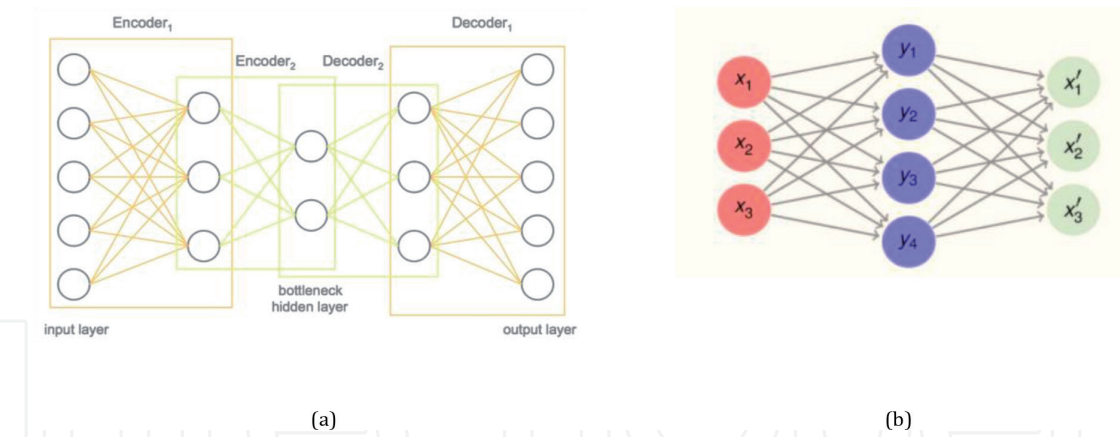
Obtaining the highest possible SNR and a good enough spectral resolution for a specific analysis are important factors while using Raman spectroscopy. The light of the Raman signal is refocused on a charge-coupled device (CCD) after dispersion by a diffraction grating which inevitably lower the signal. To obtain better Raman signals, generally, the excitation intensity is increased. However, this is not always practicable if the sample is sensitive to higher laser power. Physical and chemical properties of sensitive samples can be degraded by exposing to higher laser power. Therefore, in experimenting such, laser exposure times are extended while keeping a lower excitation intensity. As a result, stray light, environmental light, and the inherent interior noise of electronic or optical devices [13] result in noise adding up over longer integration time. These factors influence signal-to-noise ratio (SNR), thus further affecting the feature extraction of the valid signal. Fan, et al. [32] proposes a relevant automatic denoising method of convolutional denoising autoencoder (CDAE) to advance the SNR in Raman spectra without manual intervention. **Figure 9** shows the CDAE model which includes three layers of convolution and max-pooling (the encoder) and three layers of upsampling and convolution (the decoder) proposed by the authors Fan, Zeng [32]. The proposed CDAE model was implemented using Keras and Tensorflow. The authors show that the CDAE method outperforms other classical denoising methods such as Savitzky–Golay filter and wavelet transform.

#### 4.1.4 Stacked sparse autoencoder (SSAE) to extract features from the unlabeled Raman data

Sparse autoencoder (SAE) may include more (rather than fewer) hidden units than inputs, but only a small number of the hidden units are allowed to be active at the same time. This sparsity constraint forces the model to respond to the unique statistical features of the training data. Sparse feature learning



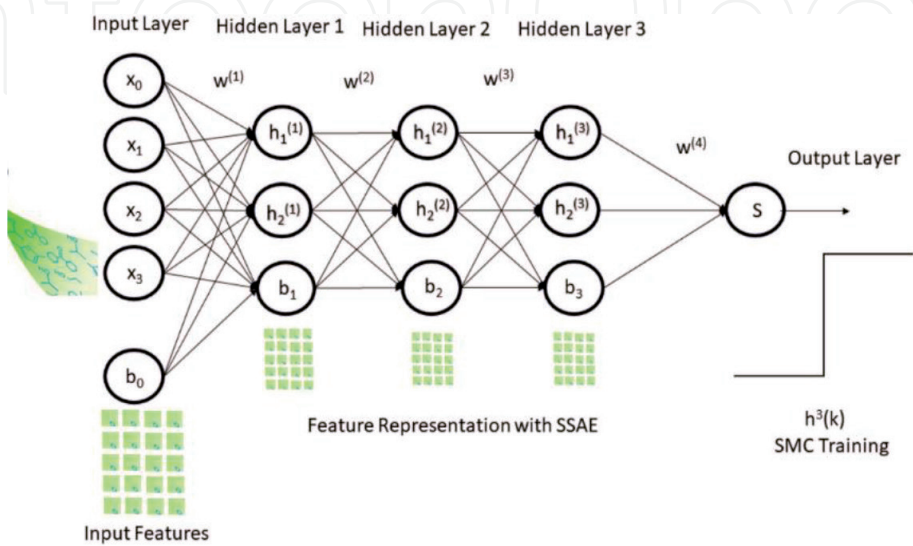
**Figure 9.**  
The architecture of CDAE used to increase SNR in Raman spectra by fan, Zeng [32].



**Figure 10.**  
Representation of (a) stacked autoencoder and (b) sparse autoencoder.

algorithms range from sparse coding approaches [33] to training neural networks with sparsity penalties. In the SAE, once the training process is performed, the decoder and reconstruction layer will be removed, and the features learned from the original data are preserved in the hidden layer. To extract high-level features, a stacked SAE (SSAE) is utilized. The SSAE consists of several SAEs, with the output of the previous SAE used as the input of a subsequent SAE (**Figure 10**) [34].

Feature extraction using a stacked sparse autoencoder integrated with a Softmax classifier (SMC) to extract the discriminative features from unlabeled Raman data of breath samples is proposed by Aslam [35]. They were successful to identify fifty peaks in each spectrum to distinguish the patients with gastric cancer and healthy persons. The architecture of this neural network comprises of two sparse auto-encoder layers and the output of the stacked sparse autoencoder was wired into a Softmax layer as shown in **Figure 11**. This system reduces the distance between the input and output by learning the features and preserve the structure of the input data set of breath samples. The proposed deep stacked sparse autoencoder neural network architecture exhibits excellent results, with an overall accuracy of 98.7% for advanced gastric cancer classification and 97.3% for early gastric cancer detection using breath analysis.



**Figure 11.**  
Architecture of the stacked sparse autoencoder with a Softmax classifier [35].

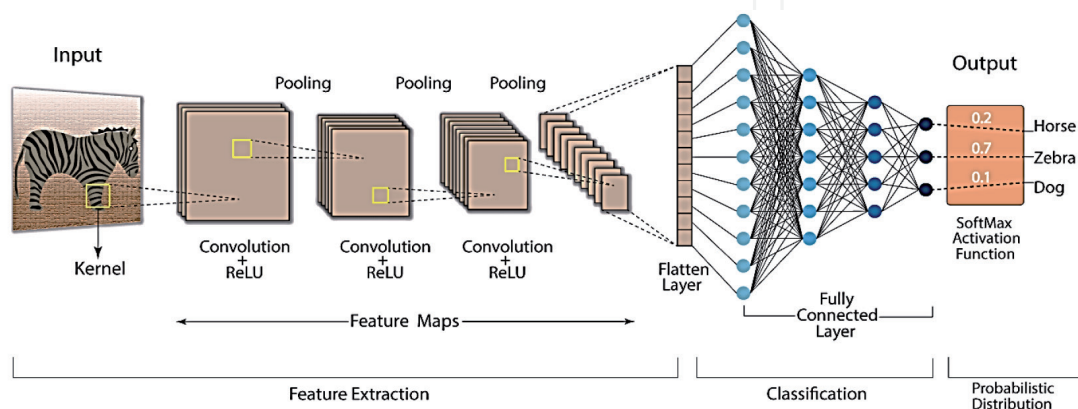
## 4.2 Convolution neural network (CNN)

Several studies show that the convolution neural network (CNN) modeling method is potential to be used for spectral analysis. With the development of *deep learning*, CNN has become a major tool specially for image analysis. Independency from prior knowledge and human extraction is a feature of machine learning and CNN is one of best examples for that. In traditional algorithms, hand-engineered filters are used in preprocessing to understand the learning process, whereas in CNN, filters (or usually known as kernels) automatically learn the optimization without human intervention.

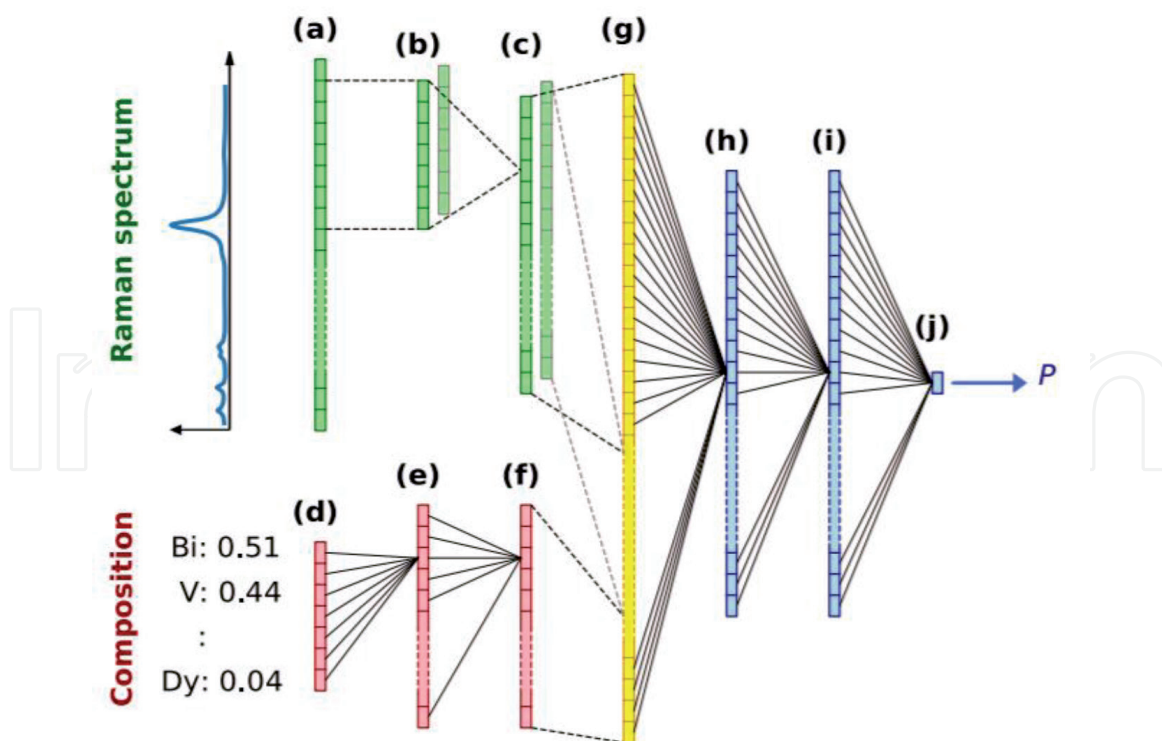
A typical CNN includes *convolutional*, *pooling* and *fully connected layers* as shown in **Figure 12**. Convolution is the process involving combination of two functions that produces the other function as a result. The convolution layers improve the performance of the network by shared weights and sparse connection. Pooling layer operates on each feature map independently. This reduces resolution of the feature map by reducing height and width of features maps, but retains features of the map required for classification. This is called Down-sampling. The output feature map(matrix) from pooling layer is converted into vector (one dimensional array) which is called flatten layer. Fully connected layer looks like a regular neural network. Soft-max is an activation layer normally applied to the last layer of network that acts as a classifier and it is used to map the non-normalized output of a network to a probability distribution.

### 4.2.1 CNN for predicting material properties and understanding composition-structure-property relationships

A CNN model was constructed by Umehara, et al. [37] in python using Keras package with Tensorflow backend to identify composition-property and composition-structure-property relationships that lead to fundamental materials insights through Raman spectra. They developed a model that could predict photoelectrochemical power density ( $P$ ) of a solar fuels photoanode from materials parameters which were their composition and Raman signals of 1379 samples. **Figure 13** shows the CNN model structure used for this study where they used 10 different layers to derive model predictions. In addition to the prediction of material property, they also analyzed gradients in the trained model which was useful to reveal key data relationships that were not readily identified by human inspection or traditional statistical analyses. For instance, what is the impact of performance of the



**Figure 12.**  
Example showing how CNN works [36].



**Figure 13.** Schematic of CNN model structure used by Umehara, Stein [37]. The model takes the Raman spectrum and the composition –(Bi, V and Dy) as input to predict  $P$  = power density. Each of the 10 layers of the CNN model are labeled a to j. red: Dense layers acting on composition, green: Convolutional 1D layers acting on spectra, yellow: Flattening and concatenation layers, blue: Dense layers acting on both the composition and spectral data.

photoanode by changing the concentrations of alloying elements? Authors have also highlighted that the human interpretation of these key relationships produces the desired fundamental understanding, demonstrating a framework in which machine learning accelerates data interpretation by leveraging the expertise of the human scientist.

#### 4.2.2 Identification of chemical species by CNN without preprocessing

Liu, et al. [38] describes an unified solution for the identification of chemical species. They used a trained convolutional neural network to automatically identify substances according to their Raman spectrum without preprocessing. Most of the Raman based regression procedures demand for preprocessing such as cosmic ray removal, smoothing and baseline correction. CNN combines preprocessing, feature extraction and classification in a single architecture which can be trained end-to-end with no manual tuning [38].

They evaluated their approach using the RRUFF spectral database [39], comprising mineral sample data and a superior classification performance was demonstrated compared with other frequently used *machine learning* algorithms including SVM, kNN, gradient boosting, CNN and random forest. In addition, 512 raw minerals spectra from the RRUFF database were used together with six widely-used baseline correction methods: modified polynomial fitting, rubber band, robust local regression estimation, iterative restricted least squares, asymmetric least squares smoothing and rolling ball. Results are shown in **Table 1** which indicates that the raw spectra on CNN achieved an accuracy of 93.3% and the second best method, kNN with rubber band baseline correction, achieved an accuracy of 82.5%.

Method	kNN (k = 1)	Gradient boosting	Random forest	SVM (linear)	SVM (Radial basis function)	CNN
Raw	0.429 ± 0.011	0.373 ± 0.019	0.394 ± 0.016	0.522 ± 0.011	0.434 ± 0.012	0.933 ± 0.007
Assymmetric least squares	0.817 ± 0.010	0.773 ± 0.009	0.731 ± 0.019	0.821 ± 0.012	0.629 ± 0.016	0.927 ± 0.008
Modified polynomial	0.778 ± 0.007	0.740 ± 0.016	0.650 ± 0.016	0.785 ± 0.014	0.629 ± 0.016	0.920 ± 0.008
Rolling ball	0.775 ± 0.009	0.737 ± 0.008	0.689 ± 0.018	0.795 ± 0.011	0.624 ± 0.013	0.918 ± 0.008
Rubber band	0.825 ± 0.007	0.792 ± 0.015	0.741 ± 0.009	0.806 ± 0.015	0.620 ± 0.010	0.911 ± 0.008
IRLS	0.772 ± 0.010	0.710 ± 0.008	0.675 ± 0.007	0.781 ± 0.011	0.614 ± 0.010	0.911 ± 0.008
Robust local regression	0.741 ± 0.009	0.694 ± 0.008	0.667 ± 0.0012	0.759 ± 0.013	0.600 ± 0.013	0.909 ± 0.007

**Table 1.**  
*Performance of different ML algorithms with and without baseline correction methods after applied on RRUFF Raman spectral data (of 512 minerals).*

#### 4.2.3 Tuning preprocessing of Raman spectra in one step by training a CNN model using simulated data

Wahl, et al. [40] show that a convolutional neural network can be trained using simulated data to handle several preprocessing steps automatically in a single step. These preprocessing methods include cosmic ray removal, signal smoothing, and baseline subtraction. Synthetic spectra were created by randomly adding peaks, baseline, mixing of peaks and baseline with background noise, and cosmic rays. Secondly, a CNN was trained on synthetic spectra and known peaks. Finally, a test set data which consisted of real Raman spectra of polyethylene, paraffin, and ethanol were used to evaluate the trained CNN model. The samples were placed on a polystyrene petri dish and their Raman measurements were taken so that the signals from the samples were mixed with signal from polystyrene. Measurements which only contained one cosmic ray were saved for the analysis. The performance of the CNN model was estimated by calculating the root mean squared error (RMSE). From 105 simulated observations, 91.4% predictions had smaller absolute error (RMSE). Authors also recommend that the similar simulation scheme for adaptations to problems with similar preprocessing challenges such as NIR, FT-IR, mass spectroscopy, and chromatograph and also take the benefit of the reduced computational time and time spent by an analyst in preparing data for the analysis. CNN preprocessing generated reliable results on measured Raman spectra from polyethylene, paraffin, and ethanol with background contamination from polystyrene.

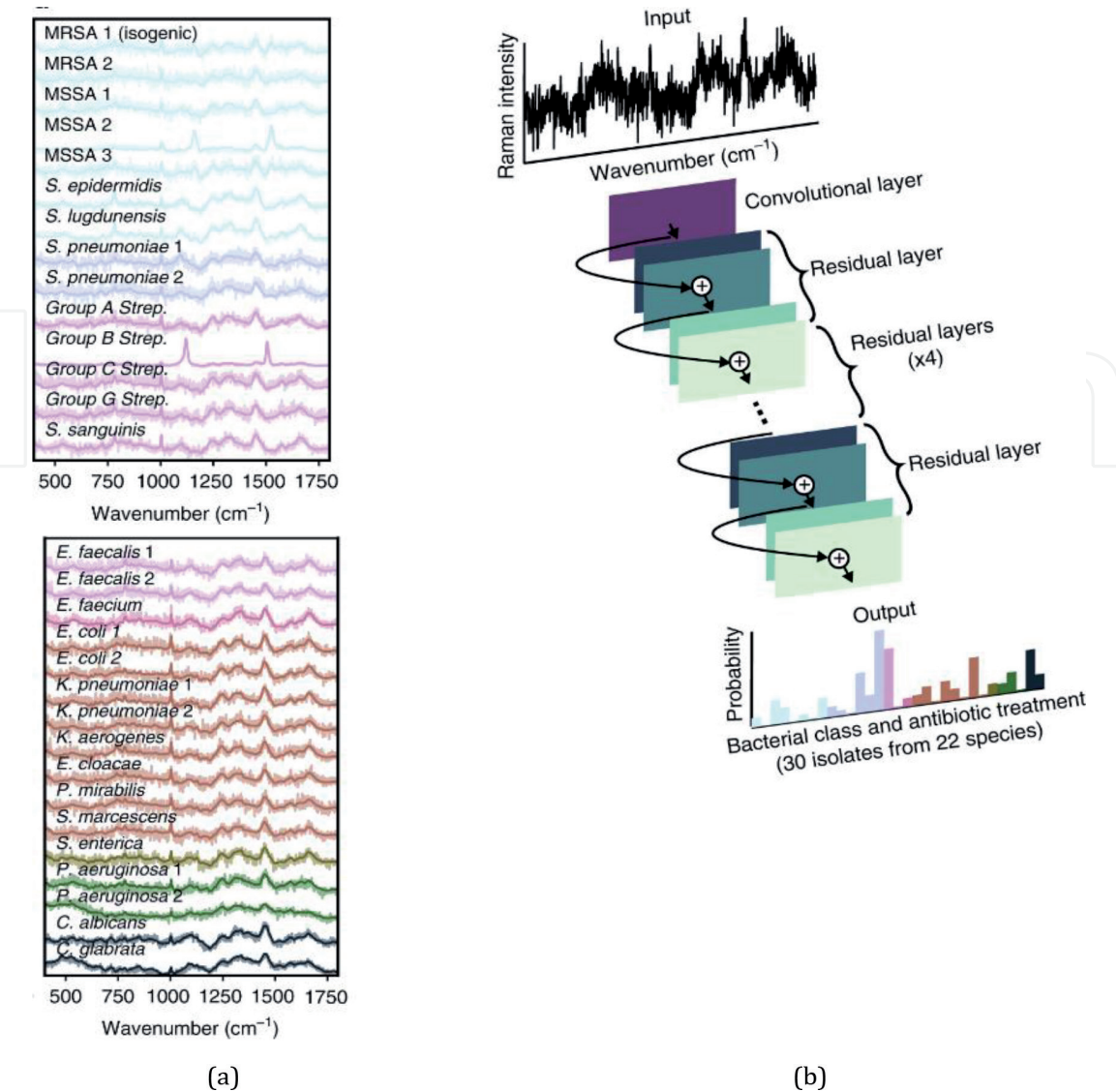
#### 4.2.4 CNN for bacterial detection, identification, and antibiotic susceptibility testing in a single step

Different bacterial phenotypes are characterized by unique molecular compositions. However they only lead to subtle differences in their corresponding Raman spectra. And due to the weak Raman scattering these subtle spectral differences are easily masked by background noise. Maintaining a higher signal-to-noise ratio by increasing the measurement time are often restricted in these types of samples. This challenge has been addressed by [41] using a trained convolutional neural network which can classify noisy bacterial spectra by using a very low measurement time of 1 second. The reference samples including bacterial and yeast isolates which generated 2000 spectra from a Raman microscope. Spectra were background corrected using a polynomial fit of order 5.

**Figure 14** shows (a) spectral variation of Raman bacterial spectra and (b) the CNN architecture. CNN architecture used by these researchers consisted of an initial convolution layer followed by 6 residual layers and a final fully connected classification layer. Each residual layer contains 4 convolutional layers, and therefore the total depth of the network was 26 layers. The initial convolution layer has 64 convolutional filters, while each of the hidden layers has 100 filters. An identification accuracies of 99.7% was achieved by the researchers in this study when they validated the method using clinical samples.

### 4.3 PCANet

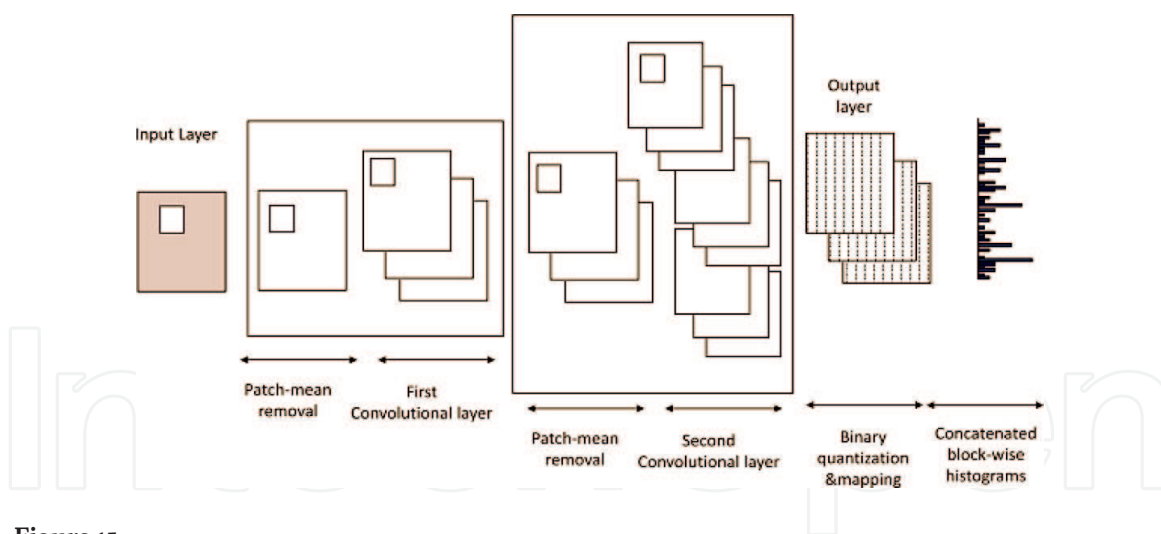
The principal component analysis network (PCANet), which is one of the recently proposed deep learning architectures, achieves the state-of-the-art classification accuracy in various databases [42]. It is also known as one of the simplest deep learning algorithms and can be adapted to small-scale data [43]. In the section below, application of PCANet deep learning for Raman spectroscopy is reviewed



**Figure 14.** (a) Averages of 2000 spectra from 30 isolates in bold and overlaid on representative examples of noisy single spectra for each isolate. Raman spectra of bacterial species can be difficult to distinguish, and short integration times (1 s) lead to noisy spectra (SNR = 4.1). Spectra are color-grouped according to antibiotic treatment (b) low-signal Raman spectra are classified as one of 30 isolates, which are then grouped by empiric antibiotic treatment and one-dimensional residual network with 25 total convolutional layers [41].

using some of prominent research studies. Architecture of the PCANet is shown in **Figure 15**. It typically consists with only two convolutional layers.

The main algorithm used to learn the convolutional filters in PCANet is principal component analysis (PCA) algorithm. PCA is a linear transformation method which transforms original data to a new orthogonal coordinate system with less dimensionality. Eigenvalues and eigenvectors are calculated from the covariance matrix of the original dataset. Eigenvectors which have the highest eigenvalues are always selected while discarding that of small values. In the convolutional layer of PCANet, all local patches are convolved with the selected eigenvectors to create a new set of data which focus on the most important features of the input data. The main flow of PCANet can be divided into three stages. The function of the first two stages is similar, and the principal eigenvector of input matrix is obtained through the cascaded multiple-PCA filter in these two stages. In the last stage, the principal eigenvectors are performed by binary hash encoding and then processed to the composed block-wise histogram. Afterward, the histogram is combined with the classification algorithms to obtain the predicted data.



**Figure 15.**  
 PCANet architecture [44].

#### 4.3.1 Recognition and quantitation of drugs in human urine by PCANet

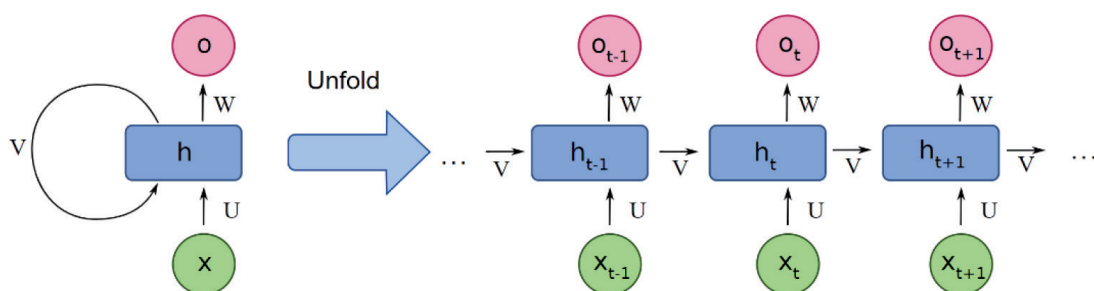
Weng, et al. [45] shows that *deep learning* networks perform better than the common *machine learning* methods (including kNN, SVM, RF, LR, and PLS) and provide feasible alternatives for the recognition and quantitation of SERS. In their study, *deep learning* networks were used as fully connected networks, convolutional neural networks (CNN), fully convolutional networks (FCN), and principal component analysis networks (PCANet) to determine their abilities to recognize drugs in human urine and measure pirimiphos-methyl in wheat extract in the two input forms of a one-dimensional vector or a two-dimensional matrix.

#### 4.3.2 Rapid detection of impurities using PCANet

Surface-enhanced Raman spectroscopy (SERS) has affected many areas in analytical detection, surface property investigation, biological event and marker sensing and imaging, and environment monitoring and its application in analytical science, food science, environmental sciences and biomedical sciences is enormous [46]. The study by Weng, et al. [47] proposes the suitability of SERS over NIR and FTIR for the automatic analysis of hazardous pesticide residues (acephate) in rice due to the significant interference from the aqueous phase. They used 82 contaminated rice samples for the model development and 14 contaminated rice samples were randomly selected as the prediction set. Finally, they combined the modeling methods in PCANet with the regression algorithms as PLSR, SVM, or RF (PCANet<sub>PLSR</sub>, PCANet<sub>RF</sub>, and PCANet<sub>SVM</sub>) to obtain the residue level.

### 4.4 Recurrent neural network (RNN)

Recurrent Neural Network (RNN) is a tool in deep learning for problems that deal with sequential data [48]. Although, RNN was firstly designed to deal with sequential information, today it shows applications in time series data, natural language and converting non-sequencing data like images to sequences. The most used recurrent units are long short-term memory (LSTM) and gated recurrent unit (GRU). LSTM is a deep learning system that avoids the vanishing gradient problems in RNN [49]. The GRU is like a LSTM but it has fewer parameters than an LSTM [50]. Some results indicate that GRUs can outperform LSTMs while others show



**Figure 16.**

Architecture of RNN;  $x$ : The input,  $O$ : The output,  $h$ : The main block of the RNN. It contains the weights and the activation functions of the network,  $V$ : Represents the communication from one time-step to the other [51].

the opposite results. The RNN models are trained with back propagation through time (BPTT) method. There are variants of RNN such as bidirectional RNN and deep RNN.

Possibility of processing input of any length, model size which is not affected with size of input, computational ability which takes into account historical information and weights which are shared across time which makes an efficient data handling are the advantages of RNN. On the other hand, it also has the drawback of having a slower computation, difficulty of accessing information from a long time ago and inability to consider any future input for the current state (**Figure 16**).

#### 4.4.1 Species identification and model transfer using RNN

Species identification of human and animal blood is of critical importance in the areas of custom inspection, forensic science, wildlife preservation, and veterinary purpose. High-performance liquid chromatography (HPLC), mass spectroscopy (MS), nuclear magnetic resonance (NMR), polymerase chain reaction (PCR) are DNA profiling suitable methods, but they require experienced experts and professional laboratory. FTIR is also a promising candidate for this purpose but the presence in water makes the spectral analysis is challenging. Considering the interference of water and the risk of contact of pathogen, Wang, et al. [52], used Renishaw inVia confocal Raman spectrometer and a laboratory-built Raman spectrometer to find a method to discriminate of 20 kinds of blood species including human, poultry, wildlife, and experimental animals. The Raman spectra pre-processing methods included cosmic ray removal, Savitzky–Golay filter, baseline removal, normalization and standardization. The processed spectra were randomly grouped into training dataset (80%), validation dataset (10%) and testing dataset (10%). Data was input to different deep learning models such as RNN, GRU, LSTM and CNN and performance was compared. This study also proposes a solution for the wavenumber drift during long term use of instruments. Analyzing the blood samples are affected by the wavenumber drift and therefore instruments are required for immediate calibration. The usual RNN model could not function well for these unexpected drifts and therefore augmented Raman spectra with certain wavenumber drift were included intentionally in this study. Another speciality of this study is the migration learning of model transfer between Raman spectrometers with different performance. This was achieved by training a cross-instrument RNN model with spectra from 2 Raman spectrometers (1463 spectra from Renishaw Raman spectrometer and 1621 spectra from laboratory-built Raman spectrometer), which could be used for identification of blood species. This combined model showed accuracy is 98.2%.

#### 4.4.2 Gated recurrent unit coupled with MCNN

The study [53] proposes the use of a gated recurrent unit (GRU) and multiscale fusion convolutional neural network (GRU-MCNN) to analyze Raman spectra of patients infected with hepatitis B virus (HBV). Current commonly used method for the detection of HBV is polymerase chain reaction, but the shortcomings of this method such as the possibility of cross-contamination of samples during the analysis which can generate false results and using a carcinogenic dying agent for the sample preparation can be eliminated using Raman spectroscopy non-invasive analysis. Unlike traditional methods for extracting spatial features, the MCNN first transforms the original data sets into a pyramid structure containing spatial information at multiple scales, and then automatically extracts high-level spatial features using multiscale training data sets [54]. GRU-MCNN model developed by [53] showed accuracy, precision, sensitivity and specificity over 0.97 for unprocessed data and it is even a higher value that was recorded for processed data.

#### 4.5 Performing a deep learning analysis for data

Various deep learning tools are available in the market today, such as Neural Designer, H2O.ai, DeepLearningKit, Microsoft Cognitive Toolkit, Keras, ConvNetJS, Torch, Gensim, Deeplearning4j, Apache SINGA, Caffe, Theano, ND4J, and MXNet. Which one is the best, depends on the user and application. Many of these machine learning algorithms are available as free software modules and/or libraries for programming environments like Python, R, C++ and C# [55–58] to mention some. In the python programming environment, Keras and TensorFlow modules are popular for deep learning. Microsoft has the free ML.NET machine learning environment that is supported using the Visual Studio tools. Matlab and Python are widely used in academics and use a GUI interface enabling ML without writing the code by the user, however, some programming skills are needed. If the user wants to explore ML in depth and write his own code from scratch, R is often preferred, but there is really no agreed consensus on this matter. Python is a programming language which consists of a large standard library. One major advantage with Python is that it is free. Matlab is most highly regarded as not only a commercial numerical computing environment, but also as a programming language. Matlab has many functions for data processing and plotting. It also contains toolboxes such as Deep Learning toolbox. Toolboxes in Matlab usually comes with added cost. R is free, open-source software designed to run statistical analyses and output graphics.

### 5. Conclusion

Most common procedure employed in spectroscopic data analysis is selecting proper tools, validating them, and highlighting their use in real-world applications by a series of examples. Getting inspiration by the field of computer vision will surely accelerate the development of more robust methods in this process. The next generation of Raman data analysis will be using more advanced algorithms to further improve the analytical performance of spectral classification, regression, clustering, and rule mining. In supplementary, it will also be the key factor to break the limitations of Raman spectroscopic applications.

For instance, literature shows that molecular spectra predictions can be made instantly using deep learning at no further cost for the end user. Spectra with

outliers are synthetically implemented and solved using autoencoders when such irregular spectra are costly or time consuming to obtain in reality. Scientists show that several Raman preprocessing steps can be performed using a single step by convolutional neural networks while in traditionally, combinations of preprocessing methods are performed as iterations to select the optimum preprocessing which demand time. Some DL algorithms show promising results by using raw spectra in entire wavelength as input for regression models region replacing monotonous variable selection methods. Classification problems in SERS and Raman spectra, have received the advantages of general image recognition deep learning methods which significantly improve selectivity and specificity over conventional classification methods. Unlabeled large Raman datasets which have been collected over years in clinical applications have been using to diagnose other diseases in addition for their main purpose where accuracy of the data interpretation are improved as dataset is being updated and heavier.

The classical linear methods of processing the extracted information from challenging Raman and SERS experiments no longer suffice. Deep learning is shaping up machine learning algorithms in many ways through carefully analyzing patterns and aberrations in those patterns. In analytical sciences, machine learning provides an unprecedented opportunity to extract information from complex datasets. Very often, the unfamiliarity of machine learning algorithms and definitions which is normally in the computer science domain, dictates the unpopularity of using them as tools in chemistry and analytical science. This chapter is aimed to elaborate the potential of deep learning methods with respect to its suitability in Raman spectral analysis. As these methods are applicable to other types of spectroscopies deep learning and artificial intelligence data processing in spectroscopy is bound to grow in the near future.

## Acknowledgements

Authors would like to acknowledge the financial assistance from Faculty of Technology, University of South-Eastern Norway.

## Conflict of interest


The authors declare no conflict of interest.

## Author details

M.H. Wathsala N. Jinadasa\*, Amila C. Kahawalage, Maths Halstensen, Nils-Olav Skeie and Klaus-Joachim Jens  
Faculty of Technology, Natural Sciences and Maritime Sciences, University of South-Eastern Norway, Porsgrunn, Norway

\*Address all correspondence to: wathsala.jinadasa@usn.no

## IntechOpen

© 2021 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

## References

- [1] Ferraro, J.R., K. Nakamoto, and C.W. Brown, *Chapter 1 - Basic Theory*, in *Introductory Raman Spectroscopy (Second Edition)*, J.R. Ferraro, K. Nakamoto, and C.W. Brown, Editors. 2003, Academic Press: San Diego. p. 1-94.
- [2] Smekal, A., *Zur Quantentheorie der Dispersion*. Naturwissenschaften, 1923. **11**(43): p. 873-875.
- [3] Raman, C.V. and K.S. Krishnan, *A New Type of Secondary Radiation*. Nature, 1928. **121**(3048): p. 501-502.
- [4] Das, R.S., Agrawal, Y. K., *Raman spectroscopy: Recent advancements, techniques and applications*. Vibrational Spectroscopy, 2011. **57**(2): p. 163-176.
- [5] Ferraro, J.R., K. Nakamoto, and C.W. Brown, *Chapter 3 - Special Techniques*, in *Introductory Raman Spectroscopy (Second Edition)*, J.R. Ferraro, K. Nakamoto, and C.W. Brown, Editors. 2003, Academic Press: San Diego. p. 147-206.
- [6] Cozzolino, D., *The Sample, the Spectra and the Maths—The Critical Pillars in the Development of Robust and Sound Applications of Vibrational Spectroscopy*. Molecules, 2020. **25**(16): p. 3674.
- [7] Mendez, K.M., D.I. Broadhurst, and S.N. Reinke, *The application of artificial neural networks in metabolomics: a historical perspective*. Metabolomics, 2019. **15**(11): p. 142.
- [8] Vogt, F., *Quo vadis, chemometrics?* Journal of Chemometrics, 2014. **28**(11): p. 785-788.
- [9] Kowalski, B., S. Brown, and B. Vandeginste, *Editorial*. Journal of Chemometrics, 1987. **1**(1): p. 1-2.
- [10] HÉBerger, K., *Chapter 7 - Chemoinformatics—multivariate mathematical–statistical methods for data evaluation*, in *Medical Applications of Mass Spectrometry*, K. Vékey, A. Telekes, and A. Vertes, Editors. 2008, Elsevier: Amsterdam. p. 141-169.
- [11] Wold, S. and M. Sjöström, *Chemometrics, present and future success*. Chemometrics and Intelligent Laboratory Systems, 1998. **44**(1): p. 3-14.
- [12] Mandrell, C.T., T.E. Holland, J.F. Wheeler, S.M.A. Esmaeili, K. Amar, et al., *Machine Learning Approach to Raman Spectrum Analysis of MIA PaCa-2 Pancreatic Cancer Tumor Repopulating Cells for Classification and Feature Analysis*. Life, 2020. **10**(9): p. 181.
- [13] Ullah, R., S. Khan, S. Javaid, H. Ali, M. Bilal, et al., *Raman spectroscopy combined with a support vector machine for differentiating between feeding male and female infants mother's milk*. Biomedical optics express, 2018. **9**(2): p. 844-851.
- [14] Sattlecker, M., C. Bessant, J. Smith, and N. Stone, *Investigation of support vector machines and Raman spectroscopy for lymph node diagnostics*. Analyst, 2010. **135**(5): p. 895-901.
- [15] Yan, S., S. Wang, J. Qiu, M. Li, D. Li, et al., *Raman spectroscopy combined with machine learning for rapid detection of food-borne pathogens at the single-cell level*. Talanta, 2021. **226**: p. 122195.
- [16] Glaws, A., R. King, and M. Sprague, *Deep learning for in situ data compression of large turbulent flow simulations*. Physical Review Fluids, 2020. **5**(11): p. 114602.
- [17] A., S.K., *The Era of Artificial Intelligence, Machine Learning, and Data Science in the Pharmaceutical Industry*. 2021: Academic Press.
- [18] Aggarwal, C.C., *Neural Networks and Deep Learning: A Textbook*. Springer.

- [19] Flach, P.A., *Machine learning : the art and science of algorithms that make sense of data*. 2017, Cambridge ; New York: Cambridge University Press.
- [20] Yamaguchi, T., Y. Kimura, and N. Hirota, *Solvent and Solvent Density Effects on the Spectral Shifts and the Bandwidths of the Absorption and the Resonance Raman Spectra of Phenol Blue*. *The Journal of Physical Chemistry A*, 1997. **101**(48): p. 9050-9060.
- [21] Tobin, M.C., *Sample Problem in Raman Spectroscopy*, in *Developments in Applied Spectroscopy*, A. W.D., Editor. 1962, Springer: Boston, MA.
- [22] Marquardt, B. *Addressing the Challenges of Process Raman Spectroscopy*. *Spectroscopy* April 2016 [cited 36; Available from: <https://www.spectroscopyonline.com/view/addressing-challenges-process-raman-spectroscopy>].
- [23] Kostamovaara, J., J. Tenhunen, M. Kögler, I. Nissinen, J. Nissinen, et al., *Fluorescence suppression in Raman spectroscopy using a time-gated CMOS SPAD*. *Optics Express*, 2013. **21**(25): p. 31632-31645.
- [24] Wei, D., S. Chen, and Q. Liu, *Review of Fluorescence Suppression Techniques in Raman Spectroscopy*. *Applied Spectroscopy Reviews*, 2015. **50**(5): p. 387-406.
- [25] Butler, H.J., Ashton, L., Bird, B., Cinque, Gi., Curtis, K., Dorney, J., Esmonde-W. K., Fullwood, N. J., Gardner, B., Martin-H., Pe L., and M.J. Walsh, McAinsh, M. R., Stone, N., Martin, F. L., *Using Raman spectroscopy to characterize biological materials*. *Nature Protocols*, 2016. **11**(4): p. 664-687.
- [26] Plaut, E., *From Principal Subspaces to Principal Components with Linear Autoencoders*. *ArXiv*, 2018. **abs/1804.10253**.
- [27] Ithapu, V.K., Singh, V., Johnson, S. C., *Chapter 15 - Randomized Deep Learning Methods for Clinical Trial Enrichment and Design in Alzheimer's Disease*, in *Deep Learning for Medical Image Analysis*, S.K. Zhou, H. Greenspan, and D. Shen, Editors. 2017, Academic Press. p. 341-378.
- [28] Hodge, V.J. and J. Austin, *A Survey of Outlier Detection Methodologies*. *Artificial Intelligence Review*, 2004. **22**(2): p. 85-126.
- [29] Sharma, S., C. Bellinger, N. Japkowicz, R. Berg, and K. Ungar. *Anomaly detection in gamma ray spectra: A machine learning perspective*. in *2012 IEEE Symposium on Computational Intelligence for Security and Defence Applications*. 2012.
- [30] Hofer-Schmitz, K., Nguyen, H., Berwanger, K. *One-class Autoencoder approach to classify Raman spectra outliers*. in *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning 2018*. Bruges (Belgium).
- [31] Houston, J., Glavin, F. G., Madden, M. G., *Robust Classification of High-Dimensional Spectroscopy Data Using Deep Learning and Data Synthesis*. *Journal of Chemical Information and Modeling*, 2020. **60**(4): p. 1936-1954.
- [32] Fan, X.-g., Y. Zeng, Y.-L. Zhi, T. Nie, Y.-j. Xu, et al., *Signal-to-noise ratio enhancement for Raman spectra based on optimized Raman spectrometer and convolutional denoising autoencoder*. *Journal of Raman Spectroscopy*, 2021. **52**(4): p. 890-900.
- [33] Olshausen, B.A., Field, D. J., *Sparse coding with an overcomplete basis set: A strategy employed by V1?* *Vision Research*, 1997. **37**(23): p. 3311-3325.
- [34] Idowu, O.P., J. Huang, Y. Zhao, O.W. Samuel, M. Yu, et al., *A stacked sparse auto-encoder and back propagation*

*network model for sensory event detection via a flexible ECoG*. Cognitive Neurodynamics, 2020. **14**(5): p. 591-607.

[35] Aslam, M. A., Xue, C., Chen, Y., Zhang, A., Liu, M., Wang, K. Cui, D., *Breath analysis based early gastric cancer classification from deep stacked sparse autoencoder neural network*. Scientific Reports, 2021. **11**(1): p. 4014.

[36] E., S.K. *Convolutional Neural Network / Deep Learning*. 2020; Available from: <https://developersbreach.com/convolution-neural-network-deep-learning/>.

[37] Umehara, M., H.S. Stein, D. Guevarra, P.F. Newhouse, D.A. Boyd, et al., *Analyzing machine learning models to accelerate generation of fundamental materials insights*. npj Computational Materials, 2019. **5**(1): p. 34.

[38] Liu, J., M. Osadchy, L. Ashton, M. Foster, C.J. Solomon, et al., *Deep convolutional neural networks for Raman spectrum recognition: a unified solution*. Analyst, 2017. **142**(21): p. 4067-4074.

[39] Lafuente, B., R.T. Downs, H. Yang, and N. Stone, *Highlights in Mineralogical Crystallography*, in *1. The power of databases: The RRUFF project*, A. Thomas and D. Rosa Micaela, Editors. 2015, De Gruyter (O). p. 1-30.

[40] Wahl, J., M. Sjö Dahl, and K. Ramser, *Single-Step Preprocessing of Raman Spectra Using Convolutional Neural Networks*. Appl Spectrosc, 2020. **74**(4): p. 427-438.

[41] Ho, C.-S., N. Jean, C.A. Hogan, L. Blackmon, S.S. Jeffrey, et al., *Rapid identification of pathogenic bacteria using Raman spectroscopy and deep learning*. Nature Communications, 2019. **10**(1): p. 4927.

[42] Wu, J., S. Qiu, Y. Kong, L. Jiang, Y. Chen, et al., *PCANet: An energy perspective*. Neurocomputing, 2018. **313**: p. 271-287.

[43] Chan, T., Jia, K., Gao, S., Lu, J., Zeng, Z., Ma, Y., *PCANet: A Simple Deep Learning Baseline for Image Classification?* IEEE Transactions on Image Processing, 2015. **24**(12): p. 5017-5032.

[44] Aly, S., Mohamed, A. , *Unknown-Length Handwritten Numeral String Recognition Using Cascade of PCA-SVMNet Classifiers*. IEEE Access, 2019. **7**: p. 52024-52034.

[45] Weng, S., H. Yuan, X. Zhang, P. Li, L. Zheng, et al., *Deep learning networks for the recognition and quantitation of surface-enhanced Raman spectroscopy*. Analyst, 2020. **145**(14): p. 4827-4835.

[46] Ouyang, L., W. Ren, L. Zhu, and J. Irudayaraj, *Prosperity to challenges: recent approaches in SERS substrate fabrication*. Reviews in Analytical Chemistry, 2017. **36**(1).

[47] Weng, S., W. Zhu, P. Li, H. Yuan, X. Zhang, et al., *Dynamic surface-enhanced Raman spectroscopy for the detection of acephate residue in rice by using gold nanorods modified with cysteamine and multivariant methods*. Food Chemistry, 2020. **310**: p. 125855.

[48] Britz, D. *Recurrent Neural Networks Tutorial, Part 1 – Introduction to RNNs*. 2015; Available from: <http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/>.

[49] Gers, F., N. Schraudolph, and J. Schmidhuber, *Learning Precise Timing with LSTM Recurrent Networks*. Journal of Machine Learning Research, 2002. **3**: p. 115-143.

[50] Britz, D. *Recurrent Neural Network Tutorial, Part 4 – Implementing a GRU/ LSTM RNN with Python and Theano*. 2015; Available from: <http://www.wildml.com/2015/10/recurrent-neural-network-tutorial-part-4-implementing-a-grulstm-rnn-with-python-and-theano/>.

- [51] Zhu, J., Z. Yang, M. Mourshed, Y. Guo, Y. Zhou, et al., *Electric Vehicle Charging Load Forecasting: A Comparative Study of Deep Learning Approaches*. Energies, 2019. **12**(14): p. 2692.
- [52] Wang, P., L. Guo, Y. Tian, J. Chen, S. Huang, et al., *Discrimination of blood species using Raman spectroscopy combined with a recurrent neural network*. OSA Continuum, 2021. **4**(2): p. 672-687.
- [53] Guo, Z., X. Lv, L. Yu, Z. Zhang, and S. Tian, *Identification of hepatitis B using Raman spectroscopy combined with gated recurrent unit and multiscale fusion convolutional neural network*. Spectroscopy Letters, 2020. **53**(4): p. 277-288.
- [54] Zhao, W. and S. Du, *Learning multiscale and deep representations for classifying remotely sensed imagery*. ISPRS Journal of Photogrammetry and Remote Sensing, 2016. **113**: p. 155-165.
- [55] Cole, M.R., *Hands-On Machine Learning with C# : Build smart, speedy, and reliable data-intensive applications using machine learning*. 1 ed. 2018: Packt Publishing.
- [56] Kolodiazhnyi, K., *Hands-On Machine Learning with C++: Build, train, and deploy end-to-end machine learning and deep learning pipelines*. 2020: Packt Publishing.
- [57] Raschka, S., *Python Machine Learning: Unlock deeper insights into Machine Learning with this vital guide to cutting-edge predictive analytics*. 2015: Packt Publishing.
- [58] Lantz, B., *Machine Learning with R - Second Edition: Expert techniques for predictive modeling to solve all your data analysis problems*. 2015: Packt Publishing.