University of Kent

Doctor of Philosophy

Deep Learning for Raman Spectroscopy: Bridging the Gap between Experimental Data and Molecular Analysis

Author:
Alex Poppe

Supervisor:
Dr Stuart Gibson

School of Physics and Astronomy
4th July 2024

Abstract

RAMAN spectroscopy is an analytical method frequently used in the fields of materials science and general chemistry, which measures the characteristic responses of molecules to light. Being a non-destructive technique, it has many scientific and industrial applications, such as material identification, drug production and airport security. By combining Raman spectroscopy with machine learning, powerful tools can be developed to make accurate predictions on unknown substances or quantities, without explicit programming.

This thesis focuses on two main areas. Firstly, through the application of machine learning and image processing, a novel tool was developed to study single molecule interactions with metal surfaces using surface-enhanced Raman spectroscopy (SERS). A convolutional autoencoder (CAE) architecture was utilised in a processing pipeline alongside various image processing techniques to extract and isolate complex, transient Raman features from SERS data. This was followed by a clustering process to obtain representative events pertaining to atomic-scale metal-molecule interactions on multiple catalyst surfaces, which provided a unique insight into the formation dynamics of atomic-scale features. The process was extended through the use of a Siamese convolutional neural network (Siamese-CNN) to incorporate spatiotemporal information relating to interactions between individual vibrational modes. This foundational research paves the way for tailoring metal-molecule interactions and assists in rational heterogeneous catalyst design. It introduces an analytical tool capable of studying metal-molecule interactions under the influence of strong local field gradients. This is a scenario that cannot be efficiently modelled with the conventional quantum mechanical method, density functional theory (DFT), which assumes a homogeneous field when analysing electronic structures of molecules.

Secondly, machine learning analysis has been applied to Raman data obtained in both nuclear and biopharmaceutical industrial applications. A key focus of this work is on the practical challenges faced in the design of data processing tasks and machine learning architectures due to real-world limitations in data collection. A fully connected (FC) autoencoder is employed as part of a regression task, which generates predictions on analyte concentrations in mixed substances. The method was shown to outperform industry standard regression tools, principal component regression (PCR) and partial least squares (PLS) regression, each used as comparative benchmarks, by over 50% in a test of model precision across various datasets in the investigated industrial applications. Advancements in the precision, speed and effectiveness of such tools are of critical importance in an industrial environment. This is driven by compelling motivations to reduce not only the costs associated with these procedures, but also to increase the quality of resulting products, or to reduce the risks within industrial operations, where applicable.

Acknowledgements

THE completion of this thesis owes a debt of gratitude to some remarkable people. Some of whom I have had the pleasure of meeting and collaborating with over the past four years, whilst others have been a part of my journey for so much longer. Many have directly influenced both me and my work, whilst others have done so indirectly. I shall forever remember their individual contributions.

First and foremost, I would like to express my gratitude to my main supervisor, Dr Stuart GIBSON. His unwavering support and invaluable guidance have led me through this journey. Whether through academic discussions beyond counting, or in more casual conversation, his trust in my abilities and commitment to this research have left me profoundly thankful. This gratuity also extends to my second supervisor, Dr Timothy KINNEAR. Throughout my time at the University of Kent, he has been a continuous source of inspiration for me. Despite my endless barrage of questions (which must have been quite annoying at times!), he has always responded to them with great enthusiasm and patience.

Considerable recognition goes to the amazing people I have collaborated with at the NanoPhotonics Centre in the University of Cambridge. Starting with Dr Bart DE NIJS, with whom important contributions had been made to our research, not to mention the paper we had worked on together alongside Stuart. Special thanks goes out to him for the hands-on experience in preparing one of the SERS experiments!. I would also like to thank Dr Jack GRIFFITHS, Dr Junyang HUANG, Dr Shu HU, and Prof Jeremy BAUMBERG for our numerous engaging discussions on the topics of nanophotonics, physics and machine learning. I appreciate the warm welcome I received when I visited to present our research to fellow members of your group. Furthermore, I am thankful for their efforts in procuring the extensive datasets that have played a pivotal role in our collaborative work.

I am immensely grateful to the team at IS-Instruments Ltd. with whom I had collaborated extensively throughout a significant portion of my research. In particular DR Michael FOSTER, Charles WARREN, and Dr Will BROOKS, who provided indispensable insight on Raman spectroscopy applications in the real-world. I am also thankful to them for collecting the spectral datasets upon which our research was based, and for valuable feedback on the structure of my thesis. Beyond this, I appreciate the opportunity to have worked alongside them part-time whilst writing this thesis, and I am excited to see where our future work together leads!

From the very beginning of my PhD I had benefitted from the in-depth knowledge on machine learning and data processing from both Dr Fangliang BAI and Dr Margarita OSADCHY. From Fangliang helping me to set up TensorFlow and assist in my understanding of the fundamentals, to Margarita providing me with critical feedback on the various neural networks I had designed and trained throughout my research, I am indebted.

I would also like to thank the institutions and companies that funded my research, namely the University of Kent, IS-Instruments Ltd., and VisionMetric Ltd. (with repeated recognition to the

contributions of Dr Stuart GIBSON and Dr Fangliang BAI from this organisation). Their support made this research opportunity a reality, allowing me to expand my depth of knowledge and delve into the unknown with confidence thanks to those brilliant individuals mentioned before.

Moving onto a more personal note, I would like to give thanks to the many people who indirectly contributed to this thesis (whether they know it or not!). To all my friends spanning the globe, of whom there are far too many to list their individual names, with the exception of two: Nicholas LONGDEN, I am grateful to have known you for so long, being someone who I could always chat to and play video games with (and for being someone I can rely on whenever I run into the odd tech-related issue!); and Siobhan IRVINE, though I have only gotten to know you this year, it feels like we have been lifelong friends - I only hope to make that feeling a reality in the many exciting years to come!

I would like to extend that thanks to all the wonderful people I have met across the various online communities, all of whom have kept me company (and sane!) since before my PhD began, throughout lockdown, and up to the present. Most recently of which are ECHO and all the associated communities for the opportunities they have given me. Most importantly of which being the folks over at The TREEHOUSE, whom I have have the pleasure of getting to know this year. They are a diverse bunch of people that I would not trade the world for. I shall cherish every single day that I get to spend with each and every one of you, and value your company far beyond what mere words can express.

Last but by no means least, I would like to acknowledge my family, in particular my parents Colin POPPE and Annette POPPE, my brother Lloyd POPPE, his partner Rebecca STOCK, and my newborn niece Madison POPPE. Thank you for steering me along the correct course. From a simple cup of tea in the morning, to invaluable guidance through difficult times, without you all there every step of the way, I certainly would have fallen from the path.

Declarations

- The scientific contribution of this thesis is focused on the development and application of machine learning analysis pipelines for Raman spectroscopy.
- The work presented in Chapters 3 and 4 was undertaken in collaboration with the NanoPhotonics group at the University of Cambridge. Members of this research group provided the data used in this research and expertise in SERS.
- The work described in Chapter 3 was adapted and extended from work which has been published in JPCL [1].
- The code used in Chapters 3 and 4 can be found here: github.com/APoppe95/Scanalyser-v1.git
- The work presented in Chapters 5 and 6 was undertaken in collaboration with IS-Instruments Ltd., who provided data and insights pertaining to industrial applications of Raman spectroscopy.
- The code used in Chapters 5 and 6 is restricted due to its commercial relevance.

Contents

List of Abbreviations

ANN	**A**rtificial **N**eural **N**etwork
BCE	**B**inary **C**ross-**E**ntropy
BPT	**Bip**henyl-4-**T**hiol
CAE	**C**onvolutional **A**uto**e**ncoder
CCD	**C**harge-**C**oupled **D**evice
CNN	**C**onvolutional **N**eural **N**etwork
DFT	**D**ensity **F**unctional **T**heory
DNN	**D**eep **N**eural **N**etwork
FC	**F**ully **C**onnected
FFT	**F**ast **F**ourier **T**ransform
FT	**F**ourier **T**ransform
IR	**I**nfra**r**ed
IgG	**I**mmuno**g**lobulin **G**
MSE	**M**ean **S**quared **E**rror
NPoM	**N**ano**p**article-**o**n-**M**irror
PCA	**P**rincipal **C**omponent **A**nalysis
PCR	**P**rincipal **C**omponent **R**egression
PLS	**P**artial **L**east **S**quares
POCO	**P**ost **O**perational **C**lean **O**ut
RRS	**R**esonance **R**aman **S**pectroscopy
ReLU	**Re**ctified **L**inear **U**nit
SAM	**S**elf-**A**ssembled **M**onolayer
SERS	**S**urface-**E**nhanced **R**aman **S**pectroscopy
SG	**S**avitzky-**G**olay
SHS	**S**patial **H**eterodyne **S**pectrometer
SNR	**S**ignal-to-**N**oise **R**atio
TBP	**T**ri**b**utyl **P**hosphate
UVRRS	**U**ltra**v**iolet **R**esonance **R**aman **S**pectroscopy

Chapter 1:

Introduction to the Thesis

Contents

RAMAN spectroscopy serves as a method for delineating the characteristic attributes of substances. This feat is achieved through the use of a laser to promote the inelastic scattering of photons from vibrational energy levels, revealing the vibrational modes of the system. In addition, rotational [2] and low-frequency [3] modes may also be probed using this technique. The result of this Raman interaction is the formation of a spectrum, which is comprised of one or more peaks that correspond to particular vibrational modes in a molecular system. These interactions are stated in wavenumbers - given in units of inverse distance, cm^{-1}, correlating to the energy of the Raman interaction; it is also common for a spectrum to be displayed using interactions per wavelength, given in units of nanometres, nm. It is possible to convert between unit scales as a function of the wavelength of the laser used to produce the Raman spectrum. Peaks in a Raman spectrum are categorised into two distinct wavenumber regions: Stokes and anti-Stokes. The positions of peaks correspond to the energy shifts of inelastically scattered photons. Positive wavenumbers - which are more common - signify reductions in energy that are associated with the Stokes region, whilst negative wavenumbers indicate energy promotions in the anti-Stokes region.

1.1 Raman Spectroscopy: Its History and Relevance in Modern Science

Raman spectroscopy, or more specifically the Raman effect, was discovered by the Indian physicist Chandrasekhara Venkata Raman (C. V. Raman) in 1928. The discovery was part of a series of investigations into the properties of light diffraction from molecules, which originates back to 1922 [4] when C. V. Raman was scrutinising the accepted explanation at the time for the blue colour of the sea, which was based on the Rayleigh scattering phenomenon that explains the blue colour of the sky. Through this research, it is now known that the blue colour of water has its origins in the molecular structure of water itself, rather than through reflections from the sky, due to the absorption of longer wavelength

light (in the red-orange region). Based on this research, he and his team discovered that non-incident light could be scattered from molecules, through refinement of their experimental procedure. This led to his 1928 publication titled “A New Radiation" [5], which earned him the 1930 Noble Prize in physics.

Being a subset of the broader field of spectroscopy, Raman spectroscopy has found widespread use as a tool for the measurement and analysis of interactions between electromagnetic radiation used as a probe, and the molecular material that is being studied. As the information provided by this technique is given by changes in energy within the molecular system, it bears similarities to a related technique of infrared (IR) spectroscopy. Where Raman spectroscopy provides information through the excitation of ‘Raman active’ vibrational modes, which originate from polarisation changes within a molecule, IR spectroscopy provides information through the excitation of ‘IR active’ vibrational modes, which originate from changes in the dipole moment of a molecule.

Another key difference between the two methods is that Raman spectroscopy measures the relative frequency of inelastically scattered radiation - based on the incident laser wavelength - whereas IR spectroscopy measures the absolute frequency. Based on the excitation of different vibrational modes, Raman and IR spectroscopy can be seen as complementary techniques to one another, each being able to provide information about a molecular system that the other typically cannot access. To better distinguish the two chemical analysis techniques, these descriptions of the differences between Raman and IR spectroscopy are further expanded upon in the theory chapter of this thesis.

Due to the process by which information is obtained through Raman spectroscopy, it has become a useful tool for chemical analysis to study the molecular properties of materials. Such properties include: the chemical structures of molecules, as the same atoms can result in different Raman spectra based on specific molecular arrangements [6]; the intramolecular bonds, whereby peaks corresponding to individual bonds enable the distinction of bond types between the same atoms, such as carbon-carbon single, double or triple bonds; and the phase transition of molecules based on the temperature at which a transition would occur [7]. Importantly, with respect to these features, Raman spectroscopy is a non-invasive technique, meaning that the procedure does not involve contact with any targeted substance besides a physical interaction with the laser, this therefore enables repeat analyses to be carried out in tandem with other spectroscopic methods.

Standard Raman spectroscopy is colloquially known as ‘spontaneous Raman spectroscopy’, and it has many scientific and industrial applications. As described, the process is non-destructive and non-contacting, with the advantage of requiring no sample preparation, which enables the analysis of substances in-situ. Such examples include the use of spontaneous Raman spectroscopy to study the properties of water molecules in-situ at a solid-liquid interface [8], which possess different structural properties from bulk water that may promote advancements in electrocatalytic processes; or studying the structural evolution of metallic electrocatalysts (electrodes) during an electrocatalytic process, such as through Raman signals produced by metal oxides [9, 10, 11] or metal-organic frameworks [12].

Other common applications of spontaneous Raman spectroscopy are in the area of mineral and organic material characterisation. Typically by using reference databases of known materials, unknown substances can be identified based on their signature Raman responses [13]. This allows for applications in the distinction between polymorphs [6, 14], and in the detection of mineral impurities or contaminants where other experimental procedures are shown to fail [15]. As water and glass give weak Raman responses, the technique naturally extends to in-vivo and in-vitro applications, such as in-vivo analyses of human skin [16, 17], extending to applications in detecting and studying dermatological conditions [18, 19]; or in-vitro analyses in the form of biopsies [20].

1.2 Variants of the Raman Technique

There exists a multitude of variations of Raman spectroscopy, each of which has been developed for the purpose of extending the capabilities and application areas of the technique. This is achieved primarily by improving the spatial resolution or intensity of the Raman response, typically through the utilisation of specialised optical configurations including: specific excitation wavelengths tuned to the analyte, the use of multiple probe wavelengths, or the inclusion of metal geometries to significantly amplify the local electric field and hence the Raman response. Such variations are commonly combined to take advantage of their numerous attributes.

Listed below are a number of key variations to spontaneous Raman spectroscopy, with qualitative explanations regarding their history, development and function, as well as typical example applications.

Spatially Offset Raman Spectroscopy (SORS). Spontaneous Raman spectroscopy is limited to measuring the Raman response of surface or near-surface materials, whereas SORS is designed to extend the functionality of Raman spectroscopy, allowing for deeper penetration and thereby achieving depths of up to 5 cm [21]. This is achieved without needing a different optics configuration to that of standard spontaneous Raman spectroscopy. Developed by Matousek *et al.* in 2005 [22, 23], the SORS technique employs multiple collection points that are spatially offset from the point of incidence by differing amounts. The nature of each spectrum measured in this way differs due to the method by which Raman photons scatter laterally throughout deeper layers of the material; as photons diffuse into deeper layers, their migration can be described as a 'random walk'. This translates to collection points at greater offsets from the point of probe incidence having equal contributions from the Raman probe at all depths, as opposed to a collection point with zero offset that has an increased density of probe photons, and weaker contributions from deeper layers. By utilising this physical effect, multivariate analysis can be employed on a dataset of Raman spectra measured at various offset distances. These offsets yield distinct relative contributions from surface and sub-surface layers in relation to the distance from the incident probe. Consequently, this approach enables the extraction of approximate pure Raman spectra for individual layers within a material, achieved through a scaled subtraction of spectra at different offsets.

Extending the functionality of Raman spectroscopy to obtain Raman spectra of sub-surface materials with SORS has created a large number of uses for the technique. Where the surface layers of a material may take the form of a container, such as a plastic bottle or capsule, SORS is capable of measuring and identifying unknown liquids or powders [24, 25], which is often employed in airport security with regards to passenger luggage; in a similar vein, SORS finds use in police and military environments in the analysis and identification of potentially illegal drugs [26, 27] or explosive compounds [28, 29, 30] through non-metallic containers. This is possible whether the container is made of glass, plastic or paper and regardless of its colour or opacity [26]. Applications of SORS also extend to medical fields, such as in the non-invasive analysis and monitoring of bone beneath skin [21, 31], in the quality assessment of blood [32], or in the detection of breast cancer [33]. SORS has also been used as a tool in the art preservation of paintings by analysing sub-surface layers of paint [34], where spontaneous Raman spectroscopy is infeasible due to obscuration by the top layers of paint.

Resonance Raman Spectroscopy (RRS). Spontaneous Raman spectroscopy is a relatively weak effect, wherein inelastically scattered photons occur approximately once every one thousand scattered events, which in turn occurs at the same approximate rate compared to the non-interacting, direct transmission of photons through a molecular compound [35, 36]. Thus, to improve upon the sensitivity of spontaneous Raman spectroscopy, the resonance effect is taken advantage of by tuning the energy of the incident photons from the Raman probe to energies close to, or equal to that of, a particular electronic transition in the analyte [37, 38]. The first observation of this effect is dated at least as early as 1946 by Harrand and Lennuier [39, 40]. The result of this tuning provides an enhancement to the Raman scattering process in the order of 10^6 [41].

This resonance effect differs from the standard (non-resonance) Raman process as the interacting photons excite electrons to an excited electronic state, rather than a virtual state of lower energy. This consequently introduces enhancements to fluorescence as an interfering factor, as a portion of the excited electrons are able to relax to the lowest level of the excited state before their emission back to the ground state [40]. However, peaks produced by resonance Raman scattering are able to be distinguished from that of fluorescence due to the fact that they are narrower, owing to the conversation of energy whereby resonance Raman photons undergo direct emission back to the ground state without an initial relaxation in the excited state [42]. Another crucial aspect differentiating RRS from non-resonance Raman techniques is that the enhancement specifically affects the electronic transition states of the chromophores within a molecular compound [30], which are the parts of a molecule responsible for its colour and may not comprise the entirety of its structure, hence there may be notable differences between spectra of the same compound when measured with resonance and non-resonance Raman techniques.

Despite the aforementioned intricacies, the enhancement effect enables the analysis of molecular compounds in either low concentrations [43], or with vibrational modes that possess weak Raman

responses [40]. Additionally, there is an improvement in the selectivity from spontaneous Raman spectroscopy, in which an amplification occurs in the intensity of the vibrational modes that are under resonance conditions, which enables RRS to serve as a technique for analysing specific vibrational modes [43, 44], or to obtain structural information regarding large biomolecules such as proteins [40, 44, 45]. Relative peak ratios as a result of this selectivity must be considered when comparisons are made between RRS spectra and those from other non-resonance Raman techniques [30, 46]. Similar to SORS, there are other applications for RRS in areas of art preservation [47] and forensic analyses involving the characterisation and comparison of inks and paints [30, 40, 46].

Stimulated Raman Spectroscopy (SRS). Where SORS and RRS are examples of extensions to spontaneous Raman spectroscopy involving standard optical configurations, with minor modifications to the signal collection method or properties of the laser probe, SRS makes a non-linear modification to the optical configuration by introducing a second photon source. The primary source uses 'pump' photons at a particular angular frequency ω_p, whilst the secondary source uses 'Stokes' photons at an angular frequency ω_s, normally chosen to match the energy of a particular vibrational or rotational transition [48] - named together as 'rovibrational'. Both photon sources may be delivered by the same probe laser. The effect of this pump-probe arrangement is the resonant amplification of the Raman response from a rovibrational transition in the order of 10^4–10^6 [49], should the energy of this transition equate to the energy difference between both photons. Regarding the energy of SRS, the enhancement factor is attributed to the coherent nature of molecular vibrations, in contrast to the inherent incoherence of spontaneous Raman spectroscopy. This coherency is explained by the synchronised oscillation of excited rovibrational modes, resulting in a directional polarisation that provides the enhancement [49].

Application areas for SRS entail mapping the spectral signature of an analyte substance, enabled by the rovibrational energy states made accessible by this technique. Such applications include: fibre optic communication over distances exceeding 800 km [50]; the development of Raman fibre lasers as a substitute for conventional silica fibres [50]; the use of SRS as a Raman shifter, wherein a Raman-active medium external to the laser source produces Stokes-shifted photons, ω_s, dependent on the material [49, 51]; the detection of different conformational structures (different rotational arrangements about a single bond) in the same organic compounds [52, 53]; and in biomedical imaging, by integrating SRS with microscopy to study the behaviour of living tissue when subjected to various stimuli [54].

Coherent Anti-stokes Raman Spectroscopy (CARS). Another coherent variation of spontaneous Raman spectroscopy, CARS shares numerous operational and mechanical functions with SRS. The CARS technique was originally discovered in 1964 by Maker and Terhune [55], and later named as such in 1974 [56]. By utilizing two photon sources in a non-linear optical process, CARS amplifies the Raman signal from molecules compared to conventional spontaneous Raman spectroscopy. Similar to

SRS, there exists a laser source emitting pump photons with frequency ω_p, and Stokes photons with frequency ω_s. However, a distinctive four-wave Raman mixing process occurs through a second interaction with the pump laser. The four-wave interaction progresses in stages: pump photons initially excite electrons from the ground state to a virtual state. Subsequently, stimulated emission to a higher vibrational energy level takes place due to the presence of the Stokes photons. Next, a second pump photon interacts with this polarised electron, causing an excitation to an even greater virtual state, with an energy of $\omega_p - \omega_s$ above the first, culminating in the emission of an anti-Stokes signal as the electron returns to the original ground state. Consequently, the primary aim of this third-order optical configuration is to produce a coherent signal with the analyte, resulting in an anti-Stokes frequency of $2\omega_p - \omega_s$ [56, 57]. The Raman response can be resonantly enhanced when the frequency difference between the pump and Stokes photons approaches a Raman frequency of the analyte [55].

CARS and spontaneous Raman spectroscopy both examine the same vibrational modes, but are distinguished by the nature of the Raman signal and the type of interference. Spontaneous Raman spectroscopy mainly focuses on the Stokes region and thus contends with interference from fluorescence, which occurs when electrons relax from higher energy states to the ground state, emitting lower-energy (positive wavenumber) photons in the Stokes region. As a result, CARS avoids this fluorescence interference by primarily occupying the anti-Stokes region [58]. However, CARS signals must contend with non-resonant, coherent interference from other transitions within a molecule, thus the use of CARS is limited by the signal-to-noise ratio (SNR) of the resonant response over the non-resonant inference, rather than by fluorescence as in spontaneous Raman spectroscopy [59].

As CARS contains a strong anti-Stokes Raman response, it is used in measuring the temperature of gases during combustion processes. The temperature of a substance may be calculated by ratio of the Stokes and anti-Stokes response, which scale non-linearly with the temperature of the analyte. CARS is used to monitor a combustion process by determining the temperature of the analyte, such as N_2, O_2, CO_2 or CH_4 [60, 61, 62, 63], which aids in the development of more efficient fuels. CARS has also been integrated with microscopy to image cells and living tissue [64], such as in lipids (fatty acids) due to the strong anti-Stokes response of the C-H bonds [65] and the in-vivo analysis and mapping of sciatic nerve tissue by surrounding fat cells via the same chemical response [66].

Surface-Enhanced Raman Spectroscopy (SERS). Variations described thus far extend the functionality of spontaneous Raman spectroscopy through modifications to excitation wavelengths of the Raman probe (RRS), or through multiple collection points (SORS) or photon frequencies (SRS and CARS). SERS is another variation capable of extending the functionality of spontaneous Raman spectroscopy, which is achieved through enhancements to the local electric field. As the Raman effect is proportional to the local electric field, any such enhancements have the ability to produce enormous increases to the measured signal - by up to 10^{12} [67, 68, 69]. There are multiple ways to achieve an enhancement of the local electric field, with typical methods involving binding an analyte to either a

roughened metal surface or nanoparticle arrangement [70], which excites local surface plasmons (analogous to photons, these are oscillations of the electron density) that facilitate the field enhancement. The metal surface is commonly composed of gold or silver, although other less expensive metals have been investigated such as aluminium [71, 72]. This signal boost enables the analysis of substances at very low concentrations, or even single molecules [73, 74, 75].

The effect was first discovered by Fleischmann *et al.* in 1973 [76] in a paper that studied the adsorption (the ability for the surface of a solid material to accumulate chemical compounds or gases that it is in contact with) of pyridine onto a roughened silver electrode. The observed enhancement could not be explained by the concentration of the analyte, as the pyridine formed at a monolayer or near-monolayer thickness on the silver surface. This lead to competing theories attempting to explain the enhancement mechanism, both of which are currently accepted to this day [77]. These are the electromagnetic theory [78, 79, 80] and the chemical theory [80, 81, 82], which will be expanded upon in the theory chapter of this thesis.

As mentioned, SERS is a high sensitivity variation to spontaneous Raman spectroscopy, capable of detecting the presence of molecules in extremely low concentrations. Hence, it is a useful technique with many applications areas [83] including: measuring responses from low population proteins in the detection of various cancers [84, 85, 86]; future design of molecular electronics and computer memory [73, 87, 88, 89]; and the design of heterogeneous catalysis, in which the catalytic material exists in a different phase to the reactant, with uses in many industries such as food processing, biopharmaceutical drug manufacturing, and plastic production [90, 91, 92, 93, 94].

1.3 Chemometrics in Raman Spectroscopy

As mentioned in Section 1.2, the vast majority of Raman spectroscopy applications are subject to fluorescence, which adds interference components to the resulting Raman spectra. There are numerous modifications that can be made to the Raman setup to attempt to counteract this issue, such as: changing the wavelength of the Raman probe laser to shift the fluorescence profile away from any peaks of interest, switching to a pulsed laser to leverage the difference in timescale between fluorescence and Raman scattering [95] (these examples are covered in more details in the theory chapter of this thesis), or by utilising a variant to spontaneous Raman spectroscopy such as CARS.

However, fluorescence effects are practically unavoidable, and are not the only source of interference for the information targeted in a Raman measurement. Such additional components include but are not limited to: noise from the detector (shot noise, read noise, dark currents), cosmic rays forming spikes in a spectrum, and aberrations in the optical equipment (such as dust on a lens) [96]. Therefore, it is typical for a procedure involving one or more data preprocessing techniques to be implemented [20, 97] in order to reduce the effects of fluorescence and other interfering components - termed 'background interference'. The removal of background interference is an important step in chemometrics, which

involves a statistics-driven analysis of a dataset of Raman spectra. Otherwise, any results or conclusions drawn from such an analysis will not be solely based on characteristic information retrieved from Raman measurements, but rather in combination with the described background inference. It should be noted that, although chemometrics is a broad field that can be applied to numerous types of chemical data, both spectral and non-spectral, this introduction predominantly focuses on Raman spectra.

Another distinctive factor that contributes an interfering element to Raman spectra, apart from the described background interference, is the nature of the sample itself. Given that the majority of substances are composed of a mixture of various molecular compounds, whether due to contaminants or impurities from an industrial process, or through necessity if the analyte requires storage in an aqueous solution, the resultant Raman spectrum of an analyte can contain peaks originating from these other molecular compounds. These additional Raman peaks have the potential to degrade a chemometric analysis, leading to biased, inaccurate, or erroneous outcomes. Such inaccuracies could manifest as the false identification of the analyte in an unknown sample, an incorrect prediction of its concentration or temperature (if Stokes and anti-Stokes regions are considered), and similar issues.

To remove background interference, and thereby improve the SNR of the Raman spectrum for subsequent analysis, an analytical chemist would conventionally perform preprocessing tasks. A subtraction of the baseline would typically be made through the use of a spline fit to the gross structure of each spectrum in a database. Each signal may also be smoothed and denoised using a Savitzky-Golay (SG) filter, which fits a series of polynomials across the spectral range, however, this process can degrade the Raman features present in the spectra. Once the spectral dataset has been preprocessed, the next stage is to perform a statistical analysis to extract the desired information. This may involve the use of a multivariate analysis technique such as principal component analysis (PCA) to distill the information present in a dataset [98], which allows for the removal of noise and other remaining components that do not pertain to the bulk of explainable variance. After which, further task-specific analyses would take place such as fitting a linear model (e.g. ordinary least squares) in a regression task [20, 98], or clustering (e.g. K-means clustering) in a classification task [20, 96, 97, 98, 99].

1.4 Example Machine Learning Techniques in Raman Spectroscopy

There are several disadvantages associated with assigning the task of data processing to a human. These include the task becoming prohibitively slow as the dataset size grows and the potential for subjectivity in the preprocessing stages, which could lead to variations in the preprocessing of the same arbitrary spectrum when carried out by different people. To mitigate these drawbacks, machine learning tools can be utilized instead of direct human involvement in data preprocessing and analysis, which may bring about additional benefits by negating the need for the repeated training or development of an analytical model when new data samples are introduced should statistical properties not deviate too strongly from the original dataset. With regards to an industry setting, there may be a financial

incentive to implementing these machine learning tools, as development and deployment costs may be considerably less than the cost of assigning the task to a domain expert.

In recent decades machine learning techniques have been applied to a wide range of scattering and spectroscopic applications, such as Raman spectroscopy, brought on both by advancements in computing power and the increasing availability of large, complex spectral datasets. Listed below are a number of historic machine learning techniques that have played an important role in the development of modern machine learning - including details of their development and qualitative descriptions of their operation - that are still used today either as standalone analysis techniques, or as a comparative benchmark for more advanced algorithms.

Support Vector Machines (SVM). Originally conceptualised in 1964 by Chervonenkis and Vapnik [100] and adapted over a 30-year period to the modern application in 1995 by Cortes and Vapnik [101, 102]. SVM is a binary classifier where the objective is to form a deterministic hyperplane, or decision boundary, which effectively separates samples into one of two distinct classes. This is achieved by forming a set of 'support vectors', which are a subset of training data points represented as vectors in a multi-dimensional space that lie nearby to a proposed hyperplane. These support vectors are used to evaluate and maximise the margin of separation (i.e. minimise the generalisation error) to the hyperplane, based on a chosen objective function constructed and solved as a quadratic optimisation task on the parameters of the hyperplane [102].

In cases with non-linear data, a kernel function such as the polynomial kernel or radial basis function is employed to map the data to a higher-dimensional space - this process is colloquially known as the 'kernel trick' - wherein the data becomes linearly separable and the aforementioned optimisation process may proceed [102]. As only support vectors are considered when defining the decision boundary, rather than all training data, SVM is therefore an efficient process in high-dimensional spaces. Once the SVM model is trained on a given task and the decision boundary is determined, new data points can be classified by identifying which side of the hyperplane they fall on.

SVM is one of the pioneering machine learning algorithms that has found chemometrics applications with Raman spectroscopy data primarily for classification tasks [103, 104, 105], although applications in regression on both Raman and IR spectroscopy data are also possible [106, 107]. Being a supervised learning model, SVM makes use of both the input data (e.g. a spectrum) and an output label (e.g. the associated chemical compound) for training the internal parameters.

Random Forests. An ensemble learning method that utilises multiple decision trees to determine the output of a classification or regression model, developed by Ho in 1995 [108]. To understand the goal of random forests as a machine learning tool, decision trees must first be understood. A decision tree, with algorithm implementations dating back as early as 1959 by Belson [109], is a hierarchical model that uses a tree-like structure, in which: each node represents a feature of the data, each branch

represents a condition or decision to be made, and each leaf represents the outcome or prediction of the model. The goal of a decision tree is to formulate a tree that best splits the data in a classification task, which is typically achieved through the use of the CART (classification and regression tree) algorithm [110]. The quality of the split may be evaluated using a simple estimator such as mean squared error (MSE). This method of model construction makes decision trees easily interpretable, although they are prone to overfitting to the training dataset.

Multiple decision trees are utilised in random forests by way of bootstrap aggregation, or 'bagging' [111], wherein random subsets of training data are chosen with replacement to train each decision tree. Importantly, the forest of decision trees is uncorrelated, which is made possible by randomly selecting with replacement a subset of features from the data accessible to each decision tree through the 'random subspace method' [112]. This approach achieves the same result as in a single decision tree but with a notable reduction in overfitting, thus random forests are preferable over single decision trees when trained on noisy data. The way in which results are obtained from a random forest is dependent on the specific task: for a classification task, the chosen class for a new sample is commonly determined by majority vote from all decision trees; for a regression task, the prediction is made by either the mean or median of all decision tree predictions.

Random forests have been used in Raman spectroscopy for a variety of applications, including: predicting the SERS response of organic compounds absorbed onto a gold surface as an effective substitute for expensive quantum mechanical predictions [113]; classifying, in combination with PCA, milk samples from different species with varying component concentrations such as proteins and fats, for use in nutritional research for infants [114]; immunology research involving the quantification of target hormones within dog serum, used together with SERS [115]; and the identification and dating of different copper polymorphs used as pigments in art at various stages of aging [116].

Bayesian Networks. Developed by Pearl in 1985 [117], Bayesian networks are probabilistic graphical models based on Bayesian probability theory [118]. Such graphs are comprised of nodes representing specific random variables that are interconnected by edges that explain some form of dependence. The specific type of graph used in Bayesian networks are directed acyclic graphs (DAG), which are directed graphs with no cycles - meaning that the edges express conditional dependencies between nodes, and that these edges do not form closed loops. Therefore, a DAG is said to be topologically ordered, whereby directed edges beginning at earlier nodes in a graph always end at later nodes. Each node in a Bayesian network has associated conditional probabilities that are dependent on all incoming edges from the parent nodes.

The main use of a Bayesian network is for probabilistic inference, in which predictions or classifications can be made on new samples based on the probability of certain variables or events given observed evidence - such evidence may take the form of intense Raman responses at determining wavenumbers for a particular chemical compound, for example. Similar to decision trees, the graphical structure of

a Bayesian network allows for direct visualisation of interactions between deciding variables that have influenced a particular outcome [119].

Because of these characteristics, Bayesian networks have been utilised across a number of applications, such as: microbiology, for the detection of harmful bacterial spores to prevent food-borne illnesses in food production using confocal micro-Raman spectroscopy [120]; chemometrics analysis, in the quantification and identification of DNA sequencing using SERS [121, 122]; and in recent research into the quality control of biopharmaceuticals by integrating Raman spectra with other data types [123].

1.5 Neural Networks and Deep Learning in Raman Spectroscopy

The advent of the artificial neural network (ANN), and later deep learning, has revolutionised the applications and capabilities of not only Raman spectroscopy, but a multitude of other scattering and spectroscopic data analysis techniques [124] - although this introduction will focus on applications in Raman spectroscopy. This development has enabled the classification and prediction of chemical morphologies on datasets that may be considered either too large or noisy for conventional analysis techniques. To begin, an ANN is a branch of machine learning with a history of development stretching back over the past two centuries with the publication of the linear neural network model by Legendre [125]. Although, a more conventional interpretation for what constitutes an ANN would be a directed graph of nodes interconnected by weighted edges, which are crucially able to update, or learn, new values for these parameters based on observed input patterns, in order to improve in its ability to correctly associate those patterns with desirable outputs. Under this interpretation, the first 'learning' ANN was published by Amari in 1972 [125, 126] with the artificial recurrent neural network (RNN). Perhaps the single most pivotal moment in the history of machine learning is the introduction of the backpropagation algorithm, which has become the standard method by which neural networks learn today. Also known as 'reverse mode of automatic differentiation' in the original 1970 publication by Linnainmaa [127], backpropagation performs an efficient implementation of the Leibniz chain rule, and was first applied to train neural networks in 1982 by Werbos [125, 128]. A more detailed explanation on the structure of the ANN, various example architectures, and a mathematical description of the backpropagation algorithm are provided in the theory chapter of this thesis.

Soon after the modern ANN had been established, applications in Raman spectroscopy began to emerge [129]. In 1993, Liu *et al.* [130] used an ANN to classify Raman and near-IR spectra of organic compounds commonly used in industrial applications as solvents, extractants and additives. In 1994, Lewis *et al.* [131] trained an ANN as a binary classifier using the Raman spectra of two types of wood samples. In 1997, Gniadecka *et al.* [132] used an ANN to distinguish between healthy and cancerous skin samples, the results of which were in agreement with a manual spectral analysis.

An issue faced by ANNs, here referred to as 'shallow' architectures, is that these models, in much

the same way as the classical machine learning tools described in Section 1.4, are incapable of learning the complete set of complex features that may be present in much larger chemical databases that are more commonly seen today. Spectral preprocessing can be implemented to improve the performance of these models, outlined in Section 1.3, which is often a labour-intensive and subjective task. Selecting the correct preprocessing methods can significantly impact the results, and manual intervention by a domain expert may be warranted. Such subjectivity, as previously described, can introduce bias and inconsistency in the analysis, especially when multiple people may carry out the same preprocessing task on separate occasions, reducing the reliable of these methods.

Deep learning is a multifaceted data processing method that has been shown to address the aforementioned preprocessing challenges. Deep learning architectures have a wide range of applications in spectroscopy due to their ability to detect complex, often non-linear features, and process large quantities of data with high throughput. As a result, deep learning has provided powerful tools that can classify substances or predict quantities without the need for potentially bias-inducing preprocessing [133] steps, which are commonly required in alternative methods such as partial least squares (PLS) regression or shallow architectures. This attribute allows deep learning to process, for example, mixtures consisting of multiple chemical compounds [134], or spectra with highly variable baselines. Such methods are also capable of providing identification despite a small number of reference samples (or even from individual reference spectra) [135]. This is because deep neural networks (DNNs), which possess many layers each of increasing levels of abstraction, offer a robustness to variability in spectra that is not linked to the underlying information aimed to be qualified. Thus, they are particularly effective at categorising spectra pertaining to unique molecular compositions, states, or transitory physical events.

1.6 Organisation of the Thesis

Early chapters in this thesis are based on exploratory collaborative research with members of the Baumberg research group at the University of Cambridge. These chapters develop a machine learning pipeline to process and analyse atomic-scale features present in SERS data based on metal-molecule interactions in nanogaps. Later chapters cover work done in collaboration with an industrial sponsor for this PhD, IS-Instruments Ltd., to design and deploy machine learning regression models. A focus in these chapters is on realistic real-world limitations on data volumes, in two distinct industrial settings: nuclear, featuring high concentration samples collected with spontaneous Raman spectroscopy data; and biopharmaceutical, featuring low concentration samples collected using ultraviolet resonance Raman spectroscopy (UVRRS), a variant of RRS. Brief summaries of the contents of each chapter are provided below.

Chapter 2: Theory. Since all work presented in this thesis involves Raman spectroscopy, this chapter covers the fundamentals associated with the method. This includes descriptions distinguishing

elastic and inelastic scattering, Raman and IR active vibrational modes, and practical considerations regarding the experimental apparatus used to capture and process Raman signals, such as a differentiation between dispersive and Fourier transform (FT) detectors. The mechanisms that drive the SERS interaction are detailed, including two prevailing theories describing the observed enhancement factor, and an example geometry that promotes such interactions, from which the work in Chapters 3 and 4 is based on.

Finally, as with Raman spectroscopy, DNN-based machine learning features heavily throughout this thesis, hence the foundations are laid for the implementation and operation of such data processing tools, such as the backpropagation and gradient descent algorithms used to train neural networks. Examples are also provided for alternative architectures that are relevant to this thesis. Visualisations for how neural networks may learn increasingly abstract, complex features, and quantitative descriptions for a range of hyperparameters and optimisation tools, which are used throughout this thesis to adapt each model to the chosen tasks, are provided with example usages.

Chapter 3: Analysing Metal-Molecule Interactions on the Atomic-Scale. The contents of this chapter detail work contained in, and surrounding, published research by the author of this thesis [1]. This will cover the design of combined machine learning and image processing techniques to create a robust data processing pipeline for the chemometric analysis of multiple single molecule, time-series SERS datasets. The data used in this work was captured using an in-house spectrometer setup in a dispersive detector arrangement. In collaboration with researchers at the University of Cambridge, the goal of this work was to characterise the formation dynamics of atomic-scale processes, in this case adatoms, which play a key role in metal-molecule interactions and are critically important in heterogeneous catalysis and various other molecular electronic applications. Such characterisation was achieved through the design and implementation of a convolutional autoencoder (CAE), combined with image processing, for the extraction of so called 'picocavity' features, which are used to determine the formation sites of metallic protrusions through a comparison of 'picocavity peaks' with those predicted through conventional quantum mechanical modelling.

Chapter 4: Temporal Extension to the Metal-Molecule Analysis Pipeline. This chapter constitutes foundational research that expands upon the data analysis pipeline described in Chapter 3. Through the use of a Siamese convolutional neural network (Siamese-CNN), a binary classification task is formed to distinguish between positively and negatively correlated picocavity peaks. These time-series peaks shift in wavenumber space over time as a result of perturbations caused by the drifting of adatoms - the source of the picocavity - that produce strong local electric field gradients. By characterising the polarity of correlated peaks, this work provides a tool capable of results similar to the previous chapter, whilst incorporating additional temporal information present in the SERS data. Such information can aid in the tailoring of catalyst surfaces, or near-surfaces, by analysing how

individual bonds interact on single molecules, which can inform proposed modifications to catalysts in order to improve selectivity and efficiency for desirable catalytic processes.

Chapter 5: Regression Modelling of High-Concentration Raman Spectroscopy in the Nuclear Industry. The content in this chapter is connected to work done in collaboration with an industrial sponsor for this PhD, IS-Instruments Ltd. The focus of this work is on the design and implementation of a machine learning regression model for predicting the concentration of mixed Raman spectra for molecular compounds commonly found in nuclear decommissioning processes. The dataset features high concentration samples captured using a spontaneous Raman spectrometer with an FT detector arrangement. A fully connected (FC) autoencoder combined with a linear regression model is used to make the predictions, and the results are shown to exceed the performance of industry standard data processing tools: principal component regression (PCR) and PLS regression. A key theme explored within this chapter is on the limitations of low data volumes, which are common to industrial settings, and the impact this has on both the design of machine learning models and on data augmentation techniques implemented to introduce sufficient data variance required to train the model.

Chapter 6: Transferring Success: Low-Concentration UVRRS in the Biopharmaceutical Industry. In extension of Chapter 5, this chapter utilises the machine learning regression model to predict concentrations of bioorganic macromolecules dissolved in aqueous solutions at low concentrations. The results are also shown to exceed the performance of PCR and PLS regression models trained on the same data. Two datasets were captured for this task using an UVRRS system, each of which containing proteins crucial to the biopharmaceutical industry for the research and manufacturing of new therapeutic drugs. These organic molecules can form protein aggregates, which are detrimental to the manufacturing process, and can result in adverse effects in the resulting drugs. Hence, accurate information on the quantities of these mixtures is important for improving quality and yield. This chapter also explores modifications to the data augmentation technique, introduced in Chapter 5, to overcome detrimental effects on model performance caused by sample measurements collected at non-uniform concentrations intervals, as well as non-linear Raman responses inherent to both protein macromolecules due to sample attenuation.

Chapter 7: Outlook and Future Work. This chapter provides final remarks, and concludes the work undertaken throughout each chapter within this thesis. The potential for future work in expanding into other application areas is also discussed.

Chapter 2:

Theory

Contents

THIS chapter covers the broader theories and concepts that are prevalent within each chapter of this thesis, namely Raman spectroscopy, its extension SERS, and machine learning. The origins of machine learning are briefly covered, and descriptions are given for the various architectures and tools that are utilised within this thesis, which have been selected to optimise the models chosen for each task. Additional theories that feature a singular usage within this thesis do not appear within this chapter, and are instead covered in the respective chapters in which they appear.

2.1 Raman Spectroscopy

Raman spectroscopy is a fingerprint technique concerning the interaction of electromagnetic radiation on matter, which can be used to determine the identity, as well as the concentration, of unknown molecular substances. It is applicable to matter in solid, liquid or gaseous states, and even has applications in complex biological structures such as DNA [136, 137]. The process in which molecules are analysed is based on the interaction between the wavelength of the electromagnetic radiation and the electronic structure of the matter, which causes an excitation of specific vibrational modes, producing an electromagnetic spectrum that is characteristic to the substance.

2.1.1 Elastic and Inelastic Scattering

The key function by which Raman spectroscopy operates is through the inelastic scattering of photons, also called Raman scattering, which is used to identify the atomic composition of molecules based on electron transitions between quantised vibrational energy levels. When a source of monochromatic light interacts with a molecular system, the electrons within that molecule can become excited, bringing them from their initial ground state, E_0, up to a virtual state. In the majority of cases elastic scattering occurs, also called Rayleigh scattering, in which the excited electron relaxes back to its initial ground state, which emits a photon with an energy equal to that of the incident photon, $h\nu_0$. A far less common occurrence is inelastic scattering, in which excited electrons relax to an energy level that is either above, $E_0 + h\nu_n$, or below, $E_0 - h\nu_n$, its initial state, where $h\nu_n$ is the energy difference between the two states. In the former case, termed Stokes scattering, the energy of the emitted photon is less than the energy of the incident photon. In the latter case, termed anti-Stokes scattering, the energy of the emitted photon is greater. Hence this photon has energy $h\nu_0 \mp h\nu_n$, depending on the type of Raman scattering.

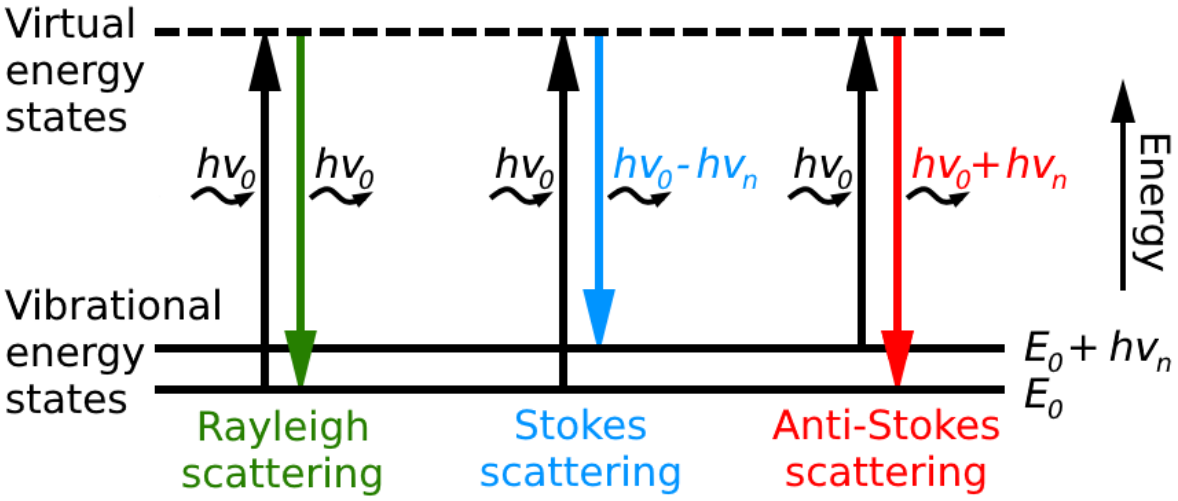


Figure 2.1: Energy level diagram for the three relevant scattering processes. *Left*, elastic scattering (Rayleigh scattering); *middle* and *right*, inelastic scattering (Stokes and anti-Stokes).

As shown in Figure 2.1, the difference in energy between two vibrational energy levels is equal to the difference in energy between the incident and scattered light. The resulting Raman shifts are expressed in wavenumbers with units of inverse distance, cm^{-1}, as this directly relates to energy. The wavenumber, Δk, can be calculated using the equation:

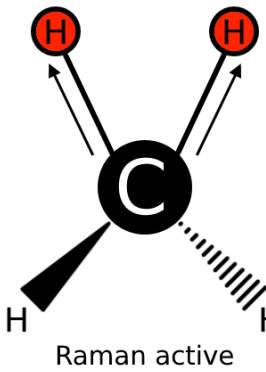
$$\Delta k = \frac{1}{\lambda_0} - \frac{1}{\lambda_n}, \tag{2.1}$$

where λ_0 is the wavelength of incident light, and λ_n is the wavelength of the emitted Raman light.

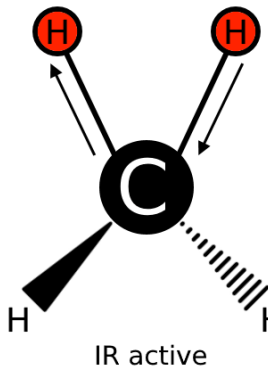
2.1.2 Raman and Infrared Vibrational Modes

A molecule undergoing Raman scattering will contain multiple vibrational modes, which are independent sets of atomic vibrational motions, which can be excited simultaneously with other modes. Each vibrational mode generates a peak in the Raman spectrum, which corresponds to the characteristic energy level transition that produced it. The resulting spectrum can therefore be used to identify the specific vibrational modes and chemical structure of a subject molecule. This gives rise to the term 'fingerprint technique' that is colloquially used to describe Raman spectroscopy. Due to the requirement of a change in polarisation for a Raman response (discussed below), the majority of materials produce a Raman signal regardless of phase, with the exception of pure metals whose structure prevents the vibrational Raman effect, and hence polarisation changes cannot occur.

With regards to the aforementioned vibrational modes, not all modes will necessarily be Raman-active. This means that Raman scattering would not excite these modes, leaving those regions empty within the Raman spectrum. Vibrational modes are either Raman- or IR-active. Raman-active modes require a change in the polarisability of the molecule, which occurs during symmetric changes in bond lengths during molecular vibrations, such as in Figure 2.2a. IR-active modes require a change in the dipole moment of a molecule, which occurs during asymmetric changes in bond lengths, such as in Figure 2.2b.



(a) Symmetric stretching, causes a change in polarisability.



(b) Asymmetric stretching, causes a change in the dipole moment.

Figure 2.2: Examples of Raman- and IR-active vibrational modes for a CH_4 molecule.

2.1.3 The Raman Signal and Instrumentation

As mentioned in Subsection 2.1.2, the Raman scattering of a molecule will generate a number of peaks based on its Raman-active vibrational modes. As Rayleigh scattering occurs in around 1/1000 incident

photons, and Raman scattering occurs roughly one thousand times less often than that, the intensity of the inelastically-scattered peaks is swamped out by the former effect [35, 36]. A standard method to exclude the elastic peaks, which exist around $0\,\mathrm{cm}^{-1}$ on the wavenumber axis for Raman spectra, is filtering, shown in Figure 2.3. Notch or low-pass filters are commonly used filters for this purpose.

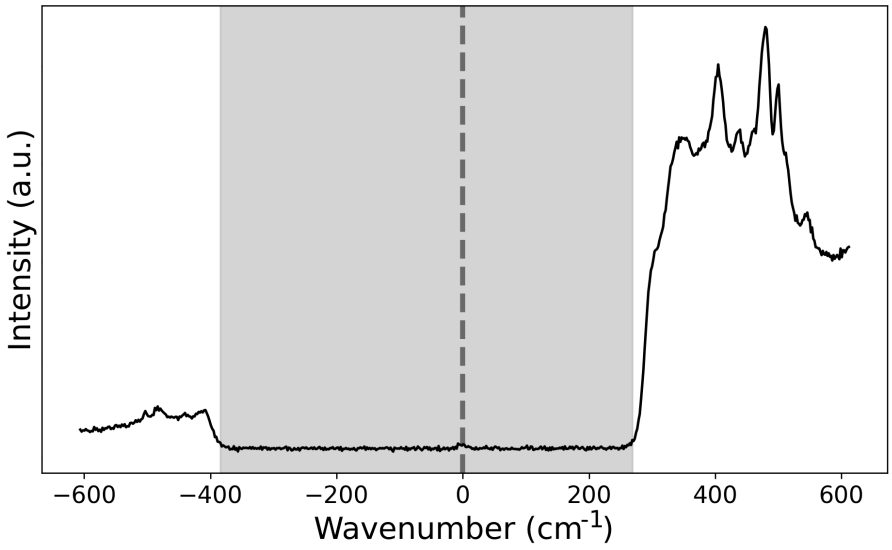


Figure 2.3: Notch filter, indicated by the shaded region, applied to an arbitrary Raman spectrum to filter the elastically-scattered signal. The remaining variance seen in the filtered region is attributed to signal contributions from the dark current (background noise).

Another effect that can worsen the SNR of a Raman spectrum is fluorescence, which can occur when the wavelength of incident light is equal to the wavelength of an energy level transition to a higher electronic state. This causes an emission back to the ground state after an extended time period in comparison to spontaneous Raman scattering - in the order of nanoseconds for fluorescence in comparison to the picosecond timescale of Raman scattering [95] - or through an initial relaxation to a lower energy level in the excited state, before the subsequent emission to the ground state. Based on the difference in timescale between fluorescence and Raman scattering, it is possible to improve the SNR of a Raman spectrum through the used of a pulsed laser probe to illuminate a sample, rather than a continuous-wave Raman laser, if the pulse is shorter than that of the fluorescence effect and if the data is collected within each pulse window [95]. As opposed to Raman scattering, fluorescence is wavelength dependent. It is therefore possible to tune the wavelength of the laser used in a Raman spectrometer in order to shift the fluorescence profile in wavenumber space away from peaks of interest in a measured analyte, thus reducing the effect on the acquired signal, shown in Figure 2.4.

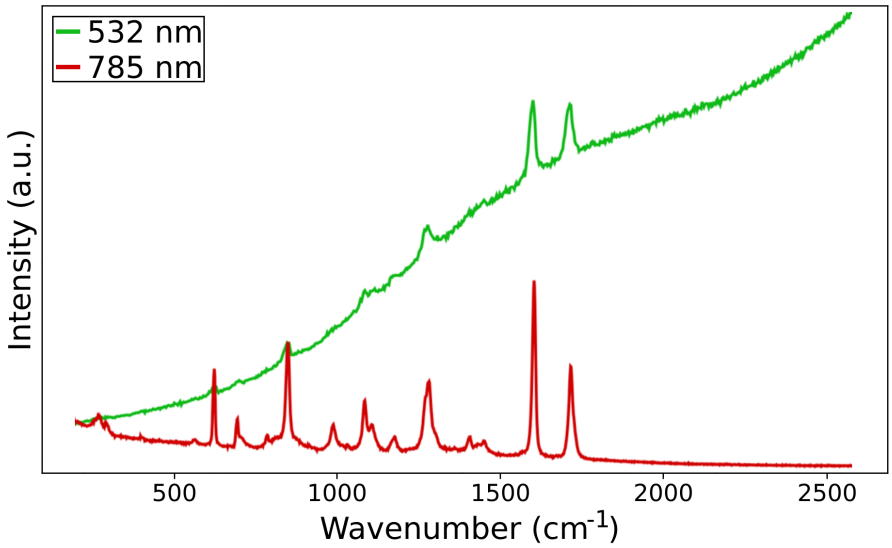


Figure 2.4: Sample measured at two different laser wavelengths demonstrating the shifting of the fluorescence profile of a sample, whilst the wavelength-independent Raman signal remains stationary. Figure adapted from Edinburgh Instruments Ltd. (2022) [138].

Raman scattered light is collected by a detector in the Raman spectrometer. There are two main types of detector: dispersive or FT. In a dispersive spectrometer, laser light scattered from a sample enters through a slit into a chamber where the light is collimated by a mirror, then resolved into individual wavelengths (or equivalently wavenumbers) by a single diffraction grating. A focusing mirror then directs the spectrally-resolved light towards a charge-coupled device (CCD) to collect the spectrum. An example dispersive system is shown in Figure 2.5.

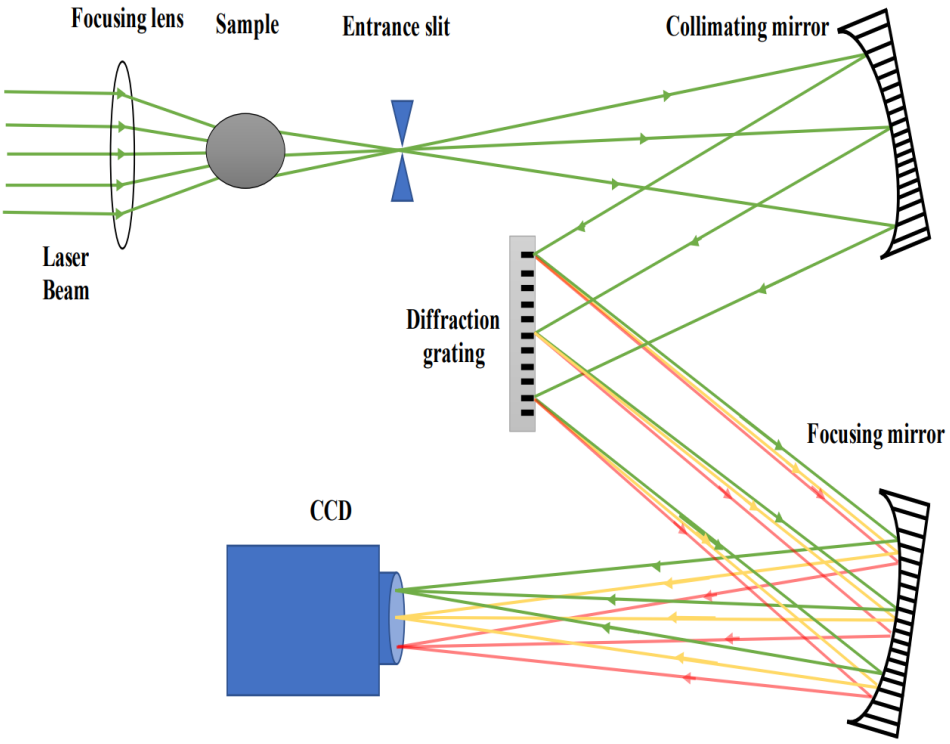


Figure 2.5: Schematic diagram of a dispersive spectrometer. This particular layout is a Czerny-Turner 'W' configuration. Image taken from Naeem *et al.* [139].

FT spectrometers use one diffraction grating in each arm of the spectrometer, this causes two interfering wavefronts to produce an interferogram of the Raman signal that is detected by the CCD, as seen in the spatial heterodyne spectrometer (SHS) configuration of Figure 2.6. The spectrum is recovered from a FT spectrometer by taking the fast Fourier transform (FFT) of the interferogram, alongside other potential preprocessing tools such as apodisation, phase removal and zero-padding.

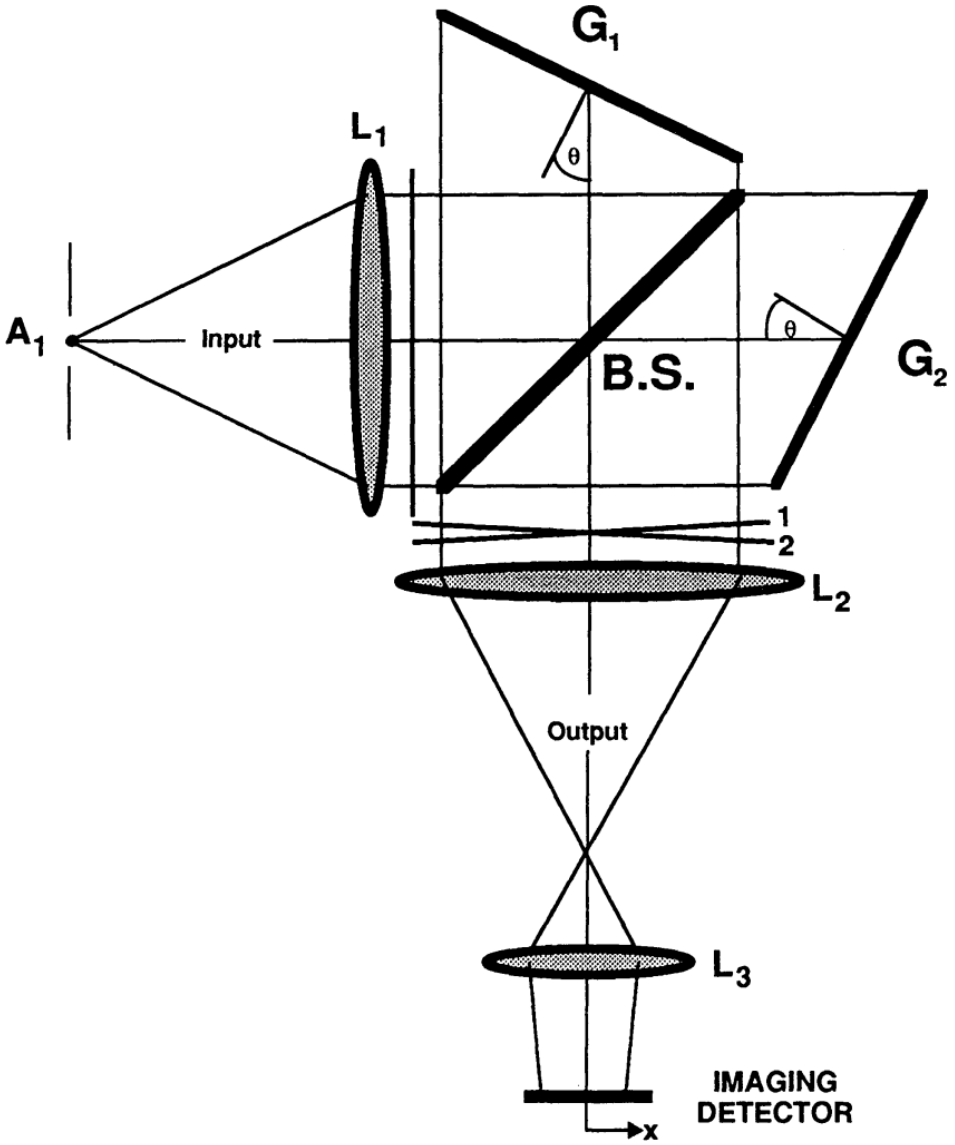


Figure 2.6: Schematic diagram of an SHS configuration. Image taken from Harlander *et al.* [140].

The noise in a dispersive system is non-uniform, as it scales proportionally to the square root of the signal in each CCD bin, by the nature each wavenumber being spectrally separated by the single diffraction grating. Conversely, the noise in an FT system is uniform, as the spectrum is recovered by an FFT of the resulting interferogram, which distributes the same level of noise amongst each pixel in the interferogram. This distinction influences the noise models used in the various data augmentation processes seen throughout this thesis.

2.2 Surface-Enhanced Raman Spectroscopy

The Raman scattering cross-section, which is the ratio of scattered Raman photons to the intensity of incident light, is very small, with typical values $\ll 10^{-28}\,\mathrm{m}^2$ being measured [141]. In order to increase the Raman-scattered signal, the SERS technique can be used, which is able to probe light-induced interactions on the molecular scale due to enhancement factors in the range 10^{10} to 10^{12} [67, 68, 69], with some papers quoting enhancement factors as high as 10^{14} [142, 143]. The mechanism that drives the enhancement of the Raman scattering cross-section is still a matter of debate [77], with two main theories about its origin: the electromagnetic (EM) theory, and the chemical theory, though the former is more commonly stated.

The electromagnetic theory suggests that the boost to the Raman signal of a molecule is due to an electric field enhancement by a metal surface that it is contacting. Via the use of a roughened metal surface or nanoparticle arrangement, local surface plasmons can oscillate orthogonally to the metal surface. Such a configuration creates a plasmonic field that facilitates enhanced Raman scattering events. This field enhances both the incident and Raman scattered light, giving rise to an enhancement factor, Ω_{SERS}, which is proportional to the 4^{th} power of the incident field strength [78, 79, 80]:

$$\Omega_{SERS} \propto I_{inc} \times I_{Raman} \propto E_{inc}^2 \times E_{Raman}^2 \approx E_{inc}^4, \tag{2.2}$$

Where I_{inc} and I_{Raman} are the incident and Raman-scattered intensities, respectively, and E_{inc} and E_{Raman} are the corresponding field strengths.

The chemical theory states that the transfer of charges in resonant conditions provides an enhancement factor of approximately 10^2 [80], depending on the analyte molecule and contacting surface [82], and thus it is thought to contribute to the overall enhancement factor in tandem with the EM effect.

Standard light-induced SERS interactions, as described, contain homogeneous fields that interact with an analyte molecule. One type of SERS system is the nanoparticle-on-mirror (NPoM) geometry [144] (see Figure 2.7), which has three main components (from the bottom-up): A flat metal film, typically gold; a self-assembled monolayer (SAM) of an analyte molecule; and metal nanoparticle spheres deposited onto the SAM surface, which are also usually made of gold. Strong optical fields are able to be generated by this geometry as a consequence of local surface plasmons in the nanoparticle coupling to image charges within the gold film - the method of replacing an object (the gold film) with an imaginary charge - which forms a plasmonic mode that tightly confines light to the analyte molecule situated between the metal-metal gap.

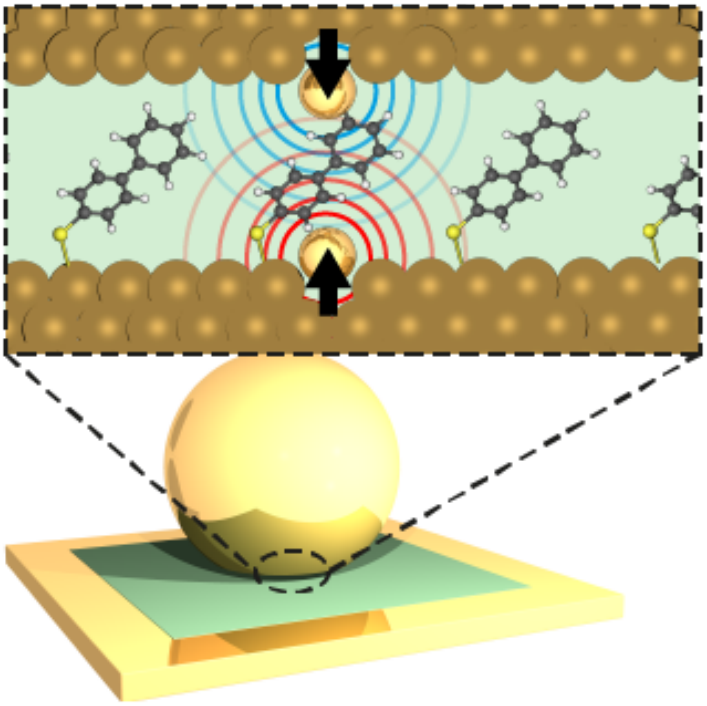


Figure 2.7: Scheme depicting a NPoM geometry containing an arbitrary spacer molecule placed between a metal surface and metal nanoparticle. The inset figure illustrates atomic-scale protrusions in the nanogap.

Homogeneous SERS interactions within NPoM geometries are termed 'nanocavities'. A further amplification to the plasmonic field is possible through the irradiation of gold with a laser, which causes the spontaneous formation of a single, transient gold adatom on either the surface of a nanoparticle or substrate film [75, 145] - using the NPoM geometry as an example. This atomic-scale feature situated within the plasmonic 'nanogap', which are crevices between metal nanostructures [75, 146, 147], termed a 'picocavity', causes an additional ∼10-100x optical field enhancement effect on an analyte molecule [73, 74, 75, 146]. This alters the selection rules that govern visible vibrational modes in Raman spectroscopy, causing previously unseen Raman-inactive IR modes to present in Raman spectra [75, 148]. This is due to the strong local inhomogeneous gradient field, which is produced by the under-coordinated adatom interacting more strongly with bonds in the molecular structure that are in a closer proximity to those that are further away, which in-turn morphs the previously unperturbed electron density. Such a perturbation can shift Raman peaks away from stable wavenumber positions [73, 149].

2.3 Machine Learning

Machine learning is a field of technology that is a subset of artificial intelligence, it allows for machines to learn from data to fulfil inference tasks and gradually self-improve. Machine learning offers myriad opportunities to improve processes across diverse domains, spanning from scientific and financial sectors, and even to applications in computer games [150]. Where once a human or standard computer program would be used to carry out a task such as classification or regression, the development of a replacement, in the form of machine learning algorithms, has been necessitated by the ever-increasing amount of available data required to be processed. This is further emphasised by the increase in computational power in recent years. Machine learning algorithms are capable of improving model performance by assessing output predictions or decisions, in contrast to standard computer programs which lack this crucial aspect. This strategy employed by machine learning is termed 'learning from examples', and it is capable of learning previously unknown patterns in the data it is provided, without being explicitly programmed to do so.

Machine learning can be broken down into two broad categories: supervised learning and unsupervised learning, which are differentiated based on dataset knowledge that an algorithm has access to. Supervised learning relates to tasks where both the input and output data are used when training a model, ergo the data structure is already known. The input data is the information processed by the algorithm, such as sets of Raman spectra, whereas the output data can be labels specifying a particular class, or a value indicating the quantity of the associated data sample. The goal of supervised learning, once a model is trained, is to assign previously unseen data to the correct class label in a classification task, or to make accurate predictions in a regression task. In contrast, unsupervised learning does not involve the use of output data, hence the goal of such a task is to learn patterns and structures that are conducive to both accurate and precise representations of an unlabelled dataset.

All machine learning systems usually divide the input data into multiple datasets. The model is fit to a data partition named the training dataset, and the remaining two partitions are used as inference datasets to evaluate model performance, which are given the names validation and testing. After each iteration of fitting the model to the training dataset, the validation dataset is used to provide an unbiased evaluation of the model, which is used to tune any relevant hyperparameters. Once the model is trained, the testing dataset is used to provide a final unbiased evaluation of the model using data that has neither been used to train the model, nor used to influence any adjustments to model hyperparameters. Although there are numerous machine learning systems, such as decision trees, instance-based learning, support vector machines, and ANNs, this section will only be focusing on the ANN and its variants.

2.3.1 Artificial Neural Networks and Variants

An ANN is a machine learning system that is capable of solving complex problems without user influence, despite the relative simplicity of the algorithm. It is loosely inspired by biological neural networks that compose the human brain. There are three main components that constitute an ANN: nodes, which store a real number called an 'activation', where larger values are said to be 'more activated' than lower values; connections, which are weights linking two nodes that can be tuned during the training process; and activation functions, which map the output of one node into the input of another. Nodes are assembled into a multi-layered structure called a neural network, where a layer is a structure that receives information from previous layers, processes it in some fashion, and then outputs it to the next layer - the processing of any one node in a layer is typically independent of other nodes within that layer. There are three broad categories for layers: an input layer, which receives external data - one data point per node - that is visible to the user; an output layer, also visible to the user, which receives information from the previous layer and outputs the final result of the network; and a hidden layer, of which an arbitrary number of these can exist between the input and output layers, and are named as such due to being naturally hidden from the end user of the neural network. The function of the neural network determines the purpose of each hidden layer, a number of examples are presented in this section, for example: FC layers, convolutional layers, and activation layers. It is common to use the term 'blocks' to group a number of layers or entire neural networks together, which is a useful level of abstraction to describe complex code succinctly.

Neural networks that contain three or more layers are categorised as deep learning, in part for the ability to learn complex, non-linear features. The input of each node is calculated by the sum of all incoming nodes in the previous layer multiplied by respective weighted connections. In addition, a bias term is added to the equation that has a purpose of shifting the value required for a node to become active. These nodes, weights, and biases can produce any arbitrary value, hence an activation function is used to project the value to a point between a finite range, which serves the purpose of enabling the network to learn complex features. The basic equation for calculating the output of a node is

$$\mathrm{A}^{\mathrm{n}} = \sigma(\mathrm{Z}^{\mathrm{n}}) = \sigma(\mathrm{W}^{\mathrm{n}}\mathrm{A}^{\mathrm{n}-1} + \mathrm{b}^{\mathrm{n}}), \tag{2.3}$$

where $\mathrm{A}^{\mathrm{n}-1}$ and A^{n} are the vectorised forms of the node outputs in the previous and current layers, respectively, Z^{n} is the raw node value before the activation function, W^{n} is the weight matrix containing the values of all weighted connections between the two layers, b^{n} is the bias vector for the current layer, and σ is an arbitrary activation function. It is common for the weight parameters to be randomly initialised using a Glorot uniform distribution [151], also called a Xavier uniform distribution, which draws samples from a uniform distribution within a limit based on the combined number of input and output units for the current layer. This random initialisation breaks the symmetry caused by weight initialisation by a constant value, which would cause each node to receive the same signal. Bias

parameters, however, are commonly initialised with a zero constant [152].

ANNs can be further divided into two categories: feedforwards networks and recurrent networks. Feedforwards networks involve connections between nodes that do not loop back to nodes in previous layers, thus information only flows forwards from the input layer, through to each hidden layer, and then to the output layer. This means that no past information influences the outcome of a node processing present data until a model output is produced for that iteration, hence feedforwards networks are commonly used to learn relationships between independent input variables, and dependent output variables. Recurrent networks involve connections that do form loops. Consequently, information processed in later layers can influence the outcome of earlier layers, hence this type of network is typically used to make predictions on sequential or time series data. As the research conducted in this thesis relates to the former category, the latter will not be expanded upon further. Figure 2.8 shows an example feedforwards neural network with three layers called a multi-layer perceptron (MLP), these standard layers are called FC layers.

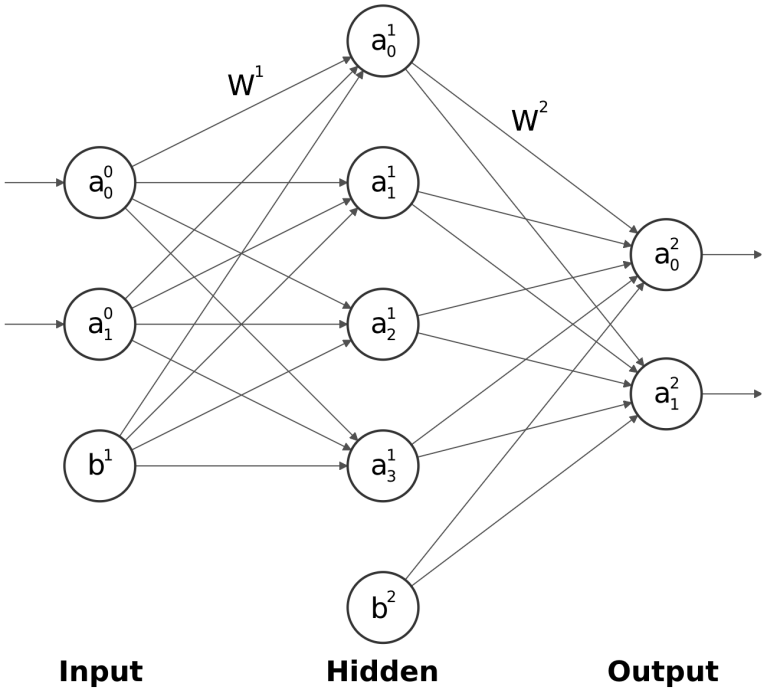


Figure 2.8: Example feedforwards ANN, where a_k^n represents the value of the k^{th} node in the n^{th} layer, W^n represents the weight matrix for layer n, and b^n represents the bias vector for layer n. Lastly, an activation function (not shown) is used to scale the output of each node, where different activation functions can be used both within the same or different layers.

In order to explain the concept of increasing levels of abstraction within deeper layers of a neural

network, it will be beneficial to first introduce a variant of an ANN: the CNN. This type of neural network is commonly used to learn and recognise patterns in image data. Where a standard MLP uses only FC layers, a CNN replaces some or all of these layers with convolutional layers. Each convolutional layer contains multiple filters, also called feature maps, which are two-dimensional collections of nodes (in the case of image data) that receive information from a set of convolutions performed on a receptive field - a portion of the entire field of view of an image - by a kernel operating on the previous layer. An example convolution operation is shown in Figure 2.9.

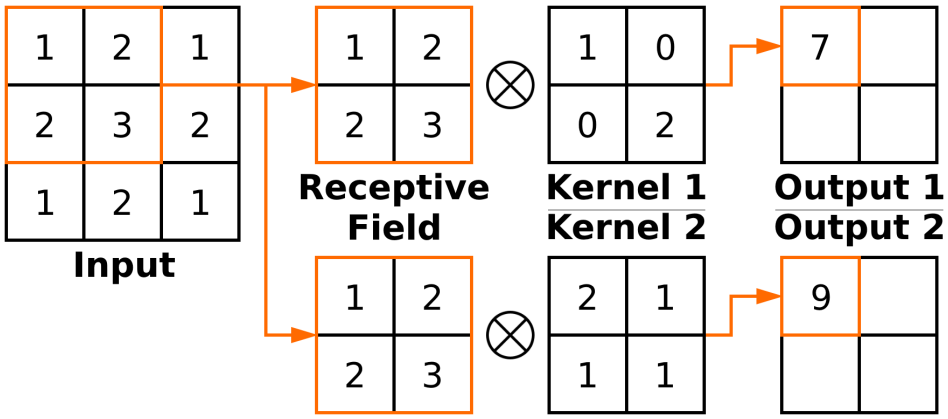


Figure 2.9: Example convolution operation in 2D with two filter kernels.

A more general case than Figure 2.9 involves multiple filters in the input layer, in which kernels in the output layer have a number of filters equal to the number of filters in the input layer, and each corresponding node in an output filter is the result of the element-wise sum of the three-dimensional receptive field - i.e. (width, height, #filters). The shape of the output along each dimension - except the output filter dimension, which is arbitrary - is given by the receptive field formula,

$$m_{out} = \left\lfloor \frac{m_{in} - 2p + k}{s} \right\rfloor + 1 \tag{2.4}$$

where m_{in} and m_{out} are the input and output feature sizes, respectively, p is the amount of zero padding added to the end of the feature dimension, k is the kernel size, and s is the stride size, which specifies the number of pixels shifted over the input array between each receptive field.

Convolutional layers learn features about the image data, which are stored within each filter, that are typically then downsampled using a pooling layer, shown in Figure 2.10. This pooling operation has the effect of maintaining the larger, more important structural features of a filter, whilst simultaneously removing finer spatial details, which are undesirable as the goal of a CNN is to produce feature maps that are shift-equivariant [153].

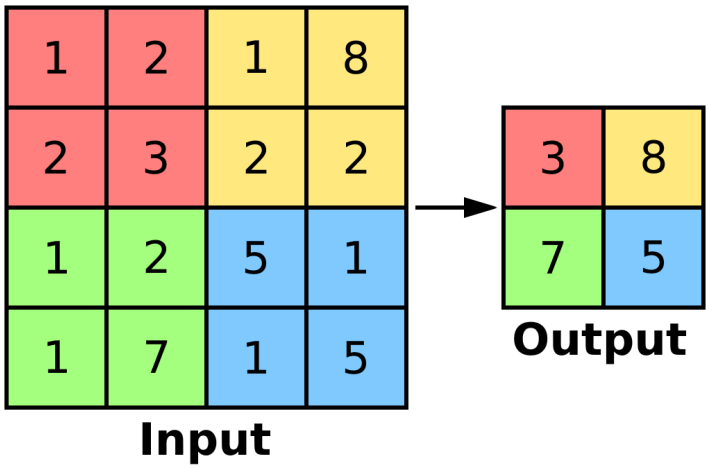


Figure 2.10: Example maxpooling operation using a (2×2) pooling kernel with stride (2×2).

The learned feature maps in shallower layers of a CNN, those closer to the input layer, contain lower-level features such as lines, edges, and basic shapes; feature maps in deeper layers can learn higher-level features that are incredibly specific to a particular class within a dataset, such as cars, buildings, and faces. This is due to the hierarchical decomposition of the input data, where the earliest convolutional layer operates on the raw input values themselves, and later convolutional layers operate on the output of previous convolutional layers, hence the concept of multiple convolutional layers learning increasing levels of abstraction about the input data. Figure 2.11 showcases an example.

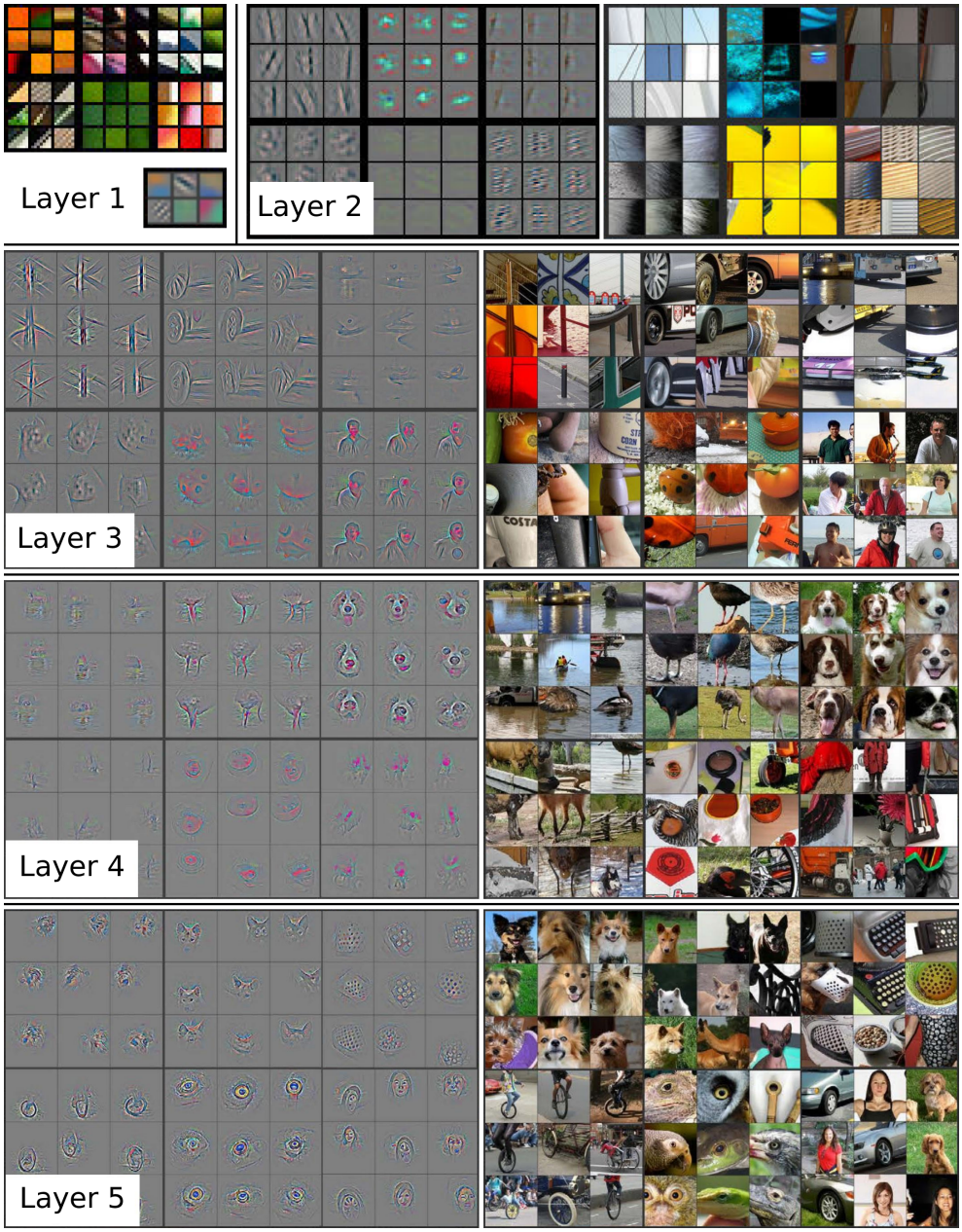


Figure 2.11: Feature map visualisation of a CNN trained on the ImageNet database. Figure adapted from Matthew D. Zeiler and Rob Fergus, 2014 [154]. The top-9 activations are shown for random filters in layers 2-5; *left*, reconstructed patterns that cause the highest activations in each filter; *right*, corresponding images. For layers 1-5, the learned features increase in levels of abstraction: edges, textures, patterns, parts, and objects. For example, one filter learned to detect dog noses in an image (layer 4, row 1, column 3), which highlights that these filters learn specific discriminative features.

With regards to the method allowing an ANN to learn, the backpropagation algorithm is employed alongside a gradient descent optimisation algorithm. Backpropagation is the process of calculating the gradient of a loss function - which is used to evaluate the performance of a model, expanded upon in Subsection 2.3.2 - with respect to all trainable parameters within a model, which are the weights and biases within each layer. This algorithm involves recursive applications of the chain rule, which calculates the gradients of the latest layers through to the earliest. This is given by the following example equations used to calculate the gradient for the weights in the last layer of a supervised neural network, which uses an arbitrary loss function, L, based on vectorised target values, Y:

$$\frac{\partial \mathrm{L}}{\partial \mathrm{W}^{\mathrm{n}}} = \frac{\partial \mathrm{Z}^{\mathrm{n}}}{\partial \mathrm{W}^{\mathrm{n}}} \frac{\partial \mathrm{A}^{\mathrm{n}}}{\partial \mathrm{Z}^{\mathrm{n}}} \frac{\partial \mathrm{L}}{\partial \mathrm{A}^{\mathrm{n}}} \tag{2.5}$$

$$\frac{\partial \mathrm{Z}^{\mathrm{n}}}{\partial \mathrm{W}^{\mathrm{n}}} = \mathrm{A}^{\mathrm{n}-1} \tag{2.6}$$

$$\frac{\partial \mathrm{A}^{\mathrm{n}}}{\partial \mathrm{Z}^{\mathrm{n}}} = \sigma'(\mathrm{Z}^{\mathrm{n}}) \tag{2.7}$$

$$\frac{\partial \mathrm{L}}{\partial \mathrm{A}^{\mathrm{n}}} = \mathrm{L}'(\mathrm{A}^{\mathrm{n}}, \mathrm{Y}), \tag{2.8}$$

where Equations 2.6 - 2.8 are the components on the RHS of Equation 2.5, whose derivatives are calculated from Equation 2.3. Similarly, the equations used to calculate the gradient of the biases in the last layer of the same network are much the same as before, but with alterations to Equations 2.5 and 2.6, as given:

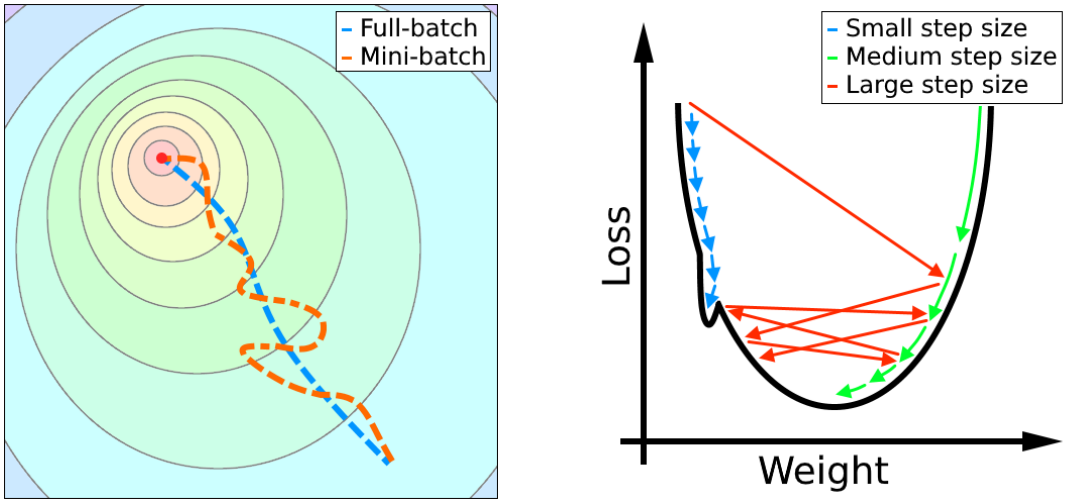
$$\frac{\partial \mathrm{L}}{\partial \mathrm{b}^{\mathrm{n}}} = \frac{\partial \mathrm{Z}^{\mathrm{n}}}{\partial \mathrm{b}^{\mathrm{n}}} \frac{\partial \mathrm{A}^{\mathrm{n}}}{\partial \mathrm{Z}^{\mathrm{n}}} \frac{\partial \mathrm{L}}{\partial \mathrm{A}^{\mathrm{n}}} \tag{2.9}$$

$$\frac{\partial \mathrm{Z}^{\mathrm{n}}}{\partial \mathrm{b}^{\mathrm{n}}} = 1. \tag{2.10}$$

Once the gradients have been calculated, a gradient descent optimisation algorithm is then used to update the trainable parameters. A standard form of this optimisation algorithm is stochastic gradient descent (SGD), which is used to minimise the loss function of a predictive model, such as a model used for either a classification or regression task, based on examples from a training dataset. The gradient descent process is referred to as stochastic as the process of updating the trainable parameters is performed using batches, also known as mini-batches, of the training dataset. Batches of data are used in place of 'full-batch' gradient descent - using the entire training dataset per batch - and fully-stochastic gradient descent - using single samples per batch - for two reasons: the first being that calculating the gradients using full-batch gradient descent is typically infeasible due to both memory restrictions and long computation times; and the second being that updating model parameters based

on fully-stochastic gradient descent both promotes overfitting of the model to those samples, thus making it harder for the training process to produce a generalised model, and negates the advantage of vectorisation decreasing computation times.

The difference between full-batch and mini-batch gradient descent is illustrated in Figure 2.12a. One important point to note with regards to the calculated gradient is that it points 'uphill', therefore the negative of the gradient is used to update the parameters in order to advance towards a local minimum of the loss function. As the complete parameter space can involve millions of parameters that require optimising for a given task, updating the parameters each iteration by the complete calculated gradient is likely to cause drastic changes to the model, thus making it extremely difficult to train. The solution then is to instead update the parameters by only a fraction of the calculated gradients - this fraction is given the names of step size or learning rate. This can cause the parameters to converge to an optimal local minima, depending on the magnitude of the learning rate, as illustrated in Figure 2.12b.



(a) Difference between full-batch and mini-batch gradient descent for a contour plot of an arbitrary loss function. Full-batch is akin to a slow, steady shift to a local minima (red dot), whereas mini-batch is akin to a faster, albeit noisier approach.

(b) Effect of step size, also called learning rate, on the optimisation of a single arbitrary weight parameter. Larger step sizes may be unable to converge, or even diverge, whilst smaller step sizes might converge to a sub-optimal local minima.

Figure 2.12: Illustrations of two main variables used in updating parameters of a neural network, which is trained using SGD with backpropagation: **a** batch size, and **b** step size.

The process of performing SGD with backpropagation is repeated every epoch. An epoch is defined as the number of times a learning algorithm will update the parameters of a model based on complete passes of the entire training dataset. As the training dataset is split into batches, each epoch can

be further divided into steps, where one step processes a single batch. Determining the appropriate number of epochs to train a model for is a challenging process due to the large number of trainable parameters, and so it is typical for a stopping criterion to be specified in a number of ways, such as: an arbitrary number of epochs; a threshold for the loss function; or an 'early stopping' procedure, which stops a network from training once overfitting occurs to the training dataset, and restores the model parameters back to the previous best epoch based on validation data losses.

Now, considering the definitions of gradient descent and backpropagation, it is useful to introduce the concepts of 'forwards passes' and 'backwards passes' that appear as common nomenclature in literature [155]. In a forwards pass, the nodes within each layer of a neural network have values determined by the inputs received from earlier layers, based on the current input data, model parameters, and activation functions, whereas a backwards pass propagates the calculated derivatives of each trainable parameter, with respect to the loss function, from the latest layers through to the earliest.

2.3.2 Hyperparameters and Optimisation Tools

As mentioned so far in Section 2.3, model hyperparameters are adjusted based on a mixture of dataset knowledge, and an evaluation how a model performances on a validation dataset. Where parameters are defined as values derived from training - such as the weights and biases within each trainable layer - hyperparameters are defined as variables that are set prior to training, which typically remain unaltered as the model learns, and whose values dictate the quality and computational speed of the training task. A number of hyperparameters have already been mentioned: learning rate, batch size, pooling, activation functions, and optimisation algorithms - the last two of which will be expanded upon further. Defined below are additional hyperparameters relevant to this thesis, which can be used to optimise and regularise (i.e. prevent the overfitting of) model performance.

Loss Functions. Also called objective functions, these are the metrics used to evaluate the performance of a model in the ability to: correctly predict the class label for a given sample in a classification task, accurately estimate a quantity in the case of regression, or some other relevant metric in the case of unsupervised learning. Two examples of loss functions used in regression tasks are: MSE, which penalises errors in model estimations by the square of the difference between true and predicted values, using the equation

$$\mathrm{L} = \frac{1}{\mathrm{N}} \sum_{\mathrm{i}=1}^{\mathrm{N}} (\mathrm{y_i} - \tilde{\mathrm{y}}_\mathrm{i})^2, \tag{2.11}$$

where $\mathrm{y_i}$ and $\tilde{\mathrm{y}}_\mathrm{i}$ are the true and predicted labels for a sample i, respectively, averaged over N samples; and mean squared logarithmic error (MSLE), which penalises errors in model estimations by a percentual difference, hence this loss function may be preferable when small differences between smaller

quantities are needed to be to equal in influence as large differences between larger quantities. MSLE is given by the equation

$$L = \frac{1}{N}\sum_{i=1}^{N}(\log(y_i + 1) - \log(\tilde{y}_i + 1))^2. \tag{2.12}$$

Other variants of loss functions for regression may incorporate the square root of the aforementioned loss functions, such as root mean squared error (RMSE), which may be used as it is easier to interpret when evaluating a model, for example if the quantity estimated is distance in metres, then a loss value produced by RMSE will have units of metres, whereas MSE will have units of metres squared.

For binary classification tasks the binary cross-entropy (BCE) loss function is used, which compares the predicted class labels as probabilities to the true class labels, and is given by the equation

$$L = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(q(\tilde{y}_i)) + (1 - y_i)\log(1 - q(\tilde{y}_i))\right], \tag{2.13}$$

where y_i are the true class labels, and $\tilde{y}_i$ are the predicted class labels that are converted to estimated probabilities, $q(\tilde{y}_i)$. For data containing multiple classes, a more general form of BCE is used: categorical cross-entropy, which is given by the equation

$$L = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{K} y_{ij}\log(q(\tilde{y}_{ij})), \tag{2.14}$$

where i indexes each of the N samples, and j indexes each of the K classes.

Optimisation Algorithms. As mentioned in Subsection 2.3.1, an optimisation algorithm is used to update the trainable parameters of a model during the training process, with a typical goal of minimising an objective function such as MSE loss. One addition to standard SGD is 'Momentum', which updates model parameters by computing the exponentially weighted averages of all previous gradients, rather than solely the current epoch. This optimisation algorithm generally decreases the number of epochs required for convergence to a local minimum [155]. The 'SGD with Momentum' algorithm uses an additional hyperparameter β_ν, which controls the weighting of previous gradients, as given by the equations

$$\nu_{dw} \coloneqq \beta_\nu \nu_{dw} + (1 - \beta_\nu)dw \tag{2.15}$$

$$w \coloneqq w - \alpha\nu_{dw}, \tag{2.16}$$

where ν_{dw} on both sides of Equation 2.15 are the values of the current and previous weighted gradients

respectively (initialised to zero), dw is the derivative of the weight parameter, and α is the learning rate. Default SGD with Momentum uses $\beta_\nu \geq 0.9$ [156]. In contrast, standard SGD can be seen as a special case of SGD with Momentum where $\beta_\nu = 0$. The example given is for a single weight parameter in an arbitrary layer, which can be easily vectorised to act on all weights within a single layer by broadcasting the value of β_ν. The equations for updating the bias parameters can be trivially obtained by replacing the weight variables with bias variables in Equations 2.15 and 2.16. As mentioned, ν_{dw} is initialised to zero, this incurs a bias (not to be confused with a bias parameter) that reduces the magnitude of the gradient updates for the earliest epochs, hence a bias correction term can also be introduced to Equation 2.15 that scales each gradient update using the equation

$$\nu_{dw}^{corr} \coloneqq \left[\beta_\nu \nu_{dw}^{corr} + (1 - \beta_\nu)dw\right] \frac{1}{1 - \beta_\nu^t}, \tag{2.17}$$

where ν_{dw}^{corr} on both sides of the equation are the corrected forms of ν_{dw} for the current and previous epoch, respectively, and t is the current epoch. This removes the aforementioned bias to the gradient, whilst having a negligible effect in later epochs as the β_ν^t term tends towards zero.

Another optimisation algorithm used to increase the speed of convergence is the root mean square propagation (RMSprop) algorithm [157, 158]. Similarly to SGD with Momentum, these gradient updates use exponentially weighted averages of gradient values from previous epochs, as described by

$$S_{dw} \coloneqq \beta_S S_{dw} + (1 - \beta_S)dw^2 \tag{2.18}$$

$$w \coloneqq w - \alpha \frac{dw}{\sqrt{S_{dw} + \epsilon}}, \tag{2.19}$$

where S_{dw} and β_S are equivalent to ν_{dw} and β_ν, respectively, dw^2 is the element-wise square of the weight derivative, and ϵ is a term used for numerical stability with a typical value of $\epsilon = 10^{-8}$. The bias parameter equations can be trivially obtained as before. The benefit of squaring the derivative of the parameter is that larger oscillations in the values over each epoch are more greatly dampened - this culminates in a steadier path towards a local minimum during gradient descent such as what is illustrated in Figure 2.12a - which is due to the square root term in Equation 2.19. The faster convergence caused by the dampening effect also allows for larger learning rates to be used.

One final optimisation algorithm that is relevant to this thesis is the Adam optimisation algorithm [159], which combines the benefits of SGD with Momentum and RMSprop, as described by

$$w \coloneqq w - \alpha \frac{\nu_{dw}^{corr}}{\sqrt{S_{dw}^{corr} + \epsilon}}, \tag{2.20}$$

where ν_{dw}^{corr} is the bias corrected form of the weighted gradients from SGD with Momentum, given in Equation 2.17, and S_{dw}^{corr} is the bias corrected form of the weighted gradients from RMSprop, which can

be similarly derived by a comparison of Equations 2.17 and 2.18. The bias parameter update equation can be trivially obtained as before. The default values for the four hyperparameters in Adam used in the original paper [159] are $\alpha = 0.001$, $\beta_\nu = 0.9$, $\beta_S = 0.999$, and $\epsilon = 10^{-8}$. It should be noted that the β_ν and β_S parameters are termed β_1 and β_2 within the following chapters of this thesis.

Activation Functions. During the forwards pass in the training of a neural network, activation functions are almost invariably used within each layer. As shown in Equation 2.3, an activation function, denoted as σ, is used to scale the raw output of a node to a point between a finite range of values. Another purpose of an activation function is to introduce non-linearity into the neural network, which allows a model to learn complex, non-linear features about the target dataset, as without this aspect a neural network will degenerate to a linear function regardless of the number of layers. This operation prevents parameters from becoming too large, which makes neural networks harder to train or even diverge. The choice of which activation function to use will depend on the layer that it is applied to, as well as dataset knowledge. For example, a common activation function applied to the output layer of a neural network trained as a binary classifier is the sigmoid function,

$$\sigma(z) = \frac{1}{1 + e^{-z}}, \tag{2.21}$$

where z is the raw node value of the output layer, also called a 'logit', which is defined as the unnormalised prediction generated by a classification model. A more generalised form of the sigmoid function is the softmax function, which is used for multi-class classification, and is given by the equation

$$\sigma(z)_i = \frac{e^{-z_i}}{\sum_{j=1}^{K} e^{-z_j}}, \tag{2.22}$$

where z_i is the logit for an arbitrary class label, i, in a multi-class scenario, and the z_j term on the denominator is the sum of all K logits, with components that sum to one. An alternate version of the sigmoid function is the tanh function; where the sigmoid is an appropriate activation function for the output layer of a binary classification model - as class labels are either zero or one, therefore projecting logits between the same range makes comparisons simpler - the tanh function is better suited for hidden layers, due to the function scaling the raw output values of each node between the range [-1, 1]. This causes a model to learn parameters with a zero-mean, which is a desirable trait for standardised data. The tanh function is defined as

$$\sigma(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}. \tag{2.23}$$

One issue with the sigmoid and tanh functions is that, for values with larger magnitudes, the gradients of both functions tend towards zero, as seen in Figure 2.13, this is known as the vanishing gradient problem, and it has the effect of slowing down or even halting training. To solve this problem

and speed up training time - in terms of the required number of epochs to train a model - the rectified linear unit (ReLU) activation function can be used,

$$\sigma(z) = \max(0,\ z), \tag{2.24}$$

which has a gradient of one for all values of z greater than zero. Although the derivative when z is less than zero is undefined, it is unlikely for a node to meet that criteria - coding libraries that implement the ReLU activation function typically have fail-safes to use a substitute derivative of zero if this situation occurs. Training a model with rectified functions produces parameters with increased sparsity compared to non-rectified functions, which is suggested to improve model performance [160, 161]. The ReLU activation function can be modified to allow for small negative values in the output activation, which prevents the case of zero-gradients from slowing down training in layers with ReLU activation functions - a situation known as the 'dying ReLU problem' - this variant is called the Leaky ReLU activation function [161, 162], and is shown in Figure 2.14. Leaky ReLU introduces the hyperparameter α, which controls the gradient of the function when z is less than zero, as defined by the equation:

$$\sigma(z) = \begin{cases} \alpha z & \text{for} \quad z < 0 \\ z & \text{for} \quad z \geq 0. \end{cases} \tag{2.25}$$

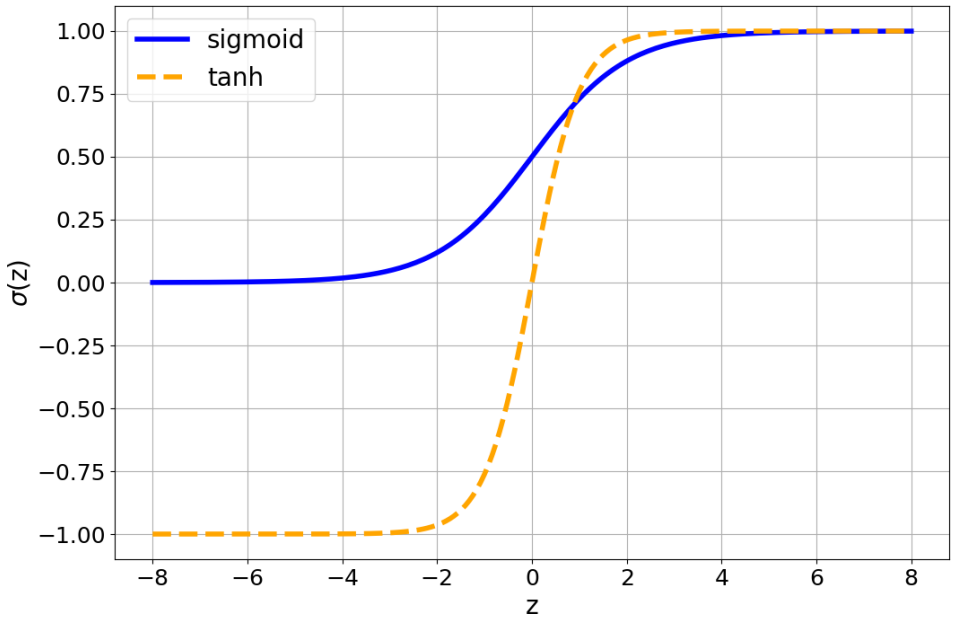


Figure 2.13: The sigmoid and tanh activation functions.

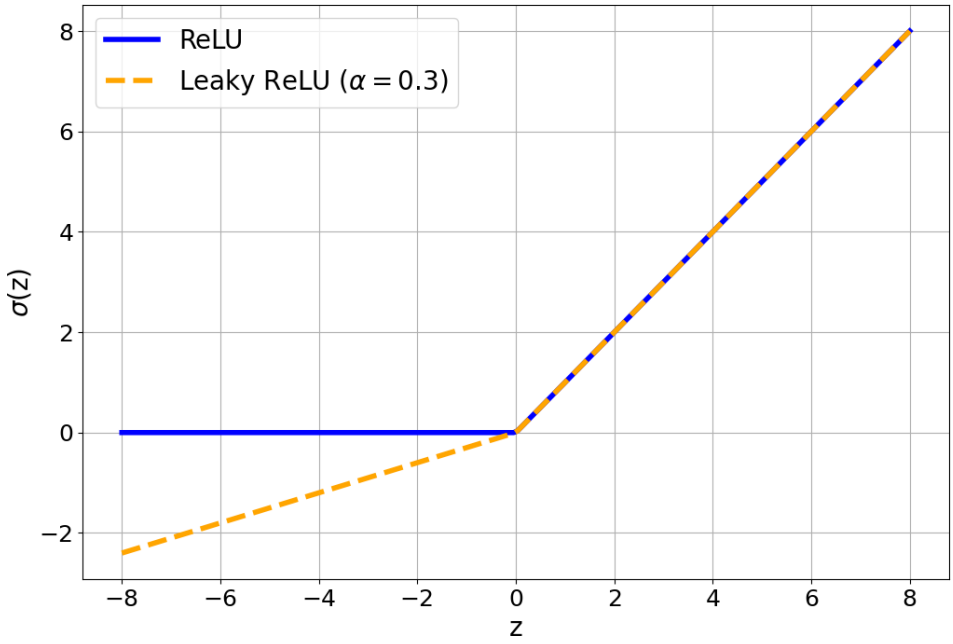


Figure 2.14: The ReLU and Leaky ReLU activation functions.

Normalisation Layers. DNNs suffer from the problem of 'internal covariate shift' during training, in which changes in the distribution of inputs in earlier layers affects the distribution of inputs in later layers. This instability, caused by the sensitivity of a neural network to the random initialisation of model parameters, requires the use of lower learning rates to train a model. A normalisation technique called batch normalisation was developed [163] that is capable of mitigating this problem, which greatly reduces the number of epochs required to train a neural network whilst simultaneously regularising layer activations [164, 165, 166]. Batch normalisation utilises two parameters: γ and β, which are used to scale and shift the input distribution in the layer it is applied to, respectively, as shown:

$$z^{i}_{\text{norm}} = \frac{z^{i} - \mu}{\sqrt{\sigma^2 + \epsilon}} \tag{2.26}$$

$$\tilde{z}^{i} = \gamma z^{i}_{\text{norm}} + \beta, \tag{2.27}$$

where $\tilde{z}^{i}$ is the scaled and shifted version of the standardised input, z^{i}_{norm}, for the i^{th} unit in an arbitrary layer; μ and σ^2 are the mean and variance of the input data, z^{i}, for the current mini-batch; and ϵ is a constant used for numerical stability commonly initialised to 0.001. The γ and β parameters are learnable, meaning that they are updated during the backwards pass of each training step.

In much the same way as normalising input data helps to remove biases from different data ranges,

introducing normalisation to each hidden layer improves the efficiency of training parameters in the succeeding layers. Batch normalisation, as mentioned, is the most common normalisation layer applied to neural networks, which normalises each mini-batch along the batch axis of the data array for whichever layer it is applied to (this technique is visualised in Figure 2.15). Two famous model architectures that make use of batch normalisation to achieve state-of-the-art results are the ResNet [167] and Inception-v3 [164] models. Although batch normalisation can be applied either before or after the activation function, it is standard practise to apply it beforehand. In addition, similar to normalising the input data during preprocessing, normalisation layers use the learned statistics, γ and β, as well as an exponentially weighted average of the mean and variance of the training data to normalise input data samples from the validation and testing datasets.

Alternative normalisation techniques have been developed that improve model performance over batch normalisation for specific tasks. One example is group normalisation [168], which counteracts errors in batch normalisation for small batch sizes by operating on partitions of data typically along the channel axis, and is more suitable to computer vision tasks where small batches may be required due to memory limitations. Another example is instance normalisation [169, 170], which normalises data samples, or instances, independently. This prevents instance-specific statistics within a mini-batch from affecting the contrast information of other instances, whilst maintaining the effects of batch normalisation. Instance normalisation is suitable for style transfer tasks [171], whereby the style of one image (such as a piece of art) is blended with the contents of another. These techniques are shown in Figure 2.15.

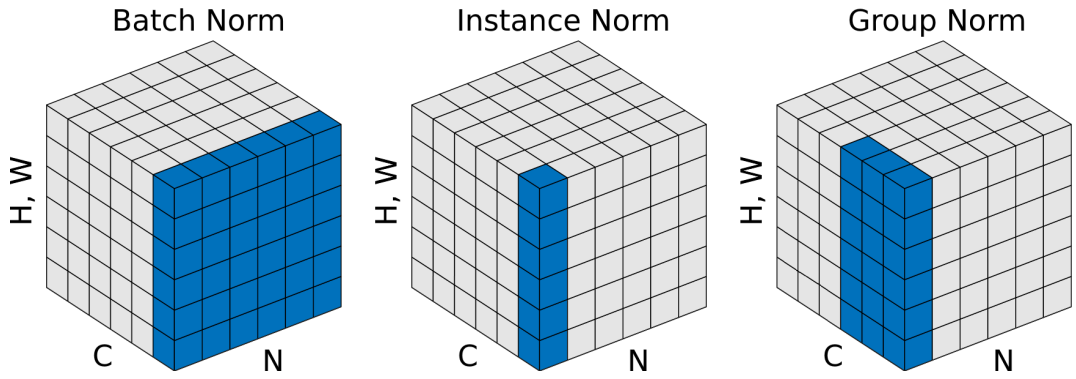
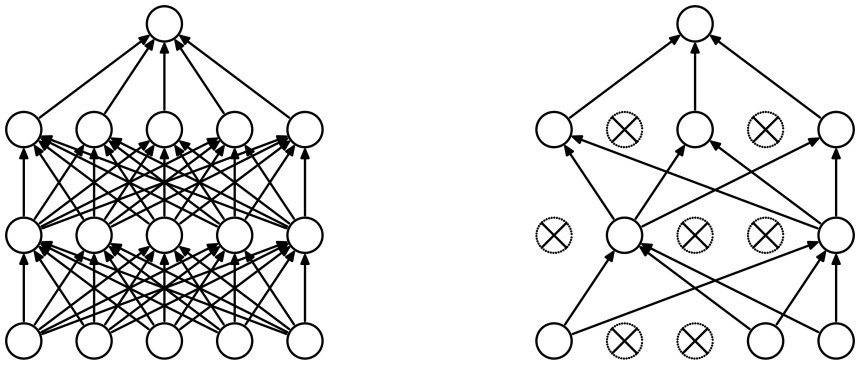


Figure 2.15: Three example normalisation methods applied to a rank-4 tensor. Each method is applied along the axis of the highlighted region, where N is the batch axis, C is the channels axis, and (H, W) are the combined features axes. Figure adapted from Yuxin Wu and Kaiming He [168].

Dropout. Overfitting is a common issue in DNNs, which would in the past be circumvented through an ensemble of large networks with different architectures. Considering the long training times to make use of such a system, a computationally inexpensive technique called dropout was developed to

regularise models [172]. During a training step, for whichever layers dropout is applied to, a fraction - known as the dropout rate - of nodes along with their input and output connections are randomly ignored, or 'dropped'. Layer outputs are then scaled by dividing each activation by this hyperparameter in order to maintain the expected output distribution. This has the effect during a backwards pass of updating a layer using data obtained from a different 'perspective' of that layer during the forwards pass - thus mimicking the effect of training a larger ensemble. During validation and testing, the dropout layers are deactivated, allowing the whole model to be evaluated.



(a) Standard MLP. (b) MLP after applying dropout.

Figure 2.16: Dropout applied to an arbitrary ANN. **a**, neural network with two hidden layers; **b**, network with dropped applied with a rate of 0.5, shown as crossed units. Figure taken from Srivastava *et al.* [172].

Weight Decay. Another tool to prevent overfitting due to high variance parameters is weight decay, which adds a penalty term applied to the weight parameter update step of backpropagation by modifying the loss function. The common type of weight decay is L2-regularisation:

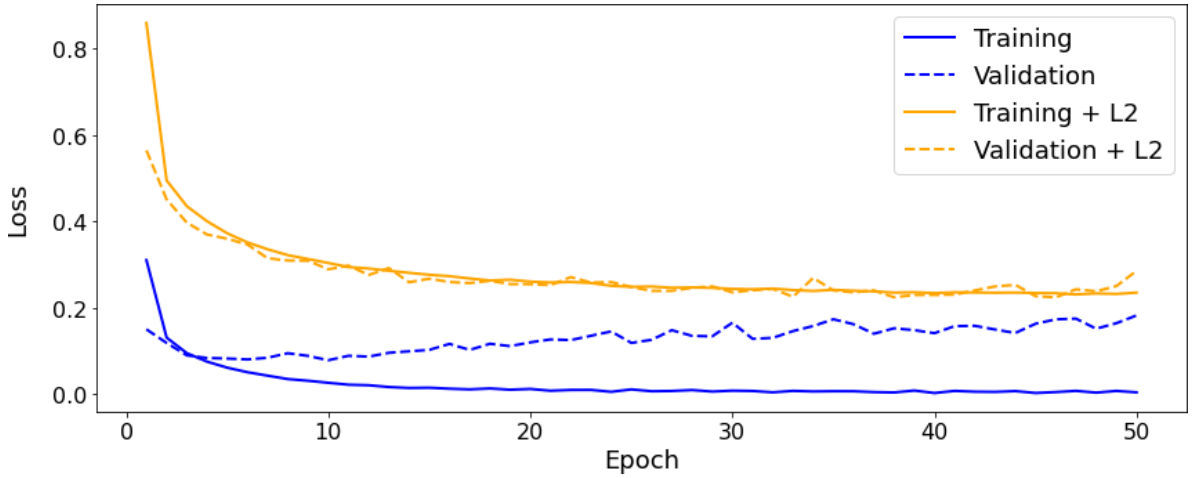
$$\mathrm{J(W,b)} = \frac{1}{\mathrm{N}} \sum_{\mathrm{i}=1}^{\mathrm{N}} \mathrm{L}(\tilde{\mathrm{y}}^{\mathrm{i}}, \mathrm{y}^{\mathrm{i}}) + \frac{\lambda}{2\mathrm{N}} ||\mathrm{W}||_{\mathrm{F}}^{2}, \tag{2.28}$$

where J(W, b) is an arbitrary global loss function over all trainable parameters; $\mathrm{L}(\tilde{\mathrm{y}}^{\mathrm{i}}, \mathrm{y}^{\mathrm{i}})$ is the loss function for each of the N training samples; λ is the regularisation factor, which is a tunable hyperparameter; and $||\cdot||_{\mathrm{F}}^{2}$ is the Frobenius norm, which is the squared Euclidean norm of the weight matrix W. Whilst an additional bias regularisation term is similarly possible, it is rarely implemented as there are abundantly more weight parameters that are sufficient enough to impact the regularisation of the model. Another weight decay method is L1-regularisation, although it is less commonly used, this simply replaces the Frobenius norm of the weight matrix with the L1-norm ($||\cdot||_1$), which produces values for the weights that are more sparse [173, 174]. The derivative of Equation 2.28 is simply the derivative of the loss function - using SGD with Momentum as an example (see Equation 2.15) - plus

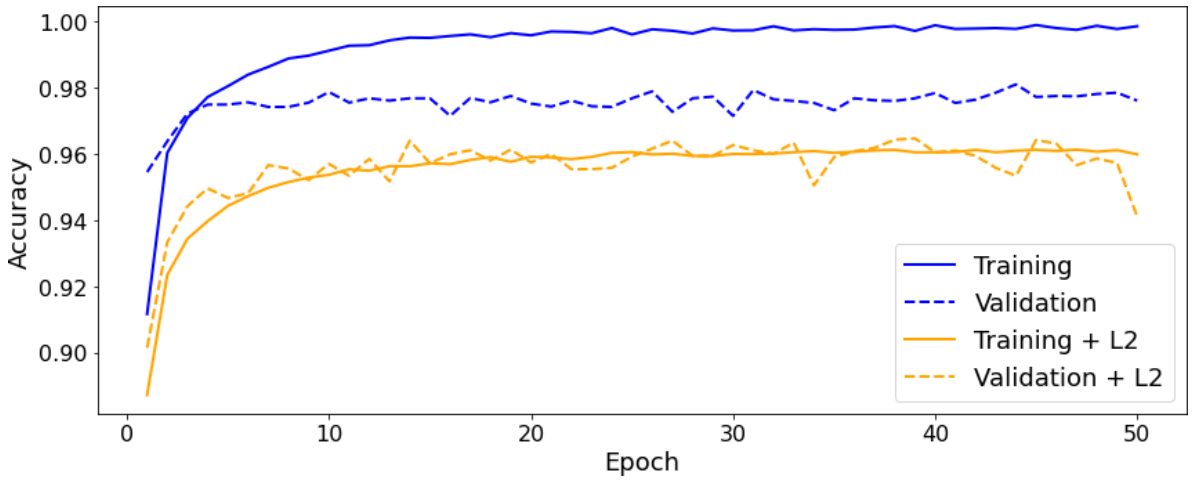
the derivative of the L2-regularisation term, and is defined as

$$\nu_{dw} \coloneqq (\text{arb. backprop.}) + \frac{\lambda}{N} W^n, \tag{2.29}$$

where N is the number of training samples in the current batch, and W^n is the weight matrix for the current layer, n. The corresponding weight update equation used during gradient descent with backpropagation - using the same SGD with Momentum example - is identical to Equation 2.16. An example of the effect of weight decay as a regularisation tool is shown in Figure 2.17, which showcases the training and validation loss curves of two near-identical models - differentiated only by one model making use of L2-regularisation on all hidden layers - which is trained on the MNIST handwritten digits dataset [175] for 50 epochs using a sparse (i.e. a binary label vector with only a single '1') categorical cross-entropy loss function.



(a) Categorical cross-entropy loss curves. Overfitting to the training dataset (blue) causes the validation loss to slowly diverge, in contrast L2-regularisation (orange) has produced a robust, generalised model.



(b) Accuracy curves. Overfitting to the training dataset (blue) causes the validation accuracy to stagnate, whereas the model using L2-regularisation (orange) achieves similar dataset accuracies throughout training.

Figure 2.17: Comparisons of **a** loss and **b** accuracy metrics between two 2-hidden layer ANNs trained to classify handwritten digits from the MNIST database [175]. One model is trained with L2-regularisation applied to both hidden layers with a regularisation factor of 0.01 (orange), and the other features no regularisation (blue). All other hyperparameters are identical. The model with L2-regularisation has removed the overfitting problem, however general performance has decreased, therefore the model parameters and hyperparameters would at this stage require tuning in order to improve classification accuracy.

Gradient Clipping. As previously mentioned in the activation function segment, deep learning models are liable to suffering from the vanishing gradient problem. Another potential issue can arise called the exploding gradient problem, which presents a similar-yet-distinct issue. Where the vanishing gradient problem involves a model becoming unable to converge to a local optimum due to decreasingly smaller gradients, the exploding gradient problem describes the situation where a model becomes unstable, in which parameters continually increase in magnitude, diverging to the point where each training step produces a model with vastly different results to the one before it. Several of the hyperparameter and optimisation tools previously described aid in preventing exploding gradients - as well as vanishing gradients - from forming: the appropriate initialisation of model weights and biases [151, 152, 176]; non-saturating activation functions such as ReLU [160] and Leaky ReLU [161, 162]; and normalisation techniques such as batch normalisation [163, 165, 166].

Another technique used to mitigate exploding gradients is gradient clipping, in which the gradients calculated during the backwards pass are prevented from exceeding some threshold by clipping outlier gradient vectors that lie beyond some threshold, before being passed to the optimiser algorithm. The basic form of gradient clipping involves setting two hyperparameters as the minimum and maximum allowed values that a gradient can take - this method is commonly called 'clipping by value' or 'clipval'. However, this method can cause the direction of a clipped gradient to change. For example, if a two-dimensional parameter space produces gradients [0.5, 10.0] with direction 87°, and the threshold values are set as [-1.0, 1.0], then the clipped gradients have the values [0.5, 1.0] with direction 63°. The alternative, used to preserve the gradient direction, is clipping the gradient by the maximum L2-norm value of the gradient vector, scaled by some hyperparameter coefficient - this technique is given the name 'clipnorm' [177]. Using the previous example, if the clipnorm value is set to 1.0, then the returned gradients would be [0.0499, 0.9988] with direction 87°, hence the gradient direction is preserved.

Chapter 3:

Analysing Metal-Molecule Interactions on the Atomic-Scale

Author's note: This chapter encapsulates and expands upon previously published research [1], which covers the design and evaluation of a novel machine learning and image processing pipeline for the chemometric analysis of SERS data in collaboration with members of the Baumberg research group.

Contents

The profuseness of catalytic processes, a drive for efficiency, and the minimisation of precious resource utilisation has developed a need for resolving molecular interactions at heterogeneous interfaces at an atomic level [92, 178]. However, few methods offer this level of resolution, and none currently allow for operando studies. Promising techniques are arising with sub-molecular sensitivity but heavily rely on indirect interpretation of spectroscopic data, making such processes prohibitively

time consuming [73, 179, 180]. To this end, bespoke and robust analysis methods are required that can digest large datasets to build up a comprehensive understanding of the atomic-scale processes involved.

Fundamentally in conjugated molecular systems, which are systems of connected p-orbitals with delocalised electrons, resonance structures exist. These are sets of Lewis structures - diagrams of a molecule showing the bonds between constituent atoms, alongside any valence electrons - that are incomplete representations of a molecule that contribute to describing how the molecule is bonded. The complete representation of a molecule is the average of all resonance structures [181]. One example of a resonance structure is Benzene, which has two Lewis structures that represent both forms of the molecule (see Figure 3.1). These forms differ in the arrangement of three sets of double bonds. The energy between the double bonds for each resonance structure is nearly identical, so it resides in both states with equal probability.

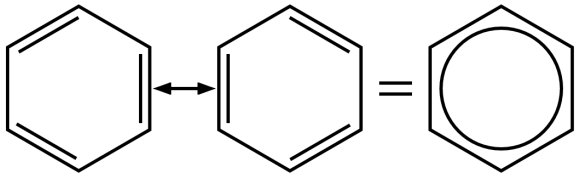


Figure 3.1: *Left and middle*, resonance structures of Benzene; *right*, the averaged structure.

If an electron donating or withdrawing group is added to the system, the charge density across the molecule will change, as will the possible resonance structures. If one of these resonance structures has a lower energy than the other - assuming two for simplicity - then the molecule will primarily exist in the lower energy state, switching to the other with some smaller probability. A time-averaged view of this resonance molecule causes some bonds to become stronger than others. With regards to SERS, the same situation can arise when resonance molecules form molecular bonds with atomic-scale features such as under-coordinated adatoms. These bonds alter the electron density of the system, forming new possible resonance structures, which cause the same asymmetry in bond strength.

For the SERS experiment carried out by the Baumberg research group at the University of Cambridge, a NPoM geometry was chosen for high optical field enhancements and high reproducibility, whilst allowing for a large number of structures to be probed [145]. A SAM of an analyte spacer molecule, biphenyl-4-thiol (BPT), is formed on a gold surface film [145]. Gold nanoparticles are then deposited onto the SAM surface, which are the regions where both nanocavity and picocavity sites can form (see Figure 3.2). Due to the asymmetry in bond strengths caused by bonds formed between BPT and gold, a different set of vibrational modes can be excited from those in the standard resonance structures of BPT under a homogeneous field. Where these modes are normally either strictly Raman-active or IR-active [36, 182], the existence of an inhomogeneous field caused by a SERS nanocavity can excite a different set of modes, including IR modes [72, 183, 184].

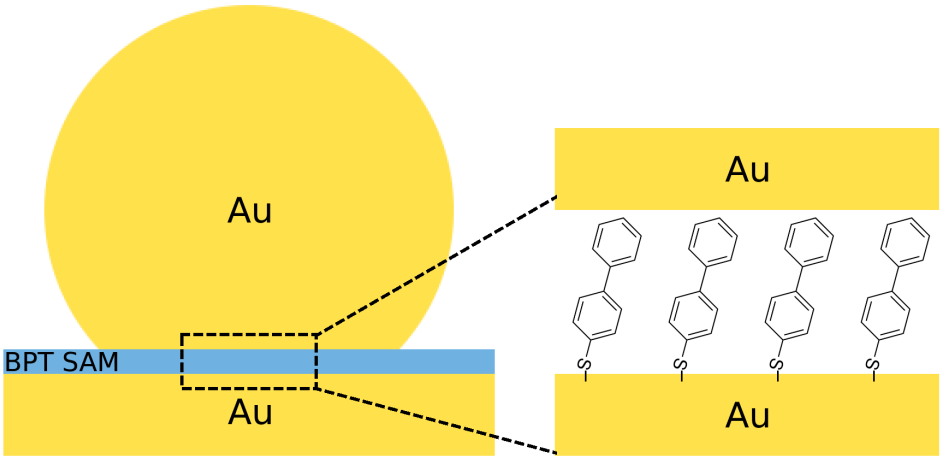


Figure 3.2: NPoM geometry of a single nanoparticle site. From bottom-to-top: the gold film, SAM of BPT, and gold nanoparticle form a device from which nanocavities and picocavities can occur. The inset shows the formation of the SAM, in which the thiol bonds at an angle to the gold film.

A transient picocavity event, which forms during the irradiation of the NPoM geometry and can last up to several seconds, corresponds to a gold adatom appearing from a facet on either the nanoparticle or substrate surface - an illustration of which is shown in Figure 3.3. Due to these fluctuations occurring on the atomic-scale, new sets of Raman lines tend to be observed, as the strong field gradients allow for IR modes to be excited [75, 148, 185]. The specific modes are based on a combination of parameters including the orientation of the molecule within the NPoM geometry, and the strength of the local electric field. Picocavities produce peaks with a higher general intensity than that of nanocavity spectra [186], but can also alter the relative intensities of those peaks, due to atoms on the molecule that are in close proximity to the inhomogeneous field being affected more intensely than those that are further away [187]. Interactions between the molecular pi-system and the low-coordinated gold adatom, which typically drifts around the nanoparticle or substrate surface, cause changes to the molecular structure of the BPT analyte. This has the effect of continually varying the positions of Raman lines from their otherwise stable frequencies [73, 188].

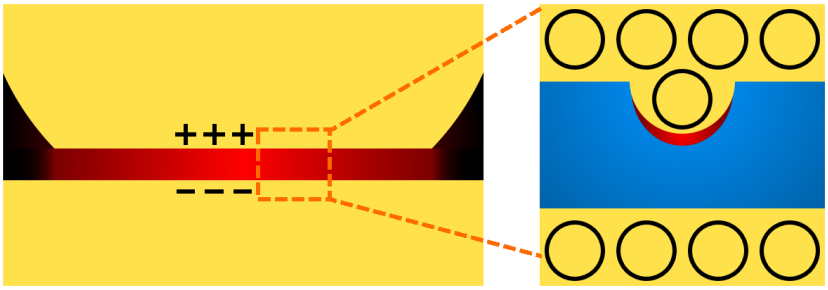


Figure 3.3: *Left*, illustration of the constant field in a nanocavity; *right*, illustration of the strong local field gradient in a picocavity due to the existence of an adatom on either the nanoparticle or substrate surface. Different vibrational modes of the spacer molecule (not shown) can be excited based on a complex of possible geometric arrangements of the gradient field in relation to the molecule.

Knowledge of which vibrational modes belong to a particular picocavity event, as evidenced by the transient SERS spectra that are produced from it, can be used to determine how that picocavity interacts with the electron density of BPT, which of its bonds are affected the most by it, and the geometric position of the picocavity field [187]. The magnitude and polarity of how transient peaks are correlated, in terms of the degrees of Raman shift with respect to one another due to adatom drifting, could also be used to extract the strength of the picocavity interactions between a gold adatom and BPT (this aspect of picocavity analysis is covered in the following chapter). Obtaining and subsequently characterising these features of the picocavity interactions is beneficial, as low-coordinated gold has applications in heterogeneous catalysis, which can be used to experimentally demonstrate how such picocavity interactions promote specific chemical reactions [92, 93, 94, 178, 189, 190, 191]. Other applications that are facilitated by this furthered understanding include molecular electronics [87], memristive switching [88], and ultra-sensitive sensing [73, 89].

In collaboration with members of the Baumberg research group, based on their knowledge of the chemical systems within this chapter, density functional theory (DFT) is utilised, which is a commonplace quantum mechanical modelling method used for studying the electronic structure of molecular systems. Using a DFT model, the vibrational modes of BPT can be analysed by simulating how an arbitrary homogeneous electric field interacts with the molecule. As the picocavity field is produced by a single gold adatom it will have a gradient effect on the BPT molecule, therefore DFT alone is an unsuitable method for studying the interactions of this inhomogeneous field. Thus the vibrational modes calculated by the DFT are used with a dipole approximation to model the strongly inhomogeneous field produced by the picocavity - based on a specified orientation of the molecule and amplitude of the field - to calculate the Raman scattering cross-section of the molecule, and retrieve the predicted Raman lines at the corresponding amplitudes and positions [187]. A shortcoming of the method is that, in order to assemble a full picture of SERS interactions with a sample molecule, it requires the individual simulation and analysis of each possible orientation of the molecule and

the varied local electric field strengths. This becomes an impractical method when considering the wide range of possible interactions between the metal protrusions and the analyte molecule, making repetitions rare and signals difficult to interpret without large amounts of data to verify predictions made by the DFT.

Through pattern recognition, machine learning is used as a part of a data analysis pipeline to extract salient features from unlabelled SERS data - thus inherently containing inhomogeneous picocavity fields. These features may arise from chemical changes in the analyte molecule, or from morphological changes in the metal surface [70, 75, 180, 192, 193, 194]. Such features are particularly prevalent in the SERS spectra of few or single molecules. These are traditionally difficult to study due to their transient nature, but offer in return a unique opportunity to elucidate the behaviour of molecules on an atomic length-scale [73]. Common picocavity events were extracted, hereby referred to as *Configurations*. This information can be used to compare to known vibrational modes of BPT to identify those that are excited by these picocavities [73]. Additional insight can be gained through extracting information such as the geometric positions of picocavity fields, and which Configurations occur most often. This would suggest an energetically favourable arrangement for the molecule when subjected to interactions with a plasmonic field, which could aid in future catalytic design.

3.1 Data Processing and Machine Learning Model Design

The workflow model that is used to analyse the BPT SERS spectra involves multiple sequential steps that can be broken down into machine learning and image processing stages, and a subsequent comparison to DFT simulations. Figure 3.4 summarises the machine learning processing stages that are described within this section.

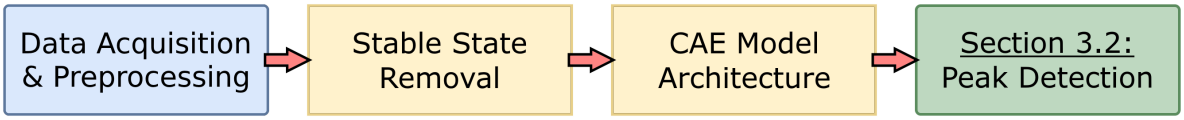


Figure 3.4: Flowchart showcasing part of the data analysis pipeline focused on the machine learning processing of the SERS data. The explanations of the input (blue) and the data processing steps (yellow), which each form a different stage of the methodology, will be expanded upon in the respective subsections. The output (green) will be expanded upon in Section 3.2, describing the remaining stages pertaining to image processing, clustering, and comparisons to DFT simulations.

Chapters 3 and 4 utilise a complex data analysis pipeline. During the research and development process, several terms were defined to succinctly define and classify various concepts and data types produced at different processing stages. These terms are summarised in table 3.1.

Table 3.1: Descriptions of terminology used throughout Chapters 3 and 4.

Term	Description
Spectrum	A SERS Raman spectrum. These constitute the main input data for the machine learning analysis pipeline.
Scan	A set (or subset) of time-series SERS spectra produced from a single NPoM site, irrespective of the presence of picocavities.
Track	A group of wavenumber positions marking a single time-series picocavity peak, which has been 'tracked' over its lifetime.
Event	A collection of Tracks that belong to the same instance of a physical picocavity event. An 'Event' could more rigorously be described as an 'Algorithmic Event' to better distinguish from the physical counterpart it is trying to encapsulate, but the former is used for the sake of brevity.
Configuration	A clustered set of Events that aims to solely contain picocavity spectra that represent the same physical picocavity event.

3.1.1 Data Acquisition and Preprocessing

To keep the number of confounding factors to a minimum, BPT was selected as the SAM analyte for the NPoM geometry. The molecule has a strong Raman cross-section as a result of a vertical dipole that can be induced along the two aromatic phenyl rings [195] (see Figure 3.5). It is also inert, with few conformational isomers, meaning that BPT is a molecule with a rigid structure that can provide a stable SERS signal over time on the order of minutes, with occasional transient events which form on the order of seconds. These advantageous qualities allow the behaviour of light-induced interactions with gold and BPT to be studied in detail, due to the limited number of configurations that BPT can take compared to other, larger tracer molecules. Once characterised, the analysis of BPT could then be applied to more complex structures, such as catalysts [196].

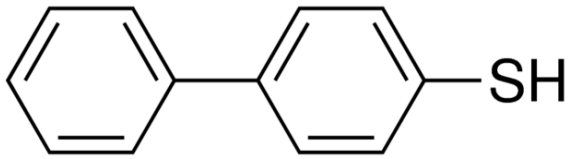


Figure 3.5: Structural formula of BPT, featuring two aromatic phenyl rings, one of which is bonded to the sulfur H-bond that forms the thiol group. It is the sulfur that forms bonds with gold adatoms to produce the observed SERS signals.

The experiment used a custom-built Raman microscope to capture the BPT SERS spectra [144], featuring an Olympus BX51 microscope, which was coupled to an Andor Shamrock 303i dispersive spectrometer, with a 600 lines/mm grating and an Andor Newton 970 BVF electron-multiplying CCD.

A schematic is provided in Figure 3.6. An in-house particle finding algorithm was used to locate each picocavity device in the NPoM geometry, which utilised threshold detections on dark-field scattering images to maintain alignment with each device [188]. A 632.8 nm HeNe laser was used at three different laser powers: 447 µW, 564 µW, and 709 µW. The spectra were collected using an Olympus NA (0.8) 100x darkfield objective. The integration time for each SERS measurement was approximately 35 ms; the duration of each measured SERS spectrum is here referred to as one time step.

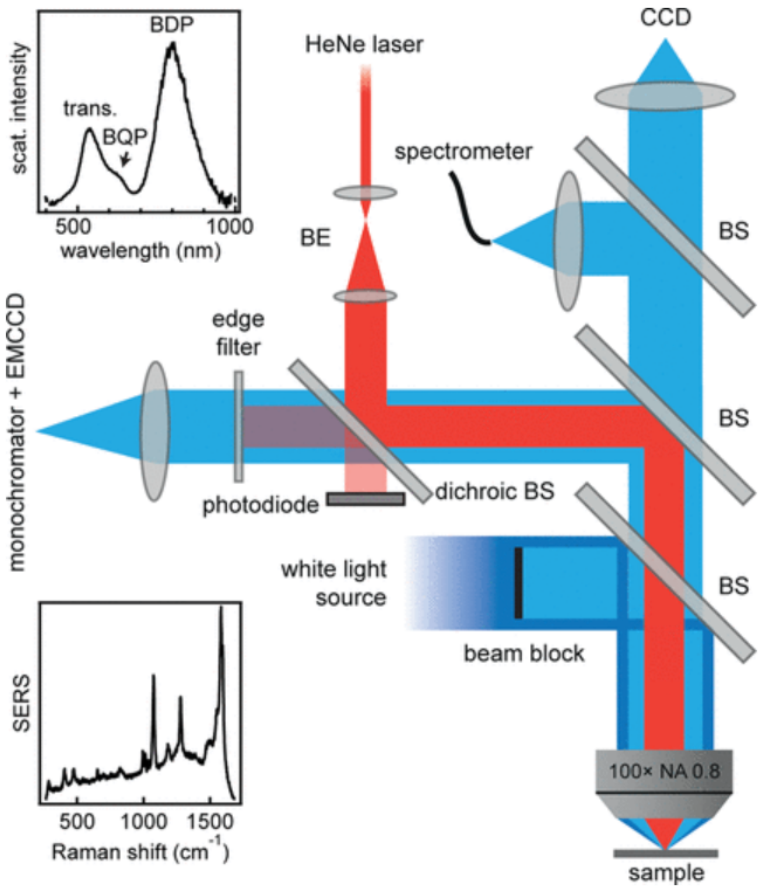


Figure 3.6: Schematic for the SERS setup, courtesy of Felix Benz, *et al.* [144]. The inset figures show typical scattering (top) and SERS spectra (bottom) produced by a single picocavity event.

Scans were produced for the BPT dataset, which are sequential time-series spectra each consisting of 1000 time steps. The signal is captured with a wavenumber range of -1606 to 1611 cm^{-1}, with a wavenumber resolution linearly increasing from 2.5 cm^{-1} in the anti-Stokes region, to 1.6 cm^{-1} in the Stokes region, as a result of the dispersive detector producing a non-uniform resolution. There were 416

scans measured at a laser power of 447 µW, 500 scans at 564 µW, and 499 scans at 709 µW - totaling 1415 scans measured, or 1415000 spectra. Scans that were found to contain any number of picocavity events were termed *True* scans, whereas scans that were not found to contain any picocavities were termed *False* scans. The inspection and labelling of each scan was carried out by a collaborator in the Baumberg research group. In total, there were 416 scans found to contain picocavities, and 999 scans were found to only contain stable nanocavity signals.

The dataset was preprocessed before use by a neural network. Firstly, all spectra had an average background of 300 counts (CCD dark current) that was subtracted. They were then interpolated to a wavenumber range of 268 cm^{-1} to 1611 cm^{-1}, at a constant wavenumber resolution of 2.625 cm^{-1}, using a cubic spline interpolation. This interpolation produced spectra with 512 bins, which excluded the wavenumber range that was filtered out using a notch filter, as well as the anti-Stokes region from the dataset. The anti-Stokes region constituted redundant information, as the spatial positions of the Stokes and anti-Stokes peaks are mirrored, and the temperature information which could be derived from the ratio of the intensities of the mirrored peaks [197] is not under scrutiny. This interpolation aligns the wavenumbers of all spectra within the dataset, regardless of laser power, and alleviates a bias incurred by having a non-uniform resolution wavenumber-space [198]. The resulting interpolated spectra were then linearly normalised between the range [0, 1]. The purpose of normalising a dataset is to equalise the contribution of each sample in the training of a neural network, as a sample with larger values would intrinsically influence how a neural network would train to a greater degree than one with smaller values - which is the case for spectra taken with a higher laser power, as shown in Table 3.2.

Table 3.2: Change in the mean and standard deviation of values (counts) between spectra as a result of preprocessing for each laser power. The values stated both before ('Raw' column) and after ('Normalised' column) normalisation were exclusively taken from the interpolated range.

Power (µW)	Raw		Normalised	
	Mean	Std	Mean	Std
447	6.35	2.96	0.26	0.07
564	7.94	3.51	0.27	0.08
709	9.88	4.28	0.26	0.08

The training dataset had random uniform noise added independently at each epoch to increase the variance of the samples, which has the effect of producing a more generalised network with smaller weights, due to it being more challenging to memorise the training samples [199]. As the spectrometer used a dispersive detector, the noise was added to each bin in a spectrum proportional to 5% of the magnitude at each wavenumber. This is because the noise at each data point in a dispersive system scales proportionally to the square root of the respective signal and is independent of other wavenumbers.

3.1.2 Stable State Removal

The BPT SERS spectra contain a noise background, alongside sets of peaks produced independently by the nanocavity and picocavity fields, which are respectively termed stable and stochastic transient SERS events. As noted in Subsection 3.1.1, BPT produces stable SERS signals, which result in nanocavity peaks appearing at consistent wavenumbers with pseudo-stable peak ratios and background intensities. These combined aspects are termed the *stable state* of the BPT SERS spectra. Leveraging this consistency, by removing the stable state from each spectrum, what remains are the features of interest: the transient peaks, produced by picocavity devices, at the isolated intensities. Isolating transient peaks is an important step for building a representative dataset used in subsequent analysis stages, as without doing so brings a risk of including peaks associated with nanocavity devices, or potentially missing several transient peaks that share wavenumbers with other peaks belonging to the stable state.

To remove the stable state from each spectrum, a neural network was trained to readily adapt to any variances in the stable state to achieve robust isolation of the transient peaks and any corresponding event characteristics for further analysis. A one-dimensional CAE was trained for this purpose, which was tasked with reconstructing the stable state of each input spectrum. This was achieved by creating training and validation datasets from scans that only contained stable states, and reserving all scans with at least one picocavity event for model inference in the testing dataset.

The strength of an autoencoder is in the ability of the architecture to learn salient features of an unlabelled dataset, which also have the property of shift-equivariance [153] due to the addition of convolutional layers. Variations exist such as a denoising autoencoder, which learn useful properties for reconstructing data with noise removed. However, in this scenario the entirety of the training data may be considered as noise, and so a standard autoencoder architecture is trained to reconstruct the input. Designing an autoencoder, and partitioning each dataset in this way, allows the stable state of True scans to be subtracted leaving behind residual intensities in the form of transient peaks. With this routine, the subtraction of a stable state from a False scan would be empty - provided the stable state was perfectly reconstructed. These subtracted spectra are termed *picocavity* spectra, although subsequent work on peak detections utilises picocavity scans due to the nature of the data, which partitions samples into batches exclusively containing each measured SERS site in a time-series.

3.1.3 CAE Model Architecture

The CAE used in this work contains 11 layers, including the input and output layers. There are four convolutional blocks in the encoder, followed by a 32-unit FC embedding layer, which was used as an input for the decoder. The model depth and size for each layer was determined through a grid search optimisation, minimising the MSE loss. The decoder mirrors the architecture of the encoder, with one FC layer whose output is reshaped to fit the next convolutional layer, and four convolutional

blocks that upscale the data to reconstruct the input spectra. The output of each layer was normalised using group normalisation [168]. This was followed by a Leaky ReLU activation function with slope coefficient, α, of 0.3. A maxpooling layer with a stride and kernel size of two was used as the final layer in each convolutional block in the encoder, and an upscaling layer with a factor of two was used as the first layer in each convolutional block in the decoder. Figure 3.7 shows a block diagram of the CAE.

The model was trained for 2500 epochs, with a static learning rate of 0.001, and a batch size of 500 spectra. This splitting of each scan into equal halves was necessitated by memory limitations in the hardware used to train the model. The training dataset consisted of 749 False scans, the validation dataset therefore contained the remaining 250 False scans, and the testing dataset contained all 416 True scans. Note that each scan contains 1000 spectra, which are the individual samples used to train the CAE, resulting in a total of 749000 training samples. This distribution of scans produced an approximate 5:2:3 dataset split. The MSE loss function was used to evaluate the difference between the input and reconstructed spectra, and the Adam optimisation algorithm (using parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-7}$) was implemented to adjust the model parameters during training. Each layer was regularised using L2 weight decay with a regularisation factor, γ, of 0.1. Clipnorm [177] was used to clip the calculated gradients to the maximum L2-norm value for each update step, to avoid the problem of exploding gradients.

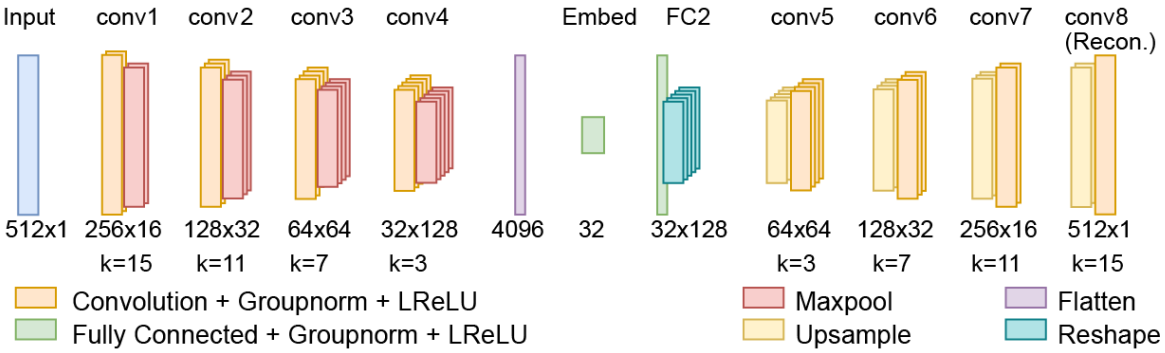


Figure 3.7: Block diagram of the CAE. The dimension labels on each convolutional layer are in the format (#Features, #Filters), which represent the output of each block. Note, the batch size dimension is equal on all layers and is thus omitted. The k-values represent the size of each convolutional kernel, which use a unitary stride. The convolutional and maxpooling layers both use zero padding to capture the entire receptive field.

3.2 Peak Detection, Track Isolation, and Picocavity Analysis

Following the use of the CAE to reconstruct the BPT stable states of picocavity spectra in Section 3.1, subsequent image processing and clustering steps are required to isolate time-series picocavity peaks,

and compare the results to data simulated by the DFT. These remaining processing and analysis steps are covered within this section, as outlined by Figure 3.8.

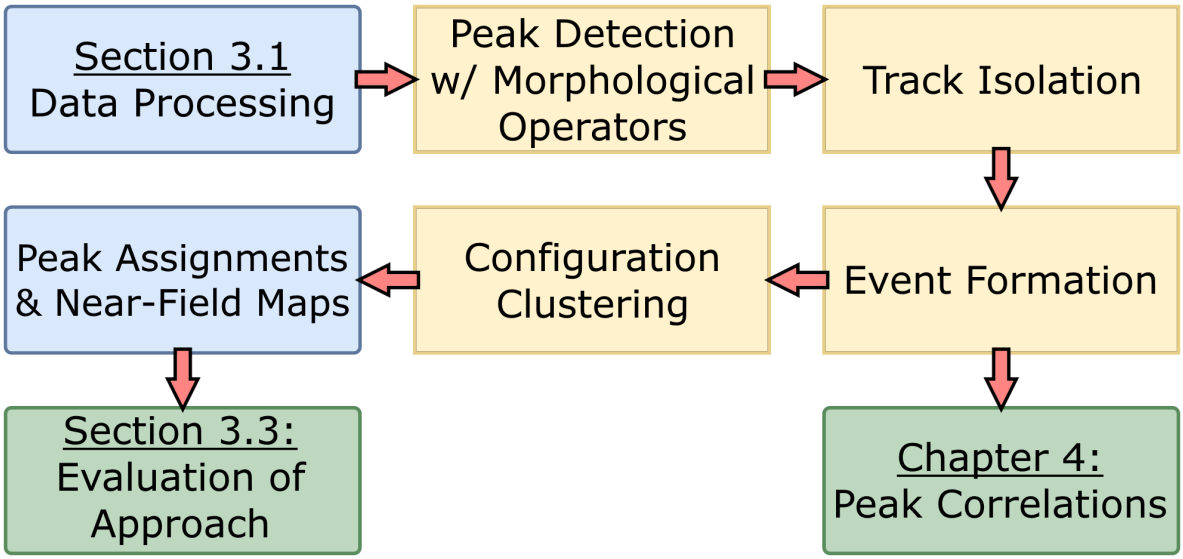


Figure 3.8: Continuation of the data analysis pipeline seen in Figure 3.4. An explanation of the data processing steps (yellow) and the resulting output (blue) will be expanded upon in the respective subsections. An evaluation of the pipeline is provided in the following section, but also a breakaway at the 'Event Formation' stage expands into the following chapter, which utilises temporal SERS information to analyse correlated peaks to extend the functionality of the data analysis pipeline.

3.2.1 Peak Detection

The picocavity scans, produced by subtracting the reconstructed stable states from the input scans, contained noise that was not captured by the model, which was disruptive to isolating each transient peak. To remove this, a series of signal processing steps were applied. Firstly, both the input and reconstructed scans were smoothed using an SG filter along the wavenumber axis, with a filter window length of 7 px and a second order polynomial. For the input scan, the filter acts as a high frequency noise filter, removing the electrical read noise from the CCD used to measure the SERS signal. The picocavity scan is then calculated using the following equation:

$$P_{\lambda,t} = \max(I_{\lambda,t} - R_{\lambda,t}, \phi) - \phi, \tag{3.1}$$

where $P_{\lambda,t}$, $I_{\lambda,t}$ and $R_{\lambda,t}$ are the picocavity, input and reconstructed scans, respectively, and ϕ is a global offset parameter, which is equal to 5% of the standard deviation of the input scan. The purpose of this parameter is to act as a threshold for peak detection, only allowing signals associated with transient peaks to remain as the residual intensities after the subtraction of the input spectra, whilst

minimising the number of false detections due to noise. For example, scans with a large SNR would have a small offset, which lowers the intensity threshold for a picocavity peak to be detected, thus allowing for weaker picocavity signals to be analysed. The values of the SG filter and offset parameters are selected through an empirical test on a representative subset of BPT data. Figure 3.9 shows a scan segment before and after stable state removal.

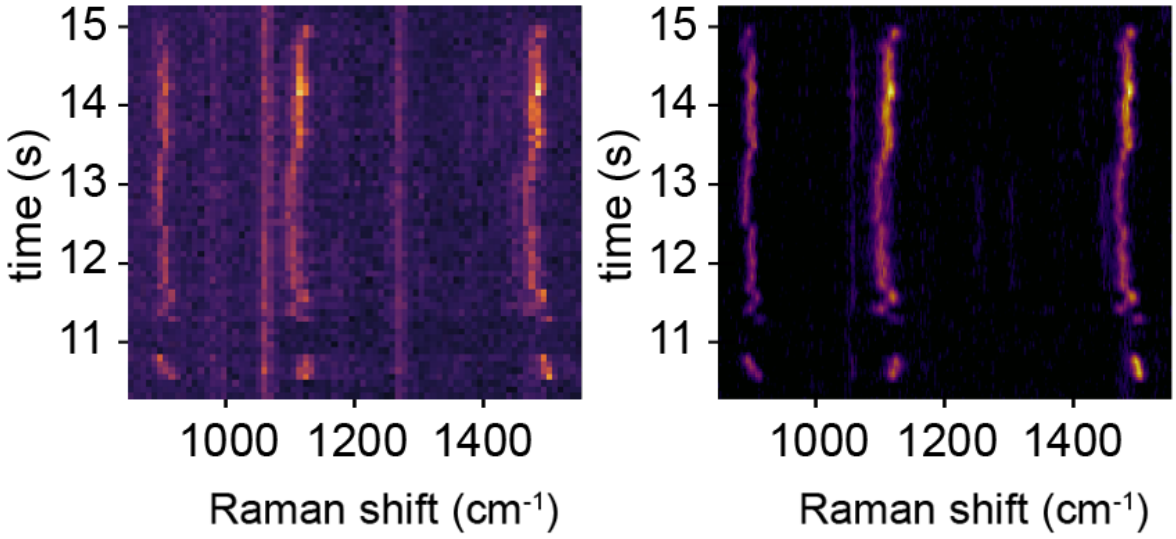


Figure 3.9: *Left*, example segment of a scan containing picocavity events; *Right*, the picocavity scan features stable nanocavity peaks that have been either suppressed or entirely removed (two of which notably occur around wavenumbers $1100\,\mathrm{cm}^{-1}$ and $1300\,\mathrm{cm}^{-1}$). The remaining residual information contains the transient peaks produced by picocavity events.

As a first step in peak detection, the most intense transient peaks are isolated by selecting the 98^{th} percentile of pixel intensities in the picocavity scans. Secondly, lower intensity transient peaks are identified in a similar fashion using the 96^{th} percentile of pixel intensities, but with the addition of a Boolean mask applied to only allow for pixels occupying the same rows or columns as previous detections to be accepted as valid pixels. A basic empirical study was performed on a representative subset of the complete BPT testing dataset, showing that the percentile values used provide the best results.

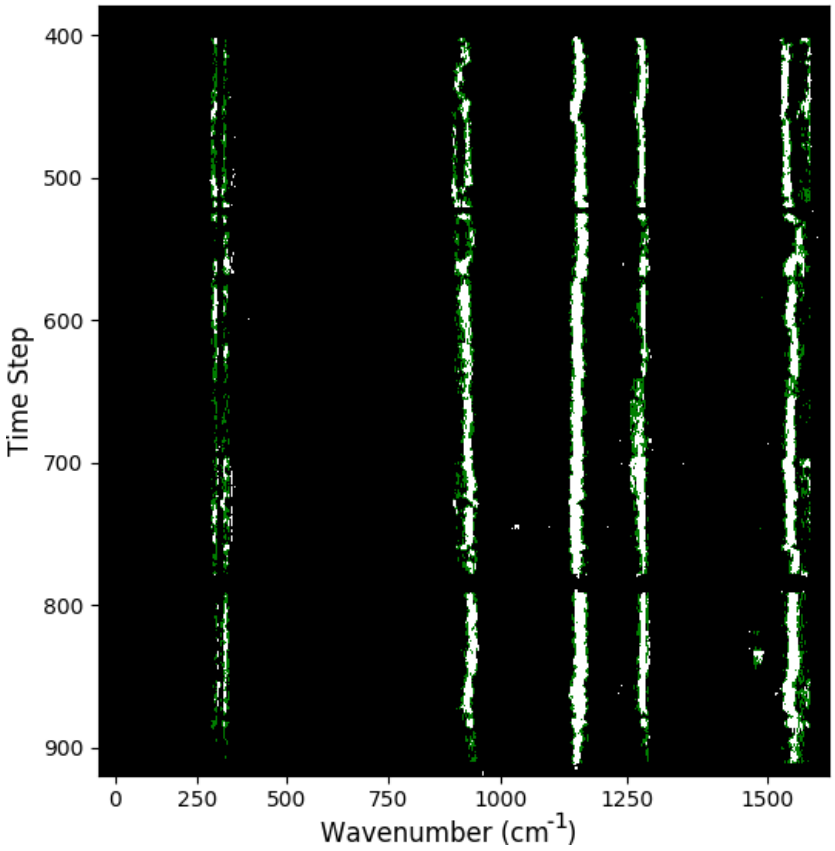
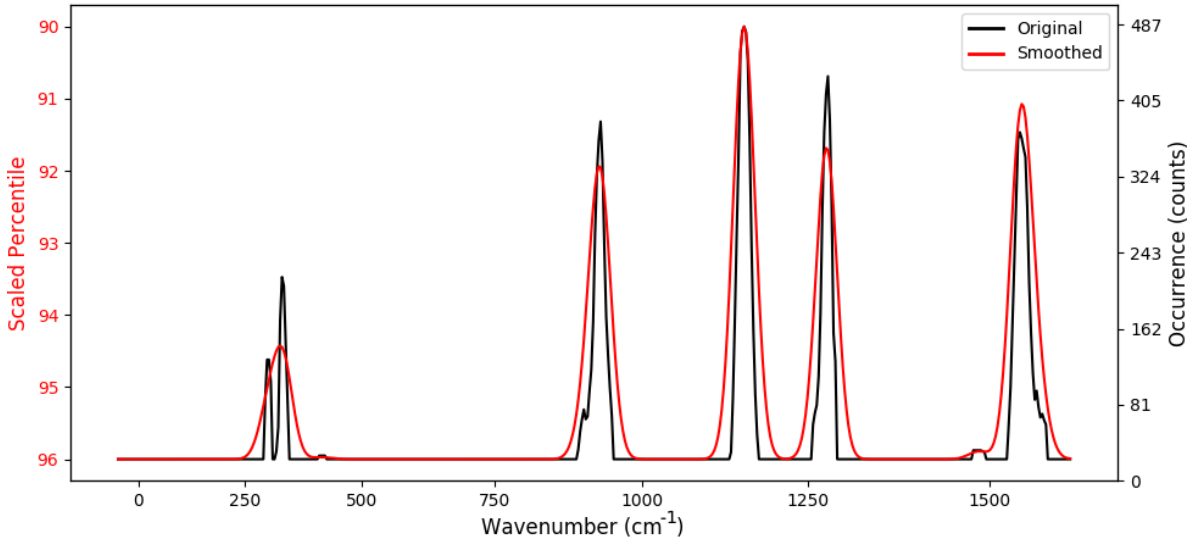


Figure 3.10: Pixels representing parts of transient peaks found by the first detection stage (98^{th} percentile) are in white, whilst additional pixels from the second detection stage (96^{th} percentile with Boolean masking) are shown in green.

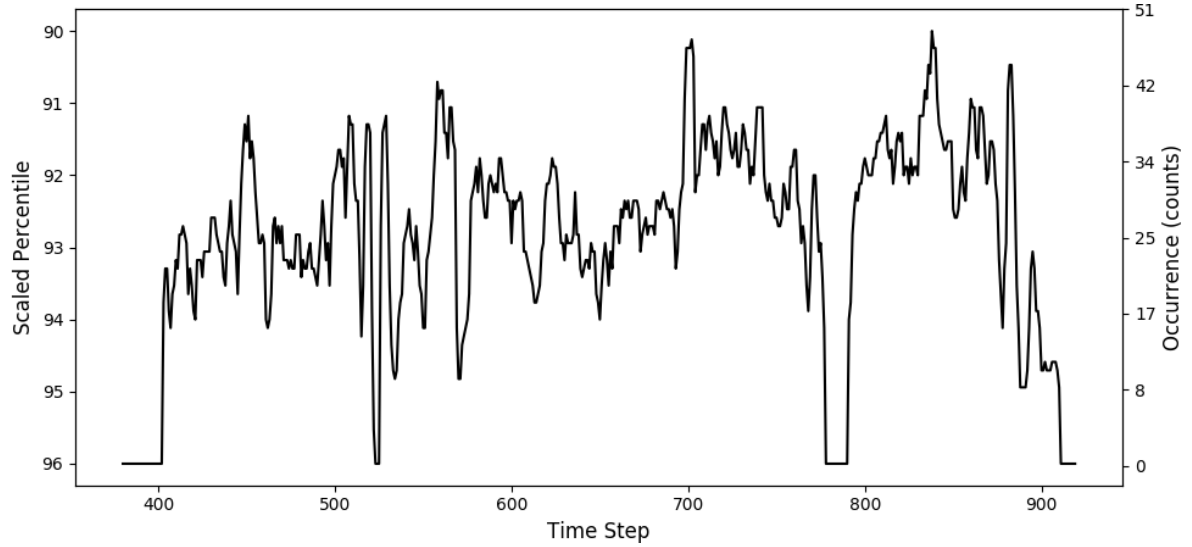
The two detection stages described capture the majority of pixels containing transient peaks in most scans. However, scans with a lower SNR were found to contain small gaps in otherwise complete sequential transient peaks in a time-series - an example of this can be seen on the transient peaks occurring around $250\,\text{cm}^{-1}$ in Figure 3.10. To counteract this, two probability density functions were estimated by counting the number of pixels detected along each axis from the combined previous stages. The probability density function projected along the wavenumber axis is smoothed, using a one-dimensional Gaussian filter at five standard deviations, to approximate the centroids of each transient peak. The time-axis probability density function is not smoothed to reduce the likelihood that noise, which could fill a gap between two distinct transient peaks separated in time, would be detected as a peak.

Two percentile ranges are then fit to the respective probability density functions, which scale between the 90^{th} and 96^{th} percentile of pixel intensities, where the lower percentile bound is applied

to the highest count, and vice versa. Note that detections made along rows or columns that had previously produced zero detections were discarded, which effectively sets a 0^{th} percentile value for those regions. Using these percentile ranges, additional pixels contributing to transient peaks are detectable along each axis on picocavity scans, and an intersection of the two sets of detected pixels forms the final coordinate set of transient peaks in pixel space, combined with the previously detected positions. Figure 3.11 shows the two probability density functions resulting from the scan segment shown in Figure 3.10, with the resulting detections seen in Figure 3.13 in the following subsection.



(a) Estimated probability density function along the wavenumber axis.



(b) Estimated probability density function along the time step axis.

Figure 3.11: Probability density functions estimated from data collected during previous detection stages. There are two y-axis scales on both plots that show the conversion from the number of pixels identified as a transient peak (right), and the percentile assigned to those values (left). The smoothed probability density function is used to detect transient peaks along the wavenumber axis to account for any misalignment to the peak centroid due to an incomplete set of detections.

3.2.2 Morphological Operators

Mathematical morphology is an analysis technique commonly used to probe and transform the structure of a digital image. Two key concepts are employed to achieve this: morphological operators and structuring elements. The morphological operator used in this work is morphological opening, which involves the dilation (expansion) and erosion (reduction) of an image. This has the effect of removing small artefacts from an image, specifically noise in this case. An illustration of this process is shown in Figure 3.12. The effectiveness of the noise removal process is determined by the chosen structuring element, which has two properties: shape and size, with the latter determining the resolution of probed features. A 3×3 rectangular structuring element was chosen for this work.

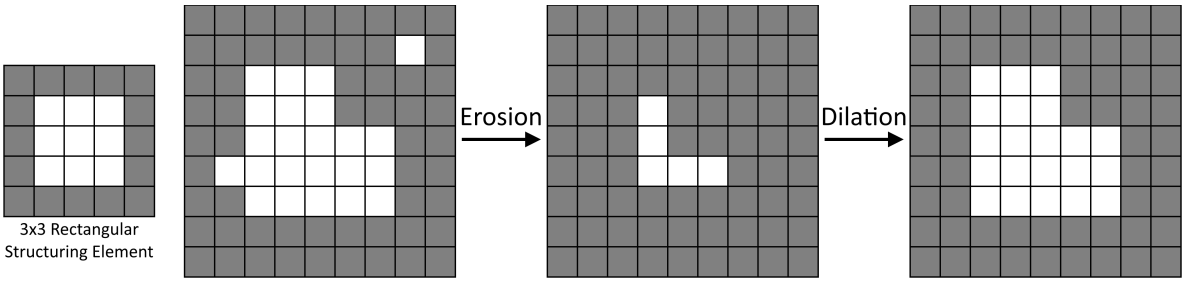


Figure 3.12: Illustration of morphological opening. Erosion is performed on the input image using the 3×3 rectangular structuring element. This shrinks the size of the main feature, and removes the 'noisy' single pixel in the upper-right. Dilation is then performed on the eroded image with the same structuring element to produce the output image. This restores most of the input image, whilst simultaneously removing noise and smoothing any small protrusions.

The three stages in the peak detection process may capture intense noise alongside the transient peaks in some scans - particularly those with a low SNR - as seen in Figure 3.10 at $1000\,\mathrm{cm}^{-1}$ and above, which may exist due to algorithmic error, or from physical aberrations such as cosmic rays. To prevent the accumulation of noisy detections potentially bridging small (1 to 3 px) gaps between otherwise distinct transient peaks, morphological opening was used after each detection stage, with a 3×3 rectangular structuring element. The result of which, applied after the final peak detection stage described in the previous subsection, is shown in Figure 3.13.

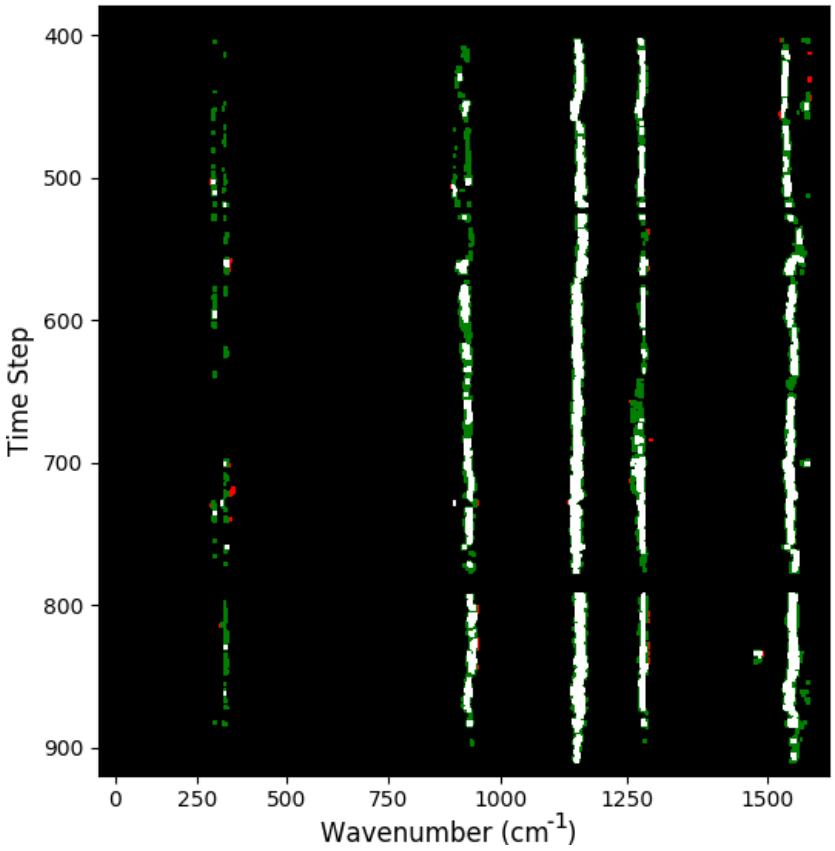


Figure 3.13: Resulting pixels detected as transient peaks after morphological opening from each stage - the first stage (98^{th} percentile) is shown in white, the second stage (96^{th} percentile with Boolean masking) is shown in green, and final stage (estimated probability density functions with scaling percentiles) is shown in red. The (row, column) coordinate pairs of each detected pixel form the dataset of available transient peaks, produced by picocavities, used in further analysis.

The choice of the specific structuring element (both shape and size) were determined based on a heuristic analysis, and from advice given by the Baumberg research group. The rectangular shape of the structuring element was chosen based on the discontinuous nature of Raman peaks in a time-series - i.e. the immediate, simultaneous appearance and disappearance. The height of the structuring element was set to three time steps, which is defined as the minimum number of time steps that two contiguous peaks must be separated by in order for them to be considered as distinct. The width of the structuring element was also set to three (around $8\,\text{cm}^{-1}$), which was found to reduce the largest portion of noise whilst still retaining the transient peaks. However, it is noted that there exists extremely short-lived transient events (named here as *ephemeral events* to distinguish them from the rest) that may last for a single time step, which would therefore be filtered by this morphological processing stage. Hence,

though ephemeral events are rare, there is room for improvement at the detection stage to incorporate events of this nature.

3.2.3 Track Isolation

Once transient peaks have been detected on a picocavity scan, the next stage is to form sets of sequential transient peaks that form a contiguous, or almost-contiguous, line through time. These are termed *Tracks*. Tracks are initially demarcated by the final detected pixels that are neighbours in an 8-connected sense - meaning that one pixel has a neighbour if it is within two orthogonal hops of another pixel - within each scan (see Figure 3.14).

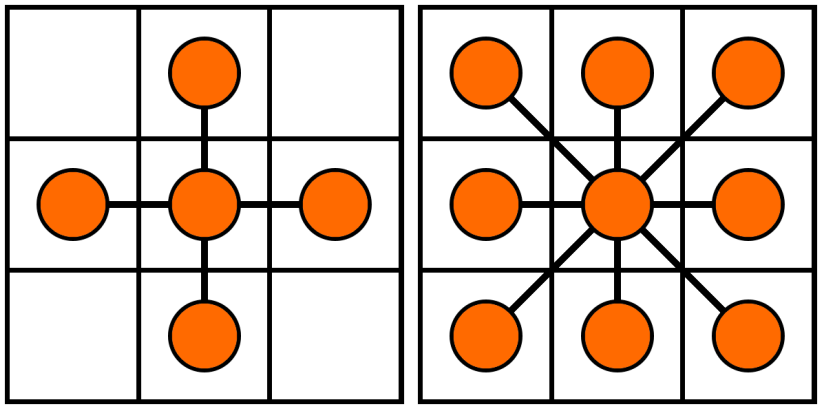


Figure 3.14: *Left*, visualisation of 4-connected pixels (edge-sharing); *right*, visualisation of 8-connected pixels (edge- and vertex-sharing). Both illustrations are based on the central pixel in each grid space.

Due to an incomplete retrieval of all transient pixels, gaps may appear along either axis, which would cause the initial 'connectedness check' to form two or more Tracks where there should otherwise only be one. Alongside this algorithmic error, cosmic rays and other noise can appear on a scan, which could bridge gaps between two Tracks that would be otherwise distinct if they were not successfully removed by the peak detection process. Additionally, gaps in detections may also be caused by physical effects such as: bistable picocavities, in which thermal fluctuations can temporarily breakdown a picocavity site during a measurement, as the adatom producing the enhanced field temporarily moves away from the BPT molecule before returning to reform the picocavity [74]; or 'flare events' [185], which appear as broad features with a dominant intensity that are picocavity-independent, and instead originate from Raman scattering within the metal. Flare events may cause low-intensity transient peaks to appear due to oversaturation, which are subsequently removed from the output picocavity spectra produced by the CAE reconstructions, thereby separating each Track that exists within those time steps occupied by the Flare event. Conversely to the aforementioned negative artefacts, a separation

of this nature may cause two Tracks to be considered genuinely, physically distinct from one another if the picocavity field was disrupted for a sufficient time. As mentioned in Subsection 3.2.2, a minimum of three time steps is required for Tracks to be considered distinct.

As the accurate identification of all Tracks within a single picocavity event is crucial, an iterative algorithm was created to merge constituent parts of a Track into one whole, whilst leaving Tracks unaltered that should remain so. This algorithm is hereby referred to as the *Zipper* - which was named as such due to its function being analogous to that of a zip on an item of clothing. The Zipper examines Tracks that are in close proximity to one another by utilising a 3×10 px sliding window. Tracks are only considered for merging if the closest vertical edges are within 10 px of one another, which translates to approximately $25\,\mathrm{cm}^{-1}$ - a value defined by the Baumberg research group as the maximum expected wavenumber shift of a transient peak - in addition, two Tracks must exist in time steps that lie within a tolerance defined by the Zipper window height (3 px). If both of these conditions are not met, then the Tracks are determined to be invariably distinct.

The top of the sliding window is placed at the earliest time step to contain both Tracks and is centred on the mean wavenumber position between both Tracks. Each sliding window calculates the mean separation, S, of the centroids between each Track, and stores this value in a running total as the sliding window moves along all valid positions with a stride of 1 px. Once all valid positions have been calculated, the 'global mean separation' is calculated as the mean of S, and the two Tracks are merged if this value either matches or falls below a tolerance value of 5 px. The Zipper as mentioned is iterative, so the described process repeats until a stop condition of zero Track updates is met for an iteration, at which point the outputs of the Zipper at that stage are the finalised forms of each Track, which are used in the analysis steps that follow. The result of the Zipper process is shown in Figure 3.15.

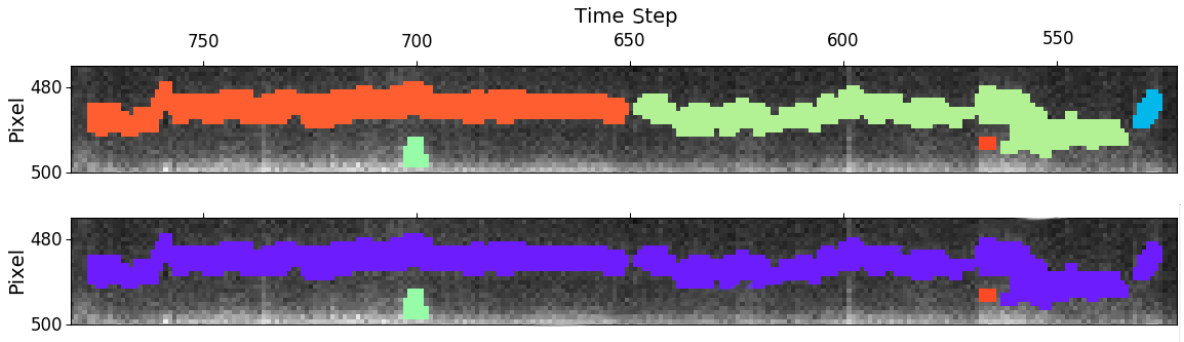


Figure 3.15: Example of the Zipper algorithm. *Top*, the initial Tracks formed through the 8-connected process; *Bottom*, the resulting merge of the three main Track sections. The Track sections in both images are arbitrarily colour-coded for clarity, showing the successful merge of the three main Track segments.

3.2.4 Event Formation

The vibrational modes associated with each Configuration typically produce multiple peaks at different wavenumbers. It is also possible for multiple Configurations to coexist [74, 75]. Because of this, the sets of peaks associated with each Configuration must be distinguished from one another. Transient peaks belonging to the same Configuration are termed *Events*. Physically, an Event should contain peaks that appear and disappear simultaneously, however, due to detection errors (explained in Subsection 3.2.3) there may be time steps shared between two Tracks that were not captured in respective formation processes. To create an Event, Tracks that share time steps are compared, and if the ratio of the number of shared time steps against the duration the longest Track exceeds a threshold, those Tracks are assigned to the same Event, as shown in the equation:

$$\frac{t_i \cap t_j}{\sum_{i>j} t_i} > T, \tag{3.2}$$

where t_i and t_j are the time steps that Tracks i and j occupy, and T is the Event threshold, given a value of 0.7, used to determine whether two Tracks should be assigned to the same Event. Two Tracks are compared this way each time, and if this equality is satisfied a new Event is formed, or an existing Event is appended to if one of the Tracks has already been assigned to an Event - see Figure 3.16.

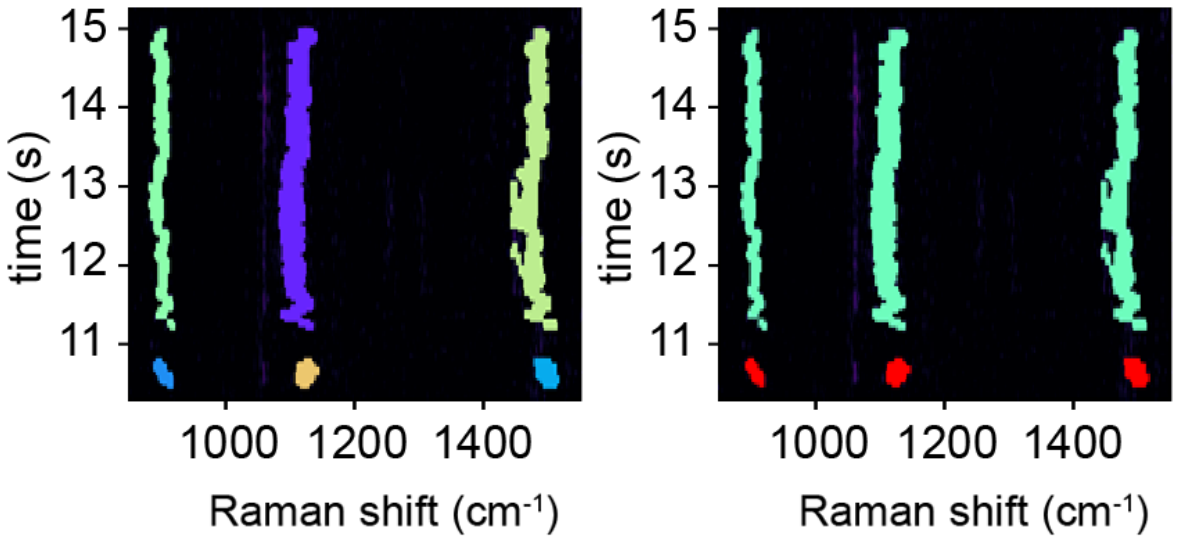


Figure 3.16: *Left*, six distinct Tracks produced by the Track isolation algorithm; *right*, two Events, consisting of three Tracks each, which have been created by the Event formation algorithm. The Tracks are arbitrarily colour-coded for clarity.

3.2.5 Configuration Clustering

As mentioned in Subsection 3.2.4, an Event represents one instance of a physical picocavity type. As a part of the process of analysing the BPT dataset, a method was used to cluster these Events into picocavity types, simply termed Configurations. One key point to note for a Configuration is that it can contain Events from different scans, as any arbitrary Configuration is scan-independent. To cluster Events into Configurations, the mean picocavity spectra of all Events are compared using the Wasserstein distance (also known as the earth mover's distance) to calculate the work required to transform between two spectra - these spectra were divided by the sum of their values to form normalised probabilities, as required by the Wasserstein metric. The equation for the Wasserstein distance is given:

$$\mathrm{D(u,v)} = \inf_{\pi \in \Gamma(\mathrm{u,v})} \int_{\mathbb{R}\times\mathbb{R}} |\mathrm{x-y}| \; \mathrm{d}\pi(\mathrm{x,y}), \tag{3.3}$$

where D(u, v) is the minimum amount of work required to transform the probability u into the probability v, |x - y| is the amount of mass needed to be moved between two points on each distribution, and dπ(x, y) is distance moved by each amount of mass. An (N × N) distance matrix is produced using this metric, where N is the number of Events. Spectral clustering is the clustering method used to form Configurations; as this technique requires a similarity matrix, the distance matrix was converted using the radial basis function,

$$\mathrm{A(u,v)} = \exp\left(\frac{-\mathrm{D(u,v)}^2}{2\delta^2}\right), \tag{3.4}$$

where A(u, v) is the similarity matrix and δ is a free parameter with a value of 0.275. The value of δ was selected to produce a well-separated distribution of the (converted) Wasserstein distances, with approximate values for the mean and standard deviation 0.5 and 0.35, respectively.

For spectral clustering the scikit-learn Python package was used [200], which requires a number of clusters to be prespecified. However, since the number of unique Configurations is unknown, the number of clusters with the highest mean silhouette coefficient [201] was selected from a range of 2 to 30. This score is a number ranging between between -1 to 1, which evaluates the classification success for all samples within a clustering operation in forming clusters that are well-separated (high inter-cluster distance), and dense (low intra-cluster distance), based off of a Euclidean sample separation. Silhouette coefficients of zero indicates that clusters are overlapped, whereas negative values indicate that misclassifications may have occurred. A subset of five picocavity scans were used as a test set for the continued development of the Configuration analysis, in order to reduce the volume of data that is required by further heuristic analysis. This subset contained scans with Configurations that were both unique, and shared between its members, through visual inspection. Figure 3.17 visualises the distribution of silhouette scores of the four largest clusters obtained through this method.

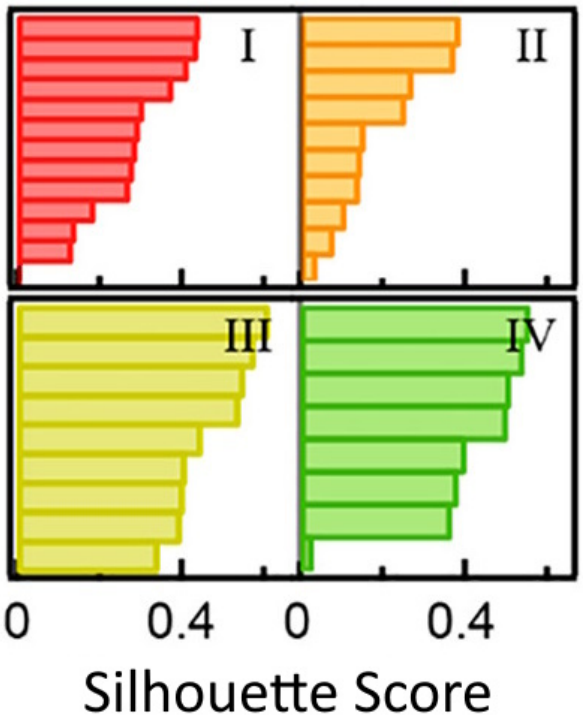


Figure 3.17: Visualisation of the silhouette scores for events in the four largest clusters (I-IV) where 1 means identical and 0 means no correlation. The representative spectra for all Configurations are shown in Figure 3.19 below.

Occasionally, rapid on-off switching of a picocavity is observed with near-identical spectra, which have minimal drift in the wavenumbers for each associated peak (see Figure 3.18). This physical effect causes multiple Events to be formed (as per the method described in Subsection 3.2.4) if there is a sufficient time separation between each occurrence, which results in a bias towards these 'flickering' (switching-type) Events at the clustering stage. This effect is more prevalent in BPT with other variations of NPoM geometries (discussed in Subsection 3.3.7).

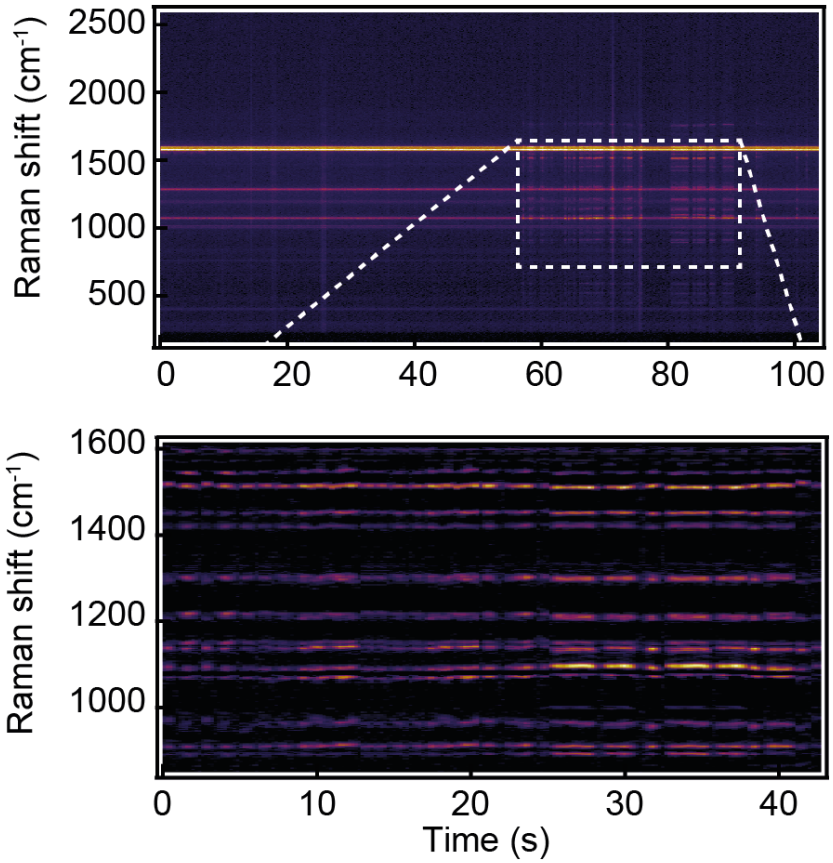


Figure 3.18: Example of on-off switching behaviour seen in some picocavities. The raw scan (top) was produced by a NPoM geometry with the addition of a monolayer of palladium functionalised onto the gold substrate, which tends to produce more 'flickering' events - Subsection 3.3.7 will expand on the analysis of this NPoM geometry. The isolated picocavity event is also shown (bottom).

To account for flickering Events, the initial Events clustered together and originate from the same scan are merged if they are within 100 time steps of each other. Then, the original clusters are dissolved, followed by a repeated spectral clustering procedure using the refined, bias-corrected Events. Lastly, because the nature of spectral clustering requires that all samples are assigned to a cluster, there are instances where exceedingly rare Events cannot form their own clusters - owing to the amount that are available, chosen based on the highest silhouette coefficient score. Due to this, these particular Events may possess negative scores within the assigned clusters. In order to improve the efficacy of each Configuration, these negative samples are discarded. From this evaluation 6 clusters (Configurations) achieved the best mean silhouette coefficient score. The representative spectra for each Configuration are shown in Figure 3.19. Table 3.3 shows Configuration statistics before and after flickering bias-correction, and the removal of Events with negative silhouette sample scores.

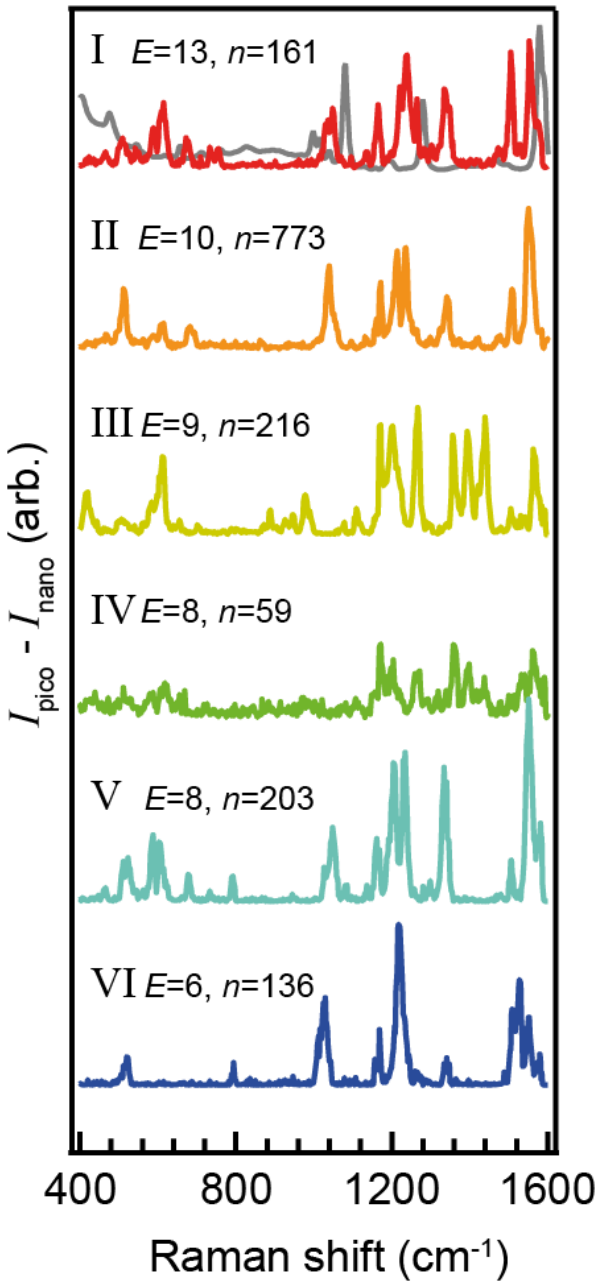


Figure 3.19: The most common Configurations extracted from the BPT testing subset, displayed in descending order of Event frequency labelled I-VI. The number of Events, E, and the total number of spectra, n, are also shown. The average spectra of each Configuration is plotted - alongside the global nanocavity that is overlaid onto Configuration I for reference to the prominent stable peaks.

Table 3.3: Reduction in the number of Events and spectra, and the subsequent improvement in the associated silhouette coefficient score, due to the averaging of 'flickering' Events assigned to the same original Configurations, and the removal of Events with negative silhouette sample scores. More impactful changes due to bias-correction are seen in alternate variations of the NPoM geometry seen in Subsection 3.3.7.

	Original	Corrected
Configurations	6	6
Total Events	56	54
Total Spectra	1548	1542
Silhouette	0.3154	0.3315

3.2.6 Peak Assignments and Near-Field Gradient Mapping

In collaboration with members of the Baumberg research group, the methodology employed in this subsection was developed to provide a chemical evaluation of the information extracted by the data analysis pipeline. Once the representative spectra from each Configuration have been obtained - by calculating the global average spectra from those contained within each member Event - the position of each metal protrusion (adatom) can be determined through a tentative assignment of the peaks detected in each Configuration spectrum to the vibrational modes predicted by the DFT algorithm. To confirm the validity of this process, the nanocavity spectrum was simulated using a commercial DFT package [202], which found good agreement with the stable nanocavity spectrum extracted from the global stable state reconstructed by the CAE (seen in Figure 3.19).

Within each scan it is common for any picocavity peak to drift in peak position over the duration of the Event in which it resides due to changes in adatom position (see Figures 3.9 and 3.15). As there are a limited number of vibrational modes available in the rigid BPT molecule, the range that each vibrational mode occupies can be estimated based on this peak drift. However, vibrational modes predicted by the DFT that have a low-intensity, or are positioned in close proximity to other vibrational modes, are not assigned to picocavity peaks due to the uncertainty in such an assignment. Examples of avoided peak-assignments are seen in part (A) of Figure 3.20 between $800\,\mathrm{cm}^{-1}$ to $1000\,\mathrm{cm}^{-1}$, in which there are two groups of three vibrational modes that are each within wavenumber ranges that are too small to be distinguished at the resolution of the data, and no single mode within either group has a Raman response that is intense enough as to be sufficiently more likely to correspond to the associated picocavity peak than its neighbours. For the sake of clarity, the predicted vibrational modes are numbered in the order of increasing energy.

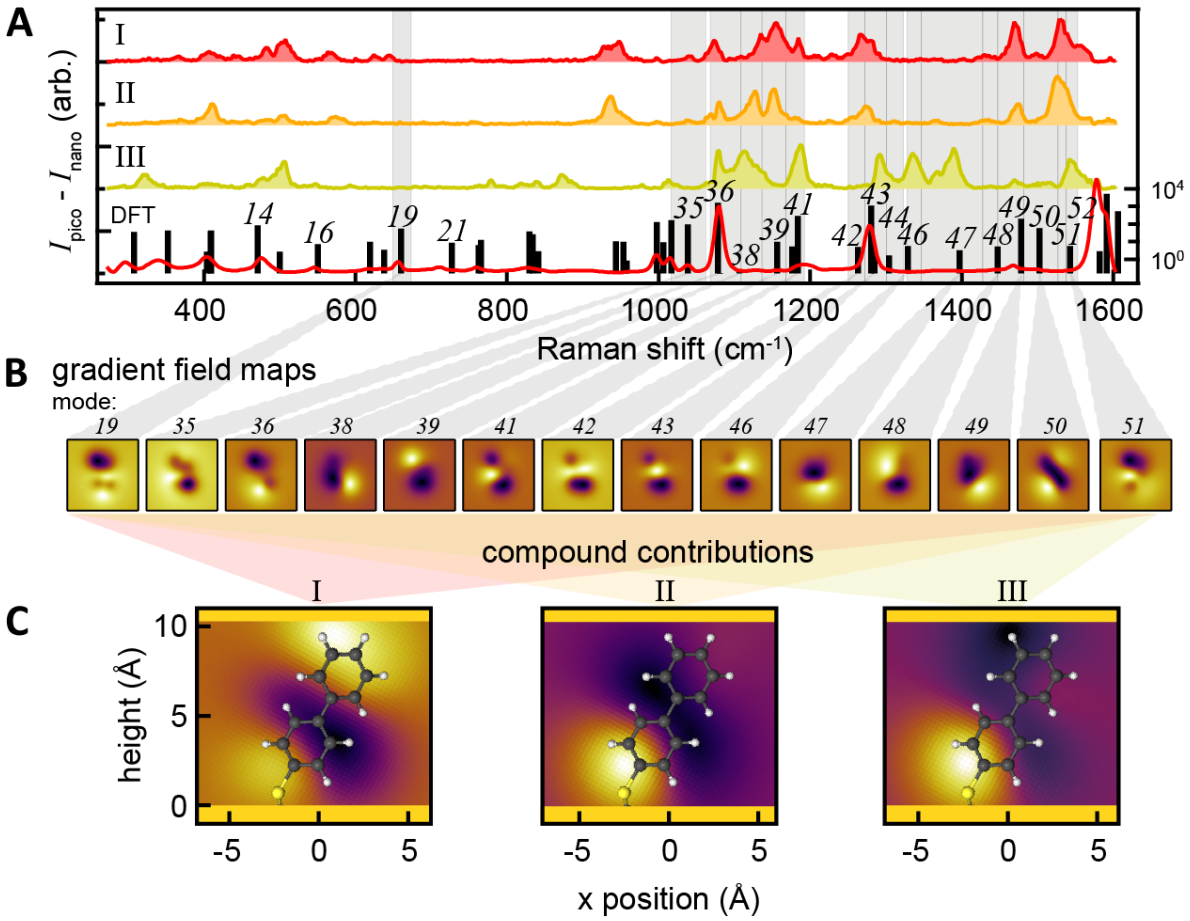


Figure 3.20: A) Peak range assignment for the three most common Configurations based on the stable state (nanocavity) DFT modelling of BPT. B) Gradient field maps showing the Raman response for each of the assigned vibrational modes (using a method adapted from Aizpurua *et al.* [203]). C) Near-field maps generated from compounded gradient-field Raman maps for Configurations I, II and III, plotted against a BPT-Au molecule depicted at a 29° angle from the surface normal [204].

In order to predict the most-likely position of the metal protrusion that caused a particular Configuration, the Raman response of a molecule in an inhomogeneous field must be calculated. Where standard DFT assumes a homogeneous field, an existing method developed by Aizpurua *et al.* [203] assumes an inhomogeneous field. Hence, this method was adapted to calculate the effects of a picocavity field gradient [205] as it is swept over the BPT molecule. This results in a map of Raman responses for each possible field position, and is repeated for each vibrational mode. These 'gradient field maps' visualise the different responses of vibrational modes to gradient fields across the molecule (see part (B) of Figure 3.20). Using this, a 'near-field map' is generated by averaging together the gradient field

maps for each vibrational mode that has been identified and matched with a corresponding picocavity peak within a Configuration. Each gradient field map is first multiplied by the corresponding peak area, and after the near-field map is generated it is normalised by the average of all gradient fields, in order to remove a systematic bias inherent to vibrational modes with a higher intensity response to the local field gradient. The equation to normalise the near-field maps is given:

$$\mathrm{M_c} = \frac{\sum \mathrm{A}_i \mathrm{M}_i}{\sum \mathrm{M}_i}, \tag{3.5}$$

where M_c is the compounded near-field Raman map A_i is the peak area corresponding to vibration i, and M_i is the gradient Raman map for vibration i. The resulting near-field maps provide an insight into the location where the field gradient originates from around the molecule, and can be used to tentatively assign the position of the atomic-scale feature giving rise to the localised field (e.g. adatom). Examples of the three most common Configurations are shown in part (C) of Figure 3.20.

3.3 Evaluation of Approach

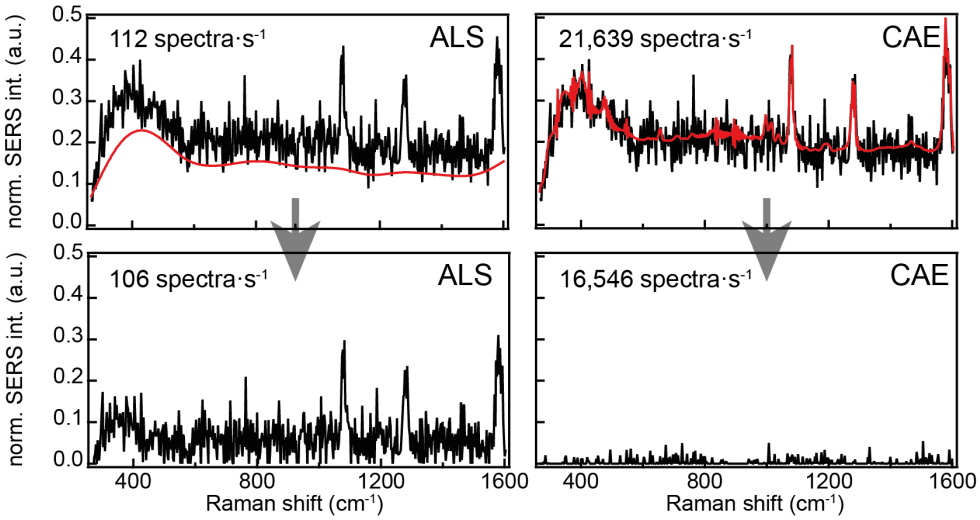
This section covers an evaluation of the combined machine learning and image processing analysis pipeline developed for processing, and subsequently analysing, the BPT database. The speed and efficacy of the stable state removal are compared to standard methodology; the effect of different normalisation methods on the performance of the CAE are reviewed; alternative methods for the formation of Tracks and Configurations, which were explored in the development of this process, are highlighted and compared to the chosen methods; the methodology and reasoning behind peak assignments to the DFT and experimental data are explained; and the robustness of the approach is demonstrated by fine-tuning the machine learning model on a new database containing additional NPoM varieties of BPT.

3.3.1 Performance of the Stable State Removal Process

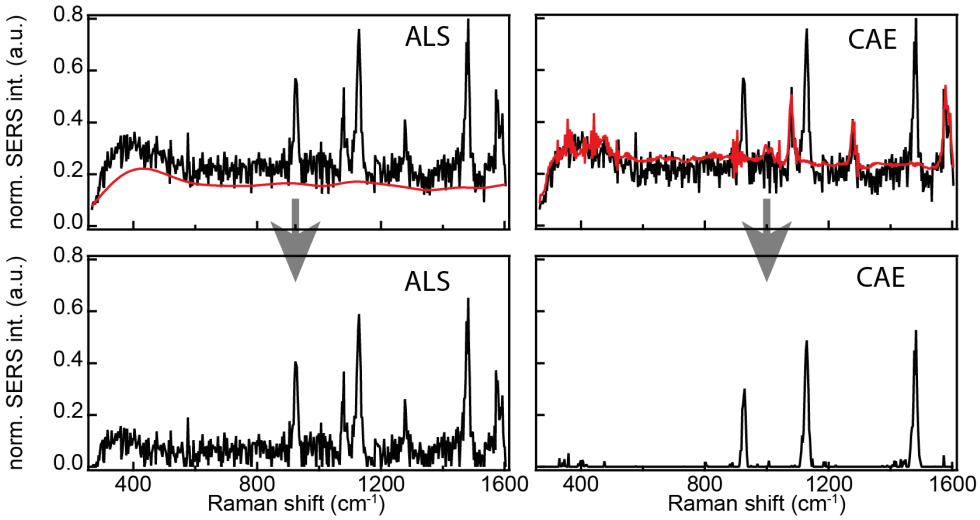
Figure 3.21 shows an example of a picocavity spectrum extracted by the CAE. Alongside this, a common spectral processing algorithm, asymmetric least squares (ALS), was used as a background fitting tool to compare the performance of both methods in terms of efficacy and speed. As the CAE was trained on stable state scans, it was able to reconstruct the peaks associated with nanocavity events and reconstruct them alongside the baseline spectrum, which was then subsequently removed using the formula for calculating the picocavity spectrum given in Subsection 3.2.1, Eq. 3.1. As the ALS background fitting method does not fit to peaks, only the baseline spectrum can be approximated, and as such ALS lacks the functionality to distinguish between stable state (nanocavity) and picocavity peaks. Hence, it proves an inappropriate tool for isolating transient features within SERS spectra.

With regards to throughput, the CAE allows for over 16000 spectra to be processed per second,

including the subtraction of the stable state, on a standard desktop computer, in comparison to ALS background fitting which is almost 200x slower. Figure 3.21a reports the exact values of this test.



(a) Background fitting (ALS) and stable state subtraction (CAE) of a stable state spectrum.



(b) Background fitting (ALS) and stable state subtraction (CAE) of a picocavity spectrum.

Figure 3.21: Comparison between ALS and the CAE. Background fit and stable state reconstructions, respectively, are shown in red. (a) ALS only reconstructs part of the baseline spectrum and does not remove stable peaks, whereas the CAE reconstructs the majority of the stable state. (b) The CAE reconstructs the stable state but not the picocavity event, allowing for isolation of the spectral properties, whereas ALS fails to separate the nanocavity and picocavity peaks.

3.3.2 Effects of Normalisation Methods on Model Generalisation

As mentioned in Subsection 3.1.3, layers in the CAE that were normalised were done so using group normalisation [168] rather than batch normalisation [163]. This was done so due to large variations in validation loss in the initial iterations of the CAE that used batch normalisation. Consequently, a test was performed in which all convolutional and FC blocks using batch normalisation were replaced with group normalisation, which was applied along the batch axis with a group size equal to the batch size.

This process is of near-equivalence to batch normalisation, with the exception being that batch normalisation uses batch statistics to normalise samples during training, and the population statistics learned from training to normalise samples during inference; however, group normalisation uses the batch statistics of each batch to normalise samples during both training and validation. This change was motivated by the fact that the data naturally comes in batches, hence the data variance captured by the batch statistics are strongly correlated to the scans, whether they are partially or fully contained within a batch.

This resulted in a reduced generalisation error of the model on the validation dataset, with suppressed oscillations in the loss curve. This can be seen in Figure 3.22 as the trend of the training and validation loss curves closely match when the CAE is trained using group normalisation, whereas in the batch normalisation model there are large variations in validation loss, and an overall increase in the validation loss indicating that the model is overfitting to the training dataset.

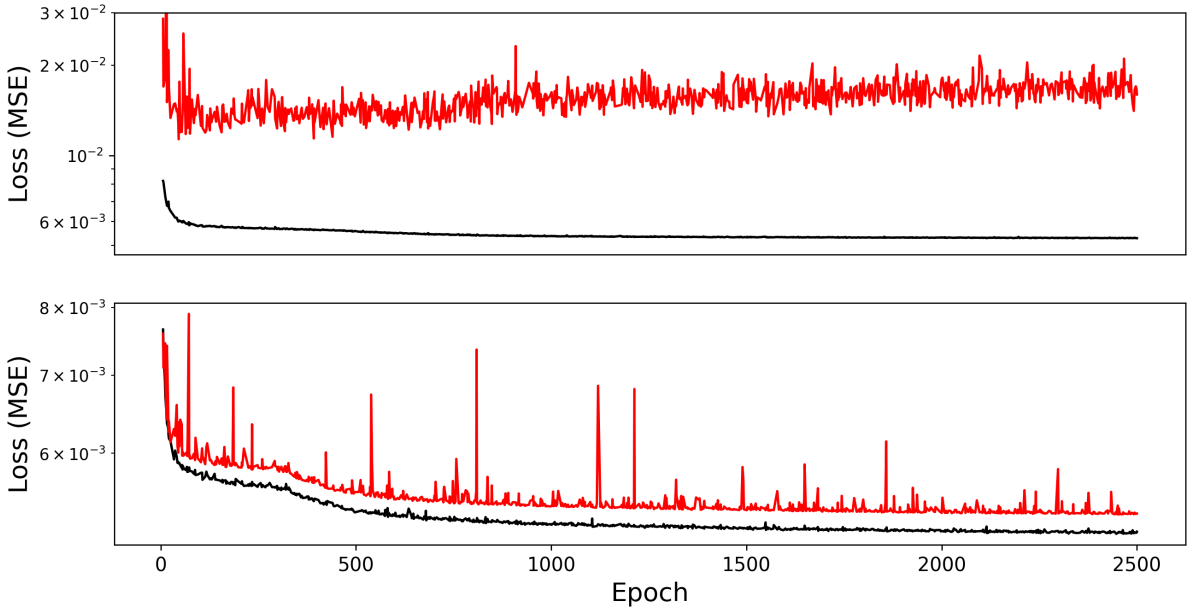


Figure 3.22: Effects of group normalisation versus batch normalisation on the CAE with all other hyperparameters fixed. *Top*, batch normalisation; *Bottom*, group normalisation. In both plots the training loss is shown in black, and the validation loss is shown in red. The use of group normalisation improves upon the overfitting of batch normalisation, along with an overall reduction in the magnitude of the validation loss. Note that the small reduction in training loss may be attributed to a difference in parameter initialisation.

3.3.3 Alternative Method to Track Isolation

Previous iterations of the Track isolation algorithm utilised Kalman filters [206] to form each Track, where new Tracks began at the earliest detected time step, and subsequent detected data points were added to each Track based on a search window with dimensions 3×10 px. The dimensions of the sliding window matched that of the Zipper algorithm described in Subsection 3.2.3. Peak detections were made for this method on a per-spectrum basis for pixel intensities exceeding a threshold of two standard deviations. Any adjacent bins (wavenumbers) in later time steps that had intensities above this threshold were fed to the search window as a single detection using the centroid wavenumber of that range, and were added to the associated Track. If detections in later time steps did not lie within the search windows of any 'active' Tracks (i.e. those that had received additions within the past 3 time steps), then new Tracks were formed with these most recent detections as the starting points.

As each transient peak naturally travels forwards in time (vertically, with respect to the spatiotemporal scan), the goal of the filter was to predict the wavenumber location of peaks that went undetected in later time steps - either due to algorithmic error, physical disruptions such as thermal fluctuations temporarily breaking down Events, or from flare events oversaturating spectra [185]. However, the

functionality of the Kalman filter proved superfluous, as the Tracks typically follow a predictable path through time, with only minor deviations due to the aforementioned thermal effects, or through adatom movements perturbing the local electric field (as seen in the various figures throughout Sections 3.1 and 3.2). In addition, transient peaks have been observed to temporarily bifurcate for a small number of time steps, from which the Kalman filter process would create two separate Tracks instead of one, as it lacks the functionality to incorporate this physical effect; in contrast the 'connectedness check' that is capable of dealing with these bifurcations.

Hence, this method of forming Tracks using Kalman filters was replaced by the considerably simpler connectedness operation, and the iterative Zipper algorithm to bridge small gaps between otherwise contiguous Tracks (described in Subsection 3.2.3). This change was jointly motivated by efforts to reduce the number of arbitrary parameters that required tuning in the development of the data analysis pipeline, and to decrease computation times. Where the Kalman filter operation required spectra to be processed sequentially, the connectedness operation was applied scan-wide, thus decreasing computation times dependent on the number and complexity of picocavities present on each scan.

3.3.4 Event Formation Threshold Tuning

The Event threshold value of 0.7 was empirically determined using the same representative subset of BPT data that was used to form and analyse the Configurations seen in Subsections 3.2.5 and 3.2.6, respectively. However, it should be noted that coexisting picocavities are rare in the BPT dataset, and as such the tuning of this parameter was not critical. However, should a database be analysed with more regular coexisting picocavities, the Event threshold may need to be more thoroughly examined, whereby higher values would ensure that distinct picocavities are assigned to separate Events. This would in turn benefit from higher SNR data and a more robust data analysis pipeline that would be capable of isolating Tracks to a greater level of completeness, to counteract the increased potential of an incorrect Event assignment.

3.3.5 Alternative Configuration Clustering Methods

Two other clustering algorithms were considered alongside Spectral clustering, which shall be referred to as the 'Champion' method. The first method, termed the 'Integral' method, implemented a simple peak-matching check. This was motivated by an attempt to avoid distinct, coexisting picocavities from being incorporated into the same Event. Before being compared, Events were first converted into 'Event vectors', which were produced through the summation of each detected pixel across the duration of the Event, and smoothed using a 3rd-order polynomial to account for inaccuracies in the mean peak position due to spectral drift (see Figure 3.23). Mathematically, the similarity between spectral pairs was calculated as the ratio between the integral of the absolute difference between two Event vectors, against the summation of the individual vectors,

$$r(A,B) = 1 - \frac{\sum_{\lambda=1}^{N} |A_\lambda - B_\lambda|}{\sum_{\lambda=1}^{N} A_\lambda + \sum_{\lambda=1}^{N} B_\lambda}, \tag{3.6}$$

where A_λ and B_λ are two arbitrary Event vectors, and λ represents the wavenumber axis. The ratio (similarity) is equal to 0 if two Events have no matching peaks and 1 if the two Events are identical. Through a heuristic analysis on a subset of BPT data, a similarity threshold of 0.35 for the ratio was found to produce the best clusters (Configurations). Hence, any two Events with a ratio at or above 0.35 would be clustered into the same Configuration.

The advantage to this method is that, by forming Event vectors exclusively through the summation of detected (isolated) transient peaks, co-existing Events could be separated into different Configurations. However, co-existing Events are rare within the BPT dataset, thus this was not a strong consideration when selecting the appropriate clustering method. Contrary to this, the main drawback to this method was that it enabled the formation of long Configuration 'chains', wherein dissimilar Events would be placed into the same Configuration due to shared similarities to one or more additional Events. In addition, as the number of peaks in an Event increased, the average number of missed or incorrectly detected peaks within an Event would increase in turn, this consequently reduced the similarity between two otherwise identical Events. The larger number of peaks also increased the similarity between two Events that seem distinct through a visual inspection (see Figure 3.23). Based on these drawbacks, the Integral method was discarded, as tuning the similarity threshold was highly dependent on the quality of the data, and on the overall effectiveness of the data analysis pipeline at fully isolating each picocavity Event.

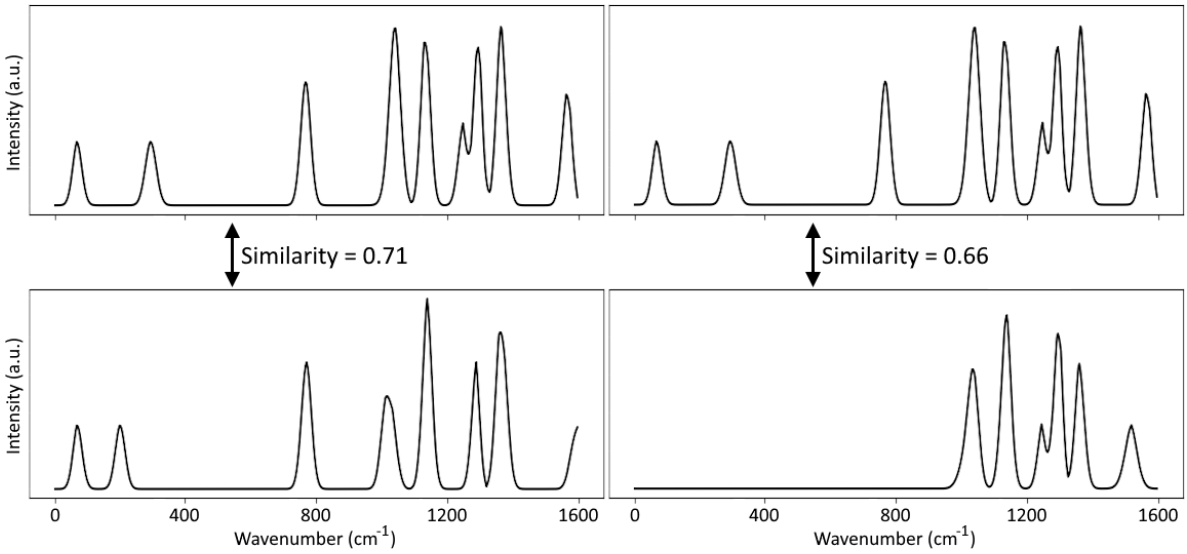


Figure 3.23: Two examples of the Integral clustering method. Event vectors are formed from the sum of all detected peaks and smoothed using a 3rd-order polynomial. *Left*, two Events are compared with similar representative spectra; there are 7 peaks that match through a visual inspection, and the score of 0.71 shows a reasonable similarity despite the mismatched peaks. *Right*, the same Event (*top*) compared to another, generating a similarity score also above the threshold, meaning that these Events would be clustered despite the second Event having no Raman peaks at lower wavenumbers.

The second alternative clustering method - referred to as the 'Pyramid' method - was considered for the possibility of expanding the application of the data analysis pipeline to complex molecular datasets where coexisting Events are commonplace. This method was based off of a binary peak-matching task using 'spatial pyramids' [207, 208], which partitioned the feature space into successively finer sub-regions, attempting to match together peaks in sets of histograms produced by each sub-region. This could then be used to form a similarity matrix - in much the same way as the Champion method - and clustered using kernel k-means.

This technique used the knowledge of which peaks belonged to which Event by using Event vectors - as in the Integral method - and could therefore exclude peaks belonging to other coexisting Events. However, by clustering using kernel k-means, it was able to overcome the issue of 'chaining' dissimilar Events together based on a static similarity threshold to form Configurations. The result of this clustering method, using a number of clusters optimised using the best silhouette coefficient score, achieved a lower silhouette score than the Champion method, and thus was not selected as the preferred method. However, if a dataset were being analysed in which coexisting transient events, or multiple picocavity devices, are common, then the Pyramid method may be a more suitable option.

Amongst the evaluated clustering techniques, the Champion clustering method achieved the best silhouette coefficient score due it leveraging the inherent rarity of coexisting picocavities found in the

BPT dataset. By utilising the mean picocavity spectrum associated with each isolated transient event, this method ensures that any undetected peaks during the initial peak isolation step are factored into the subsequent clustering process. It should be noted that the clustering methods were compared, and the Champion method was selected, before the clustering process was adapted to account for 'flickering' due to the incorporation of addition NPoM varieties (detailed in Subsection 3.3.7 to come). Based on this earlier version of the clustering process: the Integral clustering method achieved of silhouette score of 0.2117 with 4 clusters; Pyramid clustering achieved a silhouette score of 0.1841 with 9 clusters; and the Champion clustering achieved the best silhouette score of 0.2469 with 8 clusters. As shown in Subsection 3.2.5, Table 3.3, the latest iteration of the Champion method achieves a silhouette score of 0.3315 with 6 clusters on the original BPT dataset.

3.3.6 Picocavity Analysis and Comparisons to DFT Predictions

Vibrational modes calculated by DFT are tentatively assigned to peaks present within a Configuration spectrum. As mentioned in Subsection 3.2.6, the method for making these assignments is based on a number of factors: firstly, due to the limited number of vibrational modes available to a rigid BPT molecule, the drift in peak position over an Event duration is used to dictate the bounds that a vibrational mode can be said to reside in (see Figure 3.24).

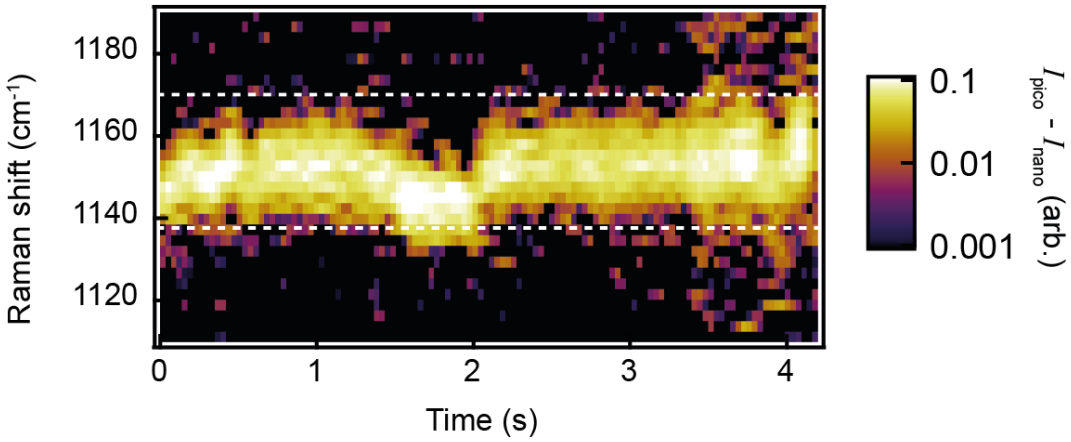


Figure 3.24: Typical peak drift observed within a scan. Shown here is a picocavity peak assigned to vibrational mode 39, with white dashed lines demarcating the upper and lower integration bounds that are used in the consideration of possible peak assignments.

Another consideration for peak assignments is peak prominence, in which the intensity of a vibrational mode must be both uniquely intense and sufficiently greater than neighbouring modes (if any). For example, modes 46, 47 and 48 in Figure 3.25 were each assigned to relatively weak peaks in the background-subtracted nanocavity spectrum on account of their large, isolated intensities, whilst mode

43 was assigned to a peak despite being in close proximity to mode 42, on account of the substantially greater amplitude relative to the neighbouring mode. One final consideration is peak location; vibrational modes in the lower wavenumber range were avoided as DFT is notoriously poor in accurately predicting low wavenumber (weaker) vibrations. For example, modes 9 and 10 in Figure 3.25 appear as though they are a good match for the peaks above them, however due to the weaker bond strength these modes, when perturbed by the picocavity, are likely to move significantly from the predicted position, making the subsequent picocavity assignment more difficult. Additionally, as there is a lesser degree of field enhancement in that region the SNR is lower. It was determined that vibrational mode number 14 was the earliest viable candidate, since the calculated peak was both relatively strong and isolated.

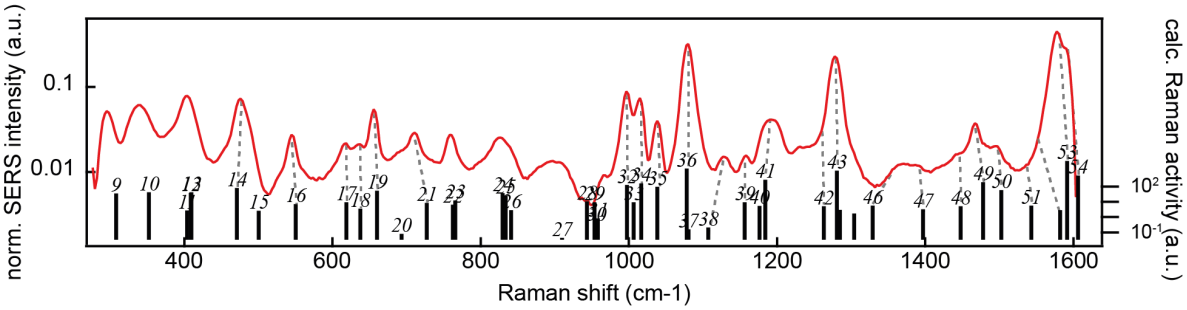


Figure 3.25: Background-subtracted global nanocavity spectrum and calculated DFT vibrational modes plotted on a log-scale to visualise both large and small peaks for tentative peak assignment.

Once the near-field maps for each Configuration had been calculated, the location of the metal protrusion (adatom) that produced the local field gradient could be predicted, and a comparison could be made between experiment and theory. For Configuration I (the largest cluster: 13 Events, 161 spectra) the near-field map suggests a picocavity that has arisen near the nanoparticle (NP type) whereas all 5 other Configurations, with a combined 41 Events made up of 1387 spectra, indicate picocavities that have formed near the substrate (see Figure 3.26). These findings show that the adatoms giving rise to the picocavity events are most likely to form on the substrate (90% of spectra), which is in line with previous observations using molecular spacers with an upper cyanide group which forms a distinguishing Raman marker (85%) [145]. However, 24% of Events are classed as coming from the nanoparticle showing that substrate Events contain on average more spectra - 34 per substrate versus 12 per nanoparticle - and that therefore these Events persist for longer.

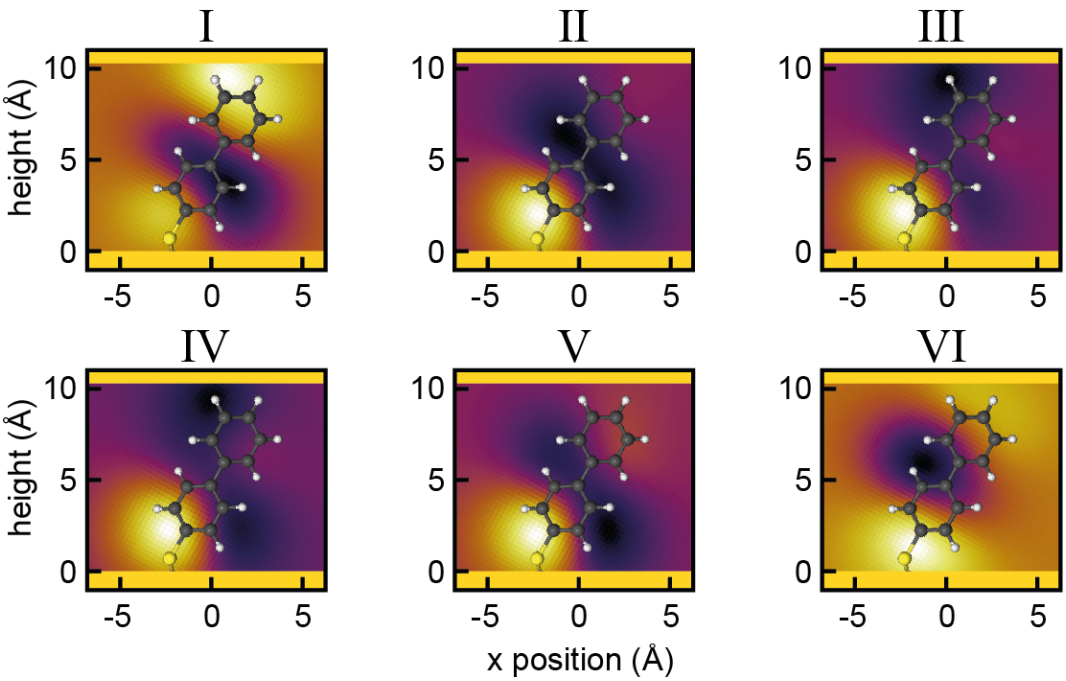


Figure 3.26: Near-field maps for the BPT NPoM geometry using commercial gold nanoparticles (Au-Au) showing one NP type and five substrate type picocavities.

3.3.7 CAE Fine-Tuning and Analysis of Additional NPoM Varieties

To verify the validity of the findings in Subsection 3.3.6, and to test the robustness of the approach, a second database was prepared. This database was captured using a second bespoke Raman microscope with a higher spectral resolution - resulting in narrower peaks - using longer integration times (time steps) of 200 ms compared to 35 ms, and in-house synthesised gold nanoparticles instead of the commercial-grade nanoparticles used in the previous database. For this spectrometer, an Olympus BX51 microscope was coupled to a Horiba Triax 320 dispersive spectrometer with a 600 lines/mm grating and an Andor Newton 970 BVF electron-multiplying CCD. The spectra were collected in the same manner as the previous BPT database, using an Olympus NA (0.9) 100x darkfield objective.

To further validate the assignment of nanoparticle versus substrate picocavities, two additional NPoM varieties were prepared, where one of the metal surfaces was functionalised with a monolayer of palladium that was grown onto that surface. The palladium was found to suppress the formation of picocavities either on the nanoparticle or the substrate [1, 209], in line with predicted adatom formation energy costs [210]. These sample varieties are here labelled by the nanoparticle and substrate metal (M) compositions using the format M_{NP}-$M_{substrate}$. Thus, the three new databases are as follows: Au-Au, Au-AuPd and AuPd-Au - for example, Au-AuPd represents strictly gold nanoparticles,

with a monolayer of palladium deposited onto the gold substrate surface (see Figure 3.28). Unless stated otherwise, any reference to the Au-Au database within this subsection assumes the later version produced from in-house nanoparticles, rather than the original with commercial-grade nanoparticles and shorter integration times (time steps).

The new database contains 1833500 spectra over 3667 scans - hence 500 spectra per scan - of which 3479 scans contain picocavities and 188 contain only nanocavity signals. The combination of longer integration times and in-house nanoparticles caused fewer scans to consist only of nanocavity spectra. The nanocavity scans were partitioned into 144 scans for the training dataset and 44 for the validation dataset. Due to the limited number of stable nanocavity spectra split between the three new NPoM varieties, which constituted an approximate 14-fold reduction in the number of training samples, the existing pre-trained CAE parameters were fine-tuned on the nanocavity scans within the new training dataset. Fine-tuning on the existing CAE also demonstrates the capability for the model to learn the stable state properties of three separate datasets simultaneously. Despite a limited number of stable nanocavity spectra used, the algorithm readily adapted to the spectral properties of the new data, as the MSE loss of the fine-tuned model showed a downwards trend similar to the initial BPT database during pre-training (Figure 3.27).

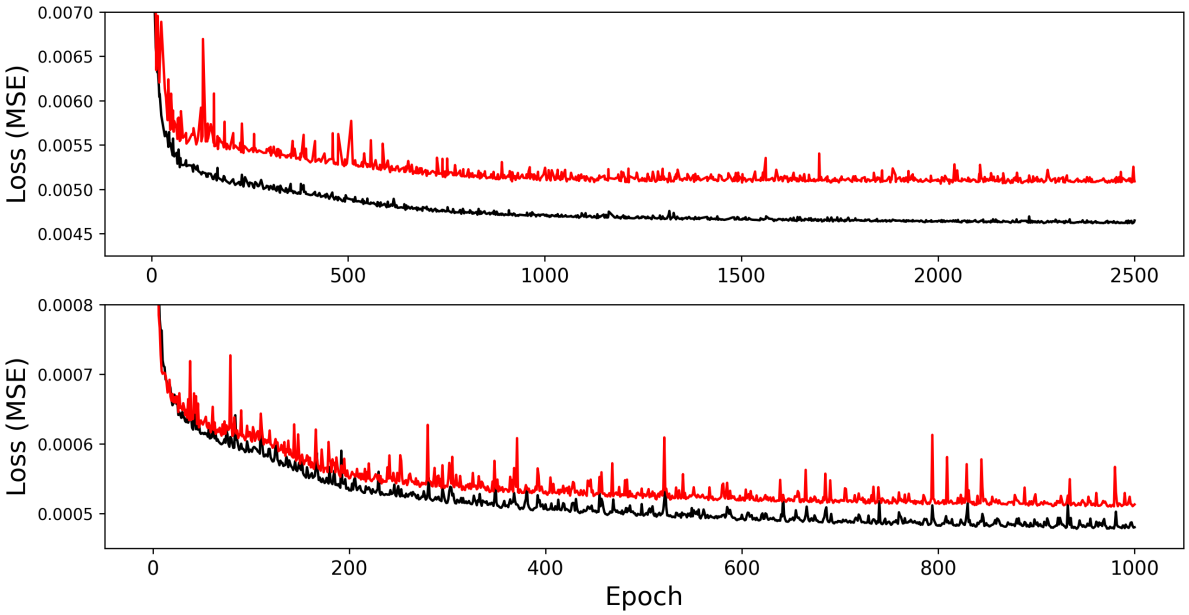


Figure 3.27: *Top*, pre-training on the original BPT database; *Bottom*, fine-tuning on the new NPoM varieties. In both plots the training loss is shown in black, and the validation loss is shown in red. The overall overfitting of the model improved by fine-tuning, as the average relative increase in validation loss proportional to training loss reduced from around 10% to 5%, showing a more generalised model. Note that the scale of the fine-tuning loss is approximately one order of magnitude lower than that of pre-training, though this could be to some extent attributed to the database itself as well as due to the nature of fine-tuning the CAE model.

As mentioned at the end of Subsection 3.2.5 (see Table 3.3), the flickering bias-correction applied to the initial Events that are formed resulted in a greater reduction of the number of Events using the new dataset. This is summarised in Table 3.4 below, which showcases an improvement to the global silhouette scores for both the Au-Au and AuPd-Au datasets, and a minor decline for the Au-AuPd dataset.

Table 3.4: Effect of flickering bias-correction on Configuration sizes for the three additional datasets. There is a greater reduction in the number of Events and spectra in comparison to the original BPT dataset.

	Au-Au		Au-AuPd		AuPd-Au	
	Original	Corrected	Original	Corrected	Original	Corrected
Configurations	6	6	6	6	10	10
Total Events	183	72	139	50	119	63
Total Spectra	5551	5481	4344	4043	2308	1568
Silhouette	0.2784	0.2953	0.3121	0.3009	0.1921	0.2656

After training, each of the three NPoM varieties were assessed separately generating 6, 6 and 10 Configurations after independent clustering of the Au-Au, Au-AuPd, AuPd-Au samples respectively. The spectrum of the most common Configuration for each NPoM variety is shown in Figure 3.28, with Figure 3.29 showing the near-field maps of the 3 most common Configurations for each (all other near-field maps are shown in Figures 3.31, 3.32 and 3.33). The in-house Au-Au NPoM shows Configurations I, II, IV and V are suggestive of substrate picocavities (totalling 53 Events over 4511 spectra) with the nanoparticle picocavities now split over Configurations III and VI (19 Events, 976 spectra). This implies 18% of picocavity spectra come from the nanoparticle with 26% of Events classed as NP type, which is in close agreement with the previous observations in Subsection 3.3.6, and in literature [145].

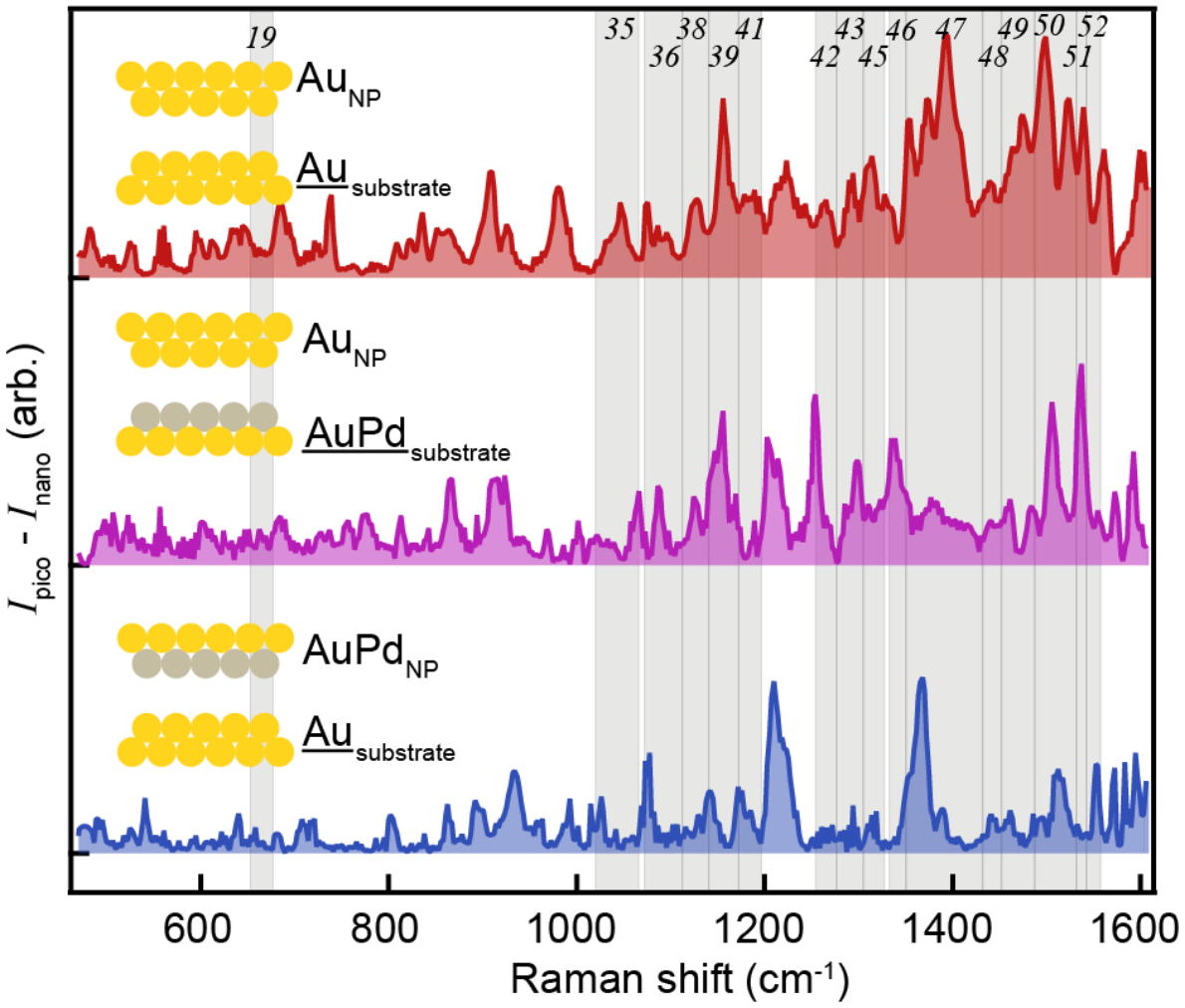


Figure 3.28: Tentative vibrational mode assignment to picocavity peaks within the most common Configurations for each of the three new sample types using in-house nanoparticles. The inset illustrations show the palladium monolayer (or lack thereof) deposited onto the specified metal surface.

Interestingly, when a palladium monolayer is introduced onto the substrate, all but one Configuration (VI) shows picocavities forming from the nanoparticle (see Figures 3.29 and 3.32), with one containing a mixture of nanoparticle and substrate contributions (IV). This results in 90% of picocavity spectra now originating from the nanoparticle and 88% of Events (excluding the mixed configuration IV). In addition, in contrast to the previous observations, more spectra are observed on average per Event for those coming from the nanoparticle versus from the substrate - 92 per substrate versus 72 per nanoparticle. For the sample type where the nanoparticle is coated in a monolayer of palladium, 10 clusters (Configurations) are formed. Of these, 7 Configurations show substrate Events and 3 show

nanoparticle Events (see Figures 3.29 and 3.33) with only 19% of Events now coming from the nanoparticle (an approximate 5% drop with respect to Au-Au samples). Interestingly, the average spectra per Event for the NP types greatly increased, with an average of 47 spectra per Event for the NP types vs 20 for the substrate types.

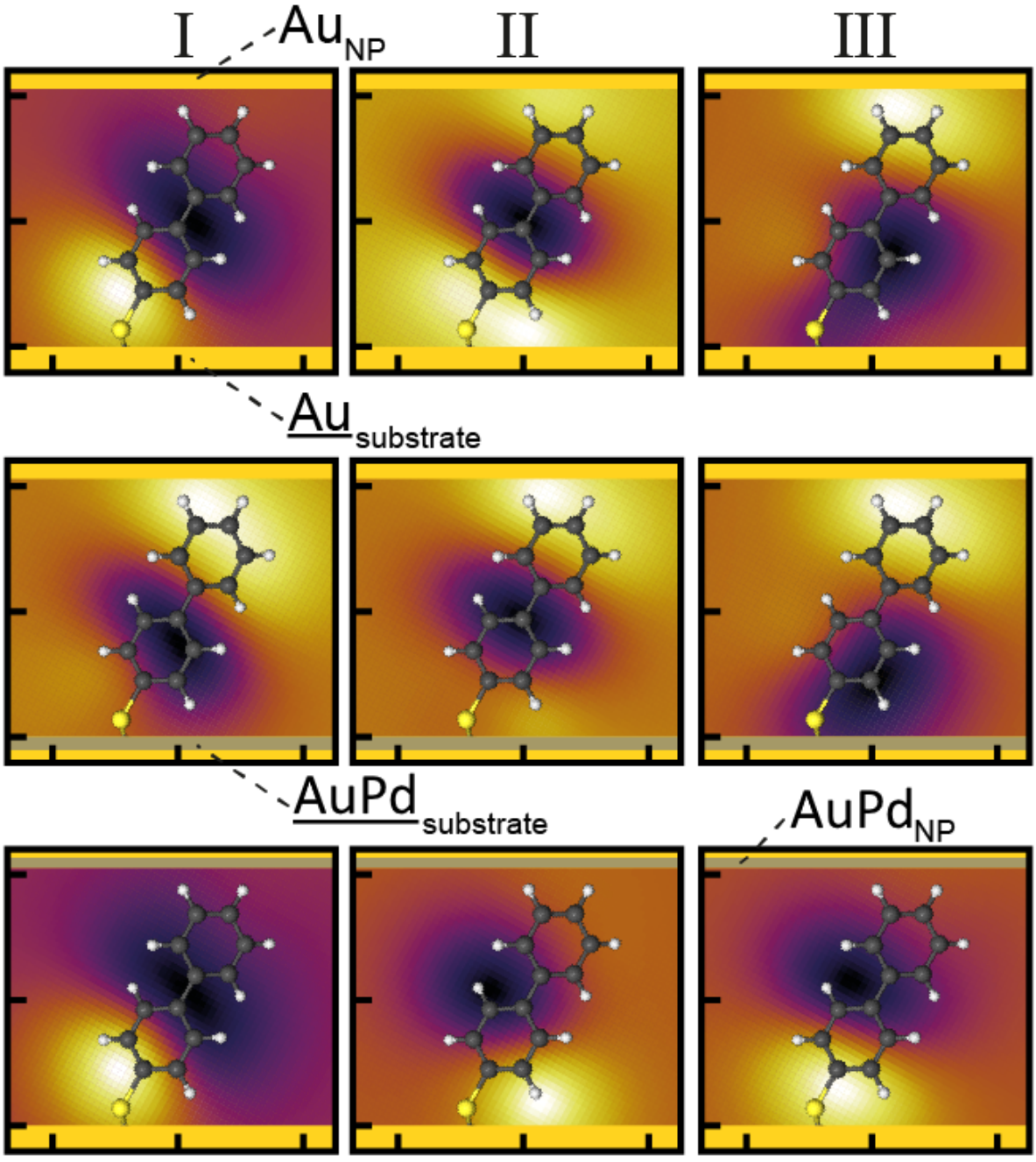
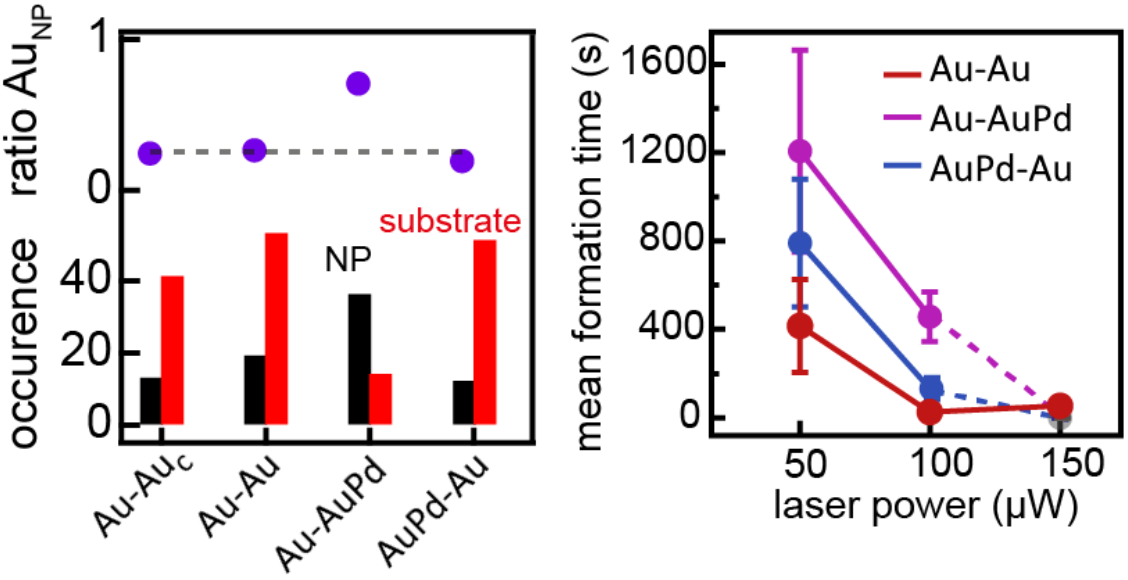


Figure 3.29: Near-field maps showing adatom positions for the three most common Configurations of the additional sample types. This highlights the role of a metal surface functionalised with a monolayer of palladium (grey region), deposited onto the respective surface, in suppressing the formation of picocavities from that surface.

Overall, comparing the Events for each of the sample varieties clearly shows a strong effect from coating either surface with a palladium monolayer. This suppresses the formation of adatoms on the newly functionalised surface (Figure 3.30a). To confirm this suppression, the mean picocavity formation times for all three variants are compared at different laser powers. This shows a similar trend, with Pd-coated substrates having the strongest effect on the formation rate - *i.e.* longer mean formation time (Figure 3.30b) - agreeing well with predictions in literature [210]. The Au-Au sample type with commercial-grade nanoparticles is excluded from the comparison of formation times in Figure 3.30b due to fundamental differences in the rate of picocavity formation between commercial and in-house nanoparticles (explained earlier in 3.3.7).



(a) Occurrence and ratio of each sample type for adatom formations on the substrate *vs* nanoparticle.

(b) Mean formation times of each sample type, for each Configuration, as a function of laser power.

Figure 3.30: Effects of gold surfaces functionalised by palladium. a) Sample types for both the commercial (denoted 'c') and in-house nanoparticles share similar proportions, with the palladium-deposited surface variants either switching (Au-AuPd) or strengthening (AuPd-Au) the occurrence between each type. b) Mean formation times are lengthened with the addition of palladium onto either surface - though the effect is stronger for lower laser powers, and for the functionalised substrate.

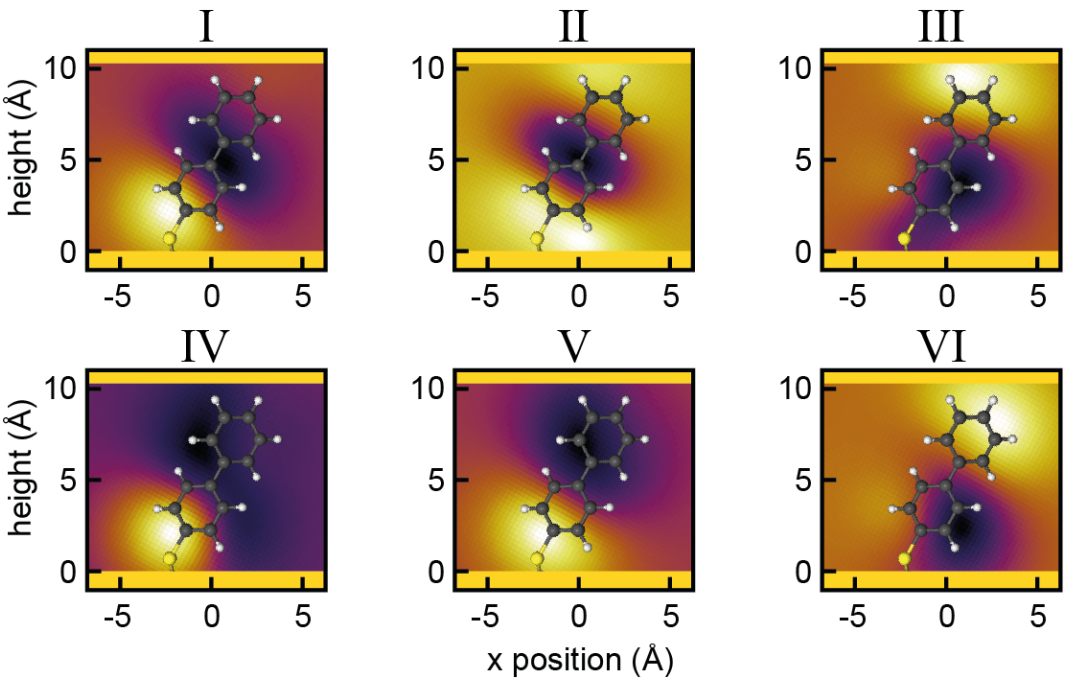


Figure 3.31: Near-field maps for Au-Au NPoM sample using in-house nanoparticles show two NP type and four substrate type picocavities.

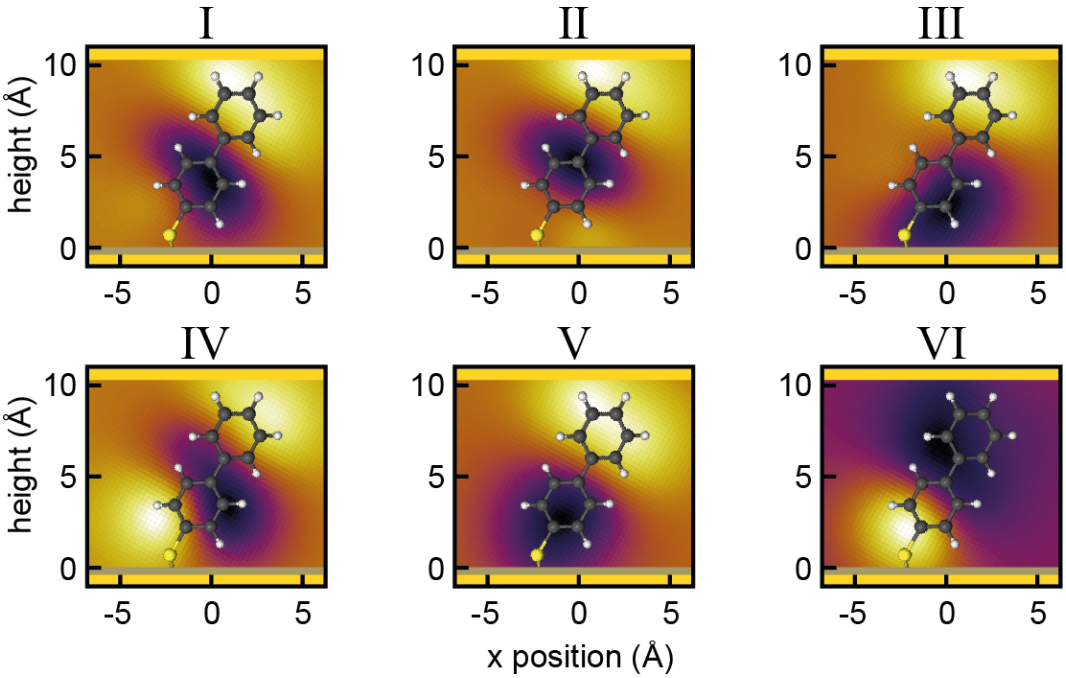


Figure 3.32: Near-field maps for Au-AuPd NPoM sample using in-house nanoparticles show four NP type, one substrate type and one mixed type (IV) picocavities, with the three largest Configurations coming from the nanoparticle.

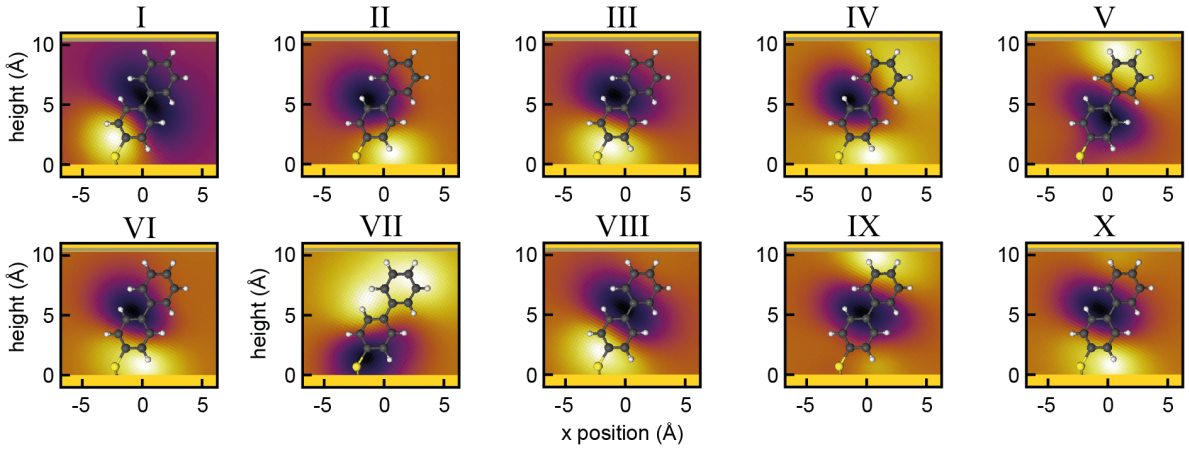


Figure 3.33: Near-field maps for AuPd-Au NPoM sample using in-house nanoparticles show three NP type and seven substrate type picocavities, with the three largest Configurations coming from the substrate.

3.4 Discussion and Conclusions

In the work presented here, labelled data are used to determine whether a scan contains transient (picocavity) events - although this could be further delineated to single spectra. This distinction is a requirement resulting from the CAE training process, in which the dataset is partitioned in order to facilitate the CAE reconstruction process and subsequently remove the stable state signal from each picocavity spectrum. Discussed within this section is the possibility for a machine learning architecture to circumvent the labelling requirement, which would provide multiple benefits: firstly, a practical hurdle would be overcome in needing an analytical chemist to manually label scans in the manner described, or to avoid the development of a potential automated labelling process; secondly, this would allow for more complex molecules to be analysed, and thus extend the capability of the data analysis pipeline developed in this study.

A promising new machine learning architecture, the transformer, has proven a powerful tool for state-of-the-art classification in spectroscopy tasks [211, 212], due to its self-attention mechanism - where 'soft' weights are used that are able to change during inference, as opposed to conventional 'hard' weights that are fixed after training. This mechanism may be the key to circumventing the label requirement. Transformers employ encoder-decoder architectures, and can utilise convolutional layers, hence drawing many similarities to the CAE model used in this research. Frequently deployed in the fields of natural language processing and computer vision, transformer architectures are capable of processing hyper-spectral Raman data and can incorporate the complete spatiotemporal information of SERS spectra in parallel. This would mean that SERS scans would be processed as images rather than individual spectra. This may enable long-term dependencies (picocavity events) to be factored into classification outputs, thus lending credence to the applicability of this architecture.

Another analyte molecule was initially considered for use in this study, MPA (methiopropamine), for which a database of SERS scans existed; measured using the same in-house experimental setup as described in Subsection 3.1.1. MPA is a complex molecule with many possible vibrational modes and no consistent stable state. As such, no labelled data existed to distinguish between stable state and picocavity scans, which is a requirement of the CAE training process, and subsequently the data analysis pipeline proved incapable of isolating the transient peaks from the background and nanocavity peaks. Future work on the analysis of such a complex dataset may be possible with the adaption of the data analysis pipeline to a transformer architecture, which could incorporate higher complexity molecules with stable states that are difficult to determine - if they exist at all.

Through convention, a typical silhouette score of 0.5 or higher is indicative of a good clustering result. However, a sufficient score is dependent on the characteristic features of the dataset, and the specific application needs of the domain. In the case of the SERS data, a wide range of distinct Configurations have been identified that possess only one (or very few) members, whilst other Configurations are much more numerous. This complexity gives rise to a challenging clustering task that requires the

formation of clusters with vastly different sizes in order to incorporate this diverse set of picocavities. As demonstrated in Subsections 3.3.6 and 3.3.7, the Champion clustering method formed clusters that found good agreement with results predicted by DFT on each of the BPT datasets.

Two key areas for potential improvements to the clustering process warrant future investigation beyond this study. The first being a review of alternatives to the silhouette coefficient score as the metric for determining an appropriate number of clusters, as this metric is global and therefore lacks the nuance to summarise the relative strengths of each cluster. Although individual clusters scores are also accessible, the process of determining an appropriate number of clusters through such a manual inspection would be labour-intensive and promote subjectivity. The second area for improvement is exploring the clustering method itself; one alternative not explored in this study is agglomerative hierarchical clustering [213], which would allow for either an automated clustering method through cutting at an optimal region of a dendrogram representing the clustered samples, or reviewing the clusters (Configurations) resulting from manual cuts. However, this would be time-consuming, and involve a large degree of trial-and-error for an analytical chemist. Manual cuts, or designing a dynamic cutting process, might allow for a diverse range of cluster sizes that correlate with the complexity of the SERS data. Another alternative to clustering worth investigating is density-based clustering, such as DBSCAN [214], which does not require a number of clusters to be specified, thus making it suitable for unsupervised data processing tasks. However, DBSCAN requires two other parameters to be specified relating to: the minimum number of samples to form a cluster, and a density parameter determining the maximum distance between samples to be considered neighbours.

These improvements could lead to analysing greater quantities of the picocavity scans simultaneously. An observation of the Champion clustering process, which limited the number of picocavity scans evaluated, was the steady decrease in the silhouette score as the number of clusters increased beyond approximately 10 to 14, based on the results of several tests. This observation is counter-intuitive to the idea that greater quantities of SERS data possess a greater diversity in picocavities, which motivates future exploratory research into alternatives to the Configuration clustering strategy.

Where conventional DFT simulations aim to accurately estimate Raman spectra from homogeneous electric fields, two problems were encountered within this research to assign peaks between representative picocavity spectra within each Configuration and simulated DFT peaks: firstly, picocavity spectra contain additional peaks due to the excitation of typically Raman inactive modes as a result of strong local electric fields produced by adatoms, this required modifications to the DFT using a gradient Raman approach (described in Subsection 3.2.6); secondly, there was an intensity difference in both absolute and relative terms with respect to sets of peaks in the picocavity spectra, which required the same DFT adaptation. However, DFT remains incapable of accurately recreating the SERS spectra used in this research, which limits the degree by which DFT can be justified as a comparative tool. A comparison to DFT was still made however, as there was a reasonable confidence from the Baumberg research group that there exists only one type of molecule (BPT) in each nanogap, meaning that the

representative picocavity spectra must approximate DFT simulated spectra. From this, the tentative peak assignments are made, focusing on more intense peaks that are in close proximity between DFT and picocavity spectra.

As described in Subsections 3.2.6 and 3.3.6, the peak assignment was manual in nature. An automated method was attempted both to remove subjectivity from this process, and to allow for batch processing of spectra, which is a crucial factor when considering the large quantities of data able to be captured. This automated process used a Gaussian fit to each picocavity peak to obtain a central wavenumber position, then if this position was within some prespecified distance of a DFT mode then a peak assignment was made. This Gaussian fit method was prone to failure, however, as nearby peaks would interfere and produce non-Gaussian fits, which would require deconvolution to accurately resolve peak positions. The threshold distance to make an assignment would also require tuning, thus failing to remove an aspect of subjectivity present in the manual method, as this threshold may vary between different spacer molecules. Ultimately, a manual peak assignment was made due to the confidence of a single molecule type within each nanogap, thus leaving room for improvement within this research area.

To conclude, a robust method to extract salient features from SERS spectra has been developed and introduced, which has been used to isolate and cluster picocavity spectra. It is also shown that, by adapting an existing inhomogeneous field Raman mapping method, a tentative position for adatoms can be extracted. Using this method it is found that the formation rate, location and lifetime of picocavities can be influenced by functionalising either the substrate or the nanoparticles with a monolayer of palladium atoms. This now provides a unique insight into the formation behaviour and the coordination geometries of adatoms in metal surfaces. This technique will translate to many other analyte molecules, as long as stable state spectra can be acquired for training purposes. Thus it is believed the combined machine learning and image processing analysis pipeline introduced here offers a powerful tool to assist in the rational design of heterogeneous catalysts.

Chapter 4:

Temporal Extension to the Metal-Molecule Analysis Pipeline

Contents

Previously the capability was shown for the robust data analysis pipeline to efficiently extract information from single molecule SERS spectra, arising from interactions between adatoms and BPT molecules. The aim of this chapter is to build a complementary tool to resolve metal-molecule interactions, thereby extending the work done in Chapter 3, starting from the Event formation stage. Figure 4.1 depicts a flowchart summarising these additional processes.

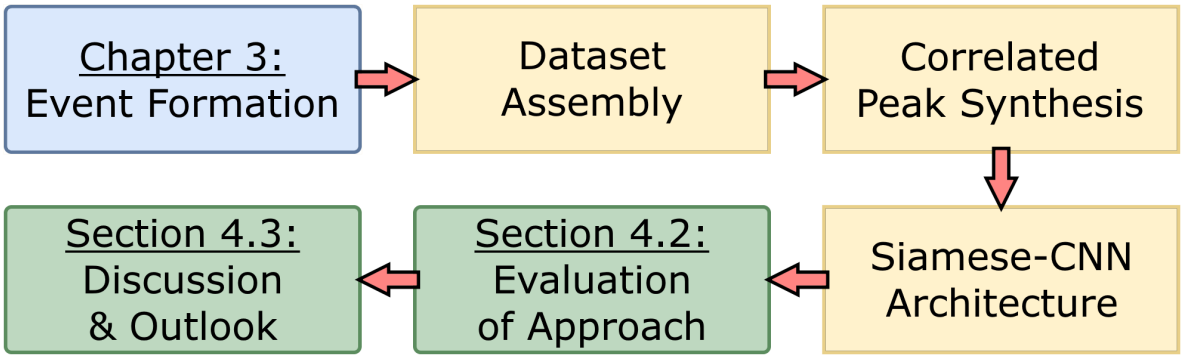


Figure 4.1: Flowchart depicting an extension to the data analysis pipeline focused on the analysis of temporal information in the initial BPT SERS dataset. The data processing steps (yellow) are described in Section 4.1, followed by an evaluation of the approach in Section 4.2, and a discussion of potential research pathways in Section 4.3.

Another desirable aspect of single molecule SERS data that has not yet been investigated is how adatoms interact with specific molecular bonds. These metal-molecule interactions are often the first step in heterogeneous catalysis, but they are often poorly understood on a fundamental level [215, 216, 217]. Developing a better understanding of metal-molecule interactions on an atomic level can aid in the intelligent design of heterogeneous catalysts. Often, such an understanding is derived from an in-depth modelling of chemical processes (such as through DFT), but few tools allow for the direct investigation of these interactions.

A simplified view of heterogeneous catalysis research involves understanding interactions between molecules and metals, and how such an understanding can promote advancements to industrial catalytic processes. Two incredibly important examples of catalysis are the Haber-Bosch process and the carbon dioxide (CO_2) reduction reaction, both of which highlight the interest in such advancements.

The Haber-Bosch reaction is an industrial chemical process used to produce ammonia (NH_3) from nitrogen (N_2) and hydrogen (H_2) gases [218]. It is one of the most important chemical processes in the world due to its significance to agriculture in the production of fertilisers to accelerate food growth, and in the production of a multitude of chemical compounds including plastics production and pharmaceuticals. The process typically uses iron as the catalyst [218, 219], and requires both high pressures and temperatures to overcome thermodynamic limitations of the reaction and maximise ammonia production [218, 219]. Other catalysts have been shown to reduce such requirements [220, 221], although these often contain other caveats such as expensive rare materials or complex production processes. However, high energy requirements and detrimental environmental impacts [222] bring about an incentive to both improve the yield and reduce undesirable by-products (such as CO_2) of this critical catalytic process.

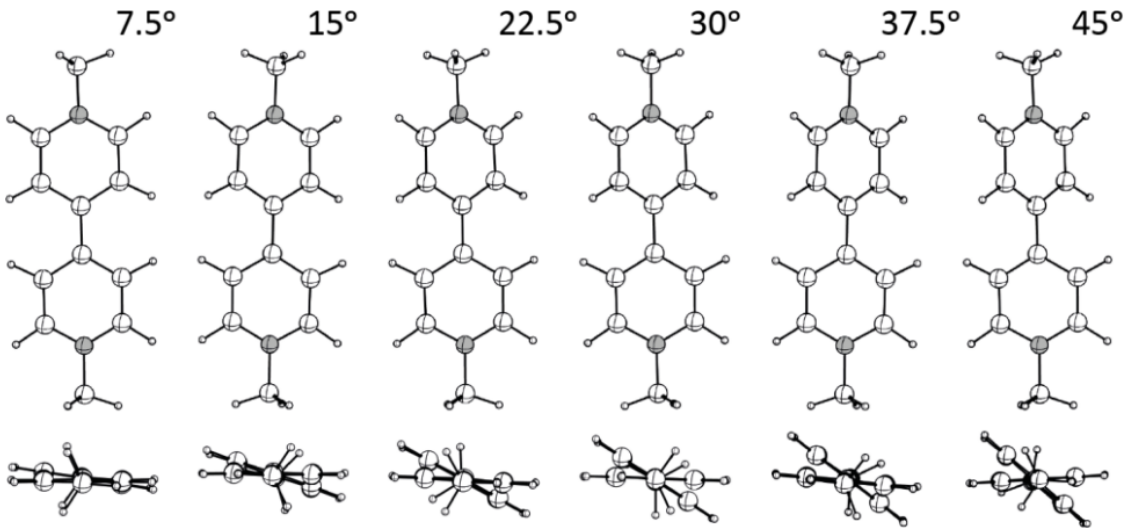
The CO_2 reduction reaction aims to utilise CO_2 as a feedstock (raw material) for conversion into valuable chemical compounds, including renewable fuels like methanol (CH_3OH) and ethanol (CH_3CH_2OH). This process consequently serves to reduce the volume of this greenhouse gas, thereby mitigating its effect on global warming. Heterogeneous catalysis enables the CO_2 reduction reaction to occur by providing active sites on the catalyst surface where CO_2 molecules can be absorbed, dissociate (break chemical bonds), and yield desirable products. In a similar vein to the Haber-Bosch reaction, this process has a high energy consumption that can undermine the environmental benefits. A range of catalyst materials have been investigated including metal nanoparticles [223, 224], metal oxides [225], and carbon-based complexes [226] that aim to reduce such requirements, however these techniques may require rare materials obtained through environmentally unfriendly extraction methods, or the resulting process may possess low selectivity and efficiency of the desired products.

Building upon the work set out in the previous chapter, this research allows changes in molecular bonds to be tracked over time as they interact with the metal atom, and can provide a map of molecular perturbations resulting from such interactions. An example of which is in a highly conjugated molecule, wherein a change in the electron density of one molecular bond can have a strong knock-on effect on

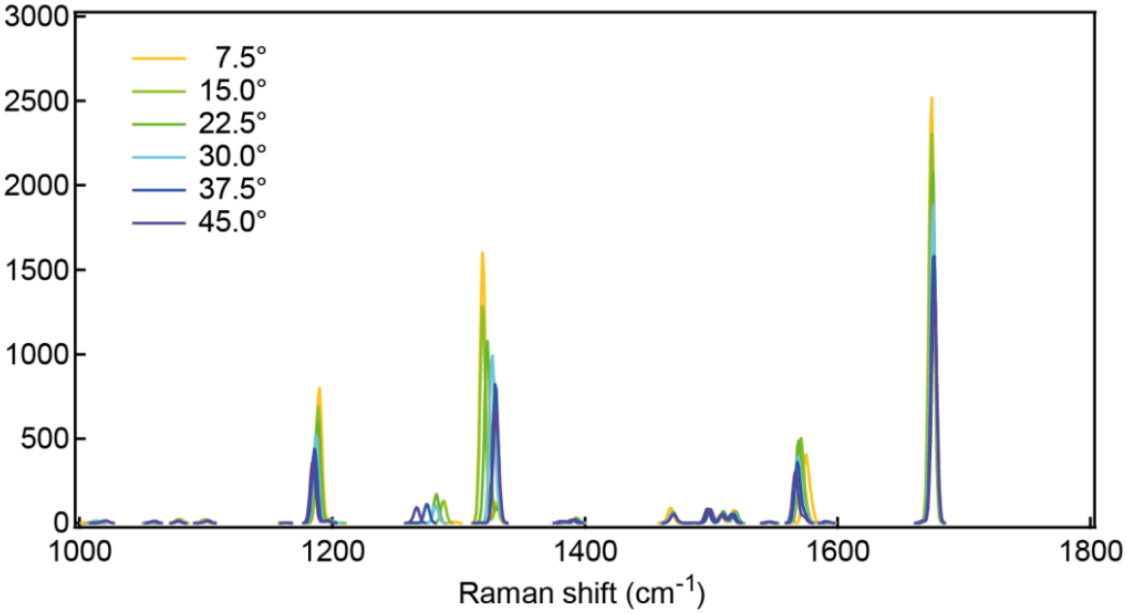
another molecular bond several atoms away. These interactions are important to understand as they provide an insight into how molecular bonds weaken or strengthen as a result of metal-molecule binding, which is a critical step in catalysis.

This is achieved through an analysis of the spatiotemporal information within an Event - which encompasses a set of transient peaks associated with the same picocavity event - revealing correlated changes in the wavenumber positions of these peaks. Such correlated changes are caused by perturbations to the local gradient field due to the source (adatom) drifting about the metal surface upon which it resides. Analysing correlated movements in the described spectral shifts provides an insight into which bonds, when perturbed, most strongly affect other bonds within the system. This can pertain to how the configuration of a molecule changes under torsion, as shown in de Nijs *et al.* [227] (see Figure 4.2), or to bond activations, where specific bonds weaken to the point of dissociation.

By utilising a pre-training and fine-tuning procedure on a machine learning architecture known as a Siamese neural network, which is capable of distinguishing between the aforementioned positive and negative correlated changes, such information can assist in the analysis of these atomic-scale events. Beyond that, it can aid in the identification of specific favourable metal-molecule interactions and promote advancements in the targeted design of heterogeneous catalysts, which aim to tackle the global issues faced today in industrial catalytic processes.



(a) En-face and in-plane depictions of methyl viologen dichloride at various pyridine torsion angles.



(b) Raman activity showing the effects of each pyridine torsion angle. Note that, as the torsion angle increases, the peaks at around $1200\,\mathrm{cm}^{-1}$ and $1600\,\mathrm{cm}^{-1}$ undergo positively correlated changes with respect to each other, contrasted to negative correlations with the two peaks around $1300\,\mathrm{cm}^{-1}$ and $1700\,\mathrm{cm}^{-1}$.

Figure 4.2: The Raman response of methyl viologen dichloride varying with torsion angle about two pyridines groups - organic molecules similar to benzene in which one methine group (CH bond) is replaced with a nitrogen. The calculated spectra demonstrate positive and negative correlated changes in peak positions resulting from changes in torsion angle. Figure reproduced with permission [227].

4.1 Processing Framework for Analysis of Correlated Peaks

To study the behaviour over time of the BPT molecules and gold adatoms, the changes in peak positions are analysed. These shifts in peak positions arise from the displacement of the adatom producing a picocavity on either the surface of the gold nanoparticle or the substrate. In an Event, these spectral shifts are correlated and can be categorised as either positive, indicating simultaneous increases or decreases in wavenumber across two Tracks (defined in Subsection 3.2.3 of the previous chapter), or negative, indicating that one Track increases in wavenumber whilst the other decreases. In Events with more than two Tracks, there could be both positive and negative correlations amongst its constituents (Figure 4.2). By assessing the polarity of correlated peak shifts in wavenumber space, as a result of adatom movements, interactions between vibrational modes can be analysed.

As previously introduced in the opening of this chapter, this section focuses on the design of a framework that enhances the capabilities of the data analysis pipeline established in the previous chapter. Branching off from the Event formation stage of the pipeline, a self-supervised training procedure of a Siamese-CNN was built using a dataset consisting of Tracks formed from the initial BPT dataset. Self-supervision is a machine learning process whereby unlabelled input data is used to generate paired training data, which is labelled, in order for a model to learn basic relationships about the data. This self-supervision utilised a data augmentation task designed to synthesise pairs of correlated Tracks using a sequential set of image processing techniques: Delaunay triangulations to form a tessellation across images containing individual picocavity peaks, and piecewise affine transformations to warp each image.

A Siamese-CNN was selected for this task as the model can predict correlations after training without requiring feature engineering, a step often associated with basic correlation analysis methods. Additionally, it is robust to noise and thus results in a more generalised model. After basic relationships about the data have been learned through pre-training, a smaller dataset of manually labelled peaks belonging to the same picocavity Events was assembled to fine-tune the Siamese-CNN to identify positive and negative correlations in a binary classification task.

4.1.1 Dataset Assembly

To analyse how the detected peaks are correlated, two-dimensional (spatiotemporal) data associated with each Track within the same Event was extracted from the picocavity scans they belong to. As the type of correlation shared between any two arbitrary Tracks is unknown, a data augmentation system was employed to create a labelled dataset with which to pre-train a neural network. The augmented Track pairs are given a binary label classifying the sign of the correlation (1 for positive, and 0 for negative). A two-dimensional Siamese-CNN architecture was selected for this purpose, due to the ability of the model to minimise a distance metric for similar objects (positive correlations), and maximise it for distinct ones (negative correlations), which the model achieves by using shared weights

between each 'arm' of the network. This concept can be intuitively understood with comparison to an example, facial recognition. The face of an employee is captured and compared to a reference, even if the faces appear similar, significant differences in individual pixel values can exist between the images. This makes a basic root mean square error unsuitable for an accurate comparison. Instead, a Siamese-CNN learns to base the comparison on unique features like facial structure and skin tone, but disregards factors such as the positioning of the face, lighting and fashion accessories. The Siamese architecture is described in Subsection 4.1.3 to follow.

As the neural network requires specific input dimensions, a minimum duration for a Track is set from which the dataset is extracted. Tracks that are at least 100 time steps in length are pooled, forming 'sub-Tracks', referred to as images, via the use of a sliding window to capture multiple sections of each Track. This process captures the information contained over the entire length of the Track in a manner that preserves the fixed dimensional requirements of the training images. The sliding window begins at the earliest time step occupied by any Track in an Event and ends at the latest time step in the same manner. The stride of the sliding window is five. For example, this results in two images forming from each Track within an Event that has a duration of 105 time steps (see Figure 4.3). The horizontal position of each sliding window is centred on the mean wavenumber of each image, and captures the complete peak at each time step detected by the Track isolation algorithm - plus five additional pixels each side of the peak to account for cases where the full peak was not captured. The width of the input shape is set to 25 px; images whose widths did not match the width of the input shape were linearly interpolated. This interpolation does not affect the time step axis to prevent non-physical aberrations in the resulting images.

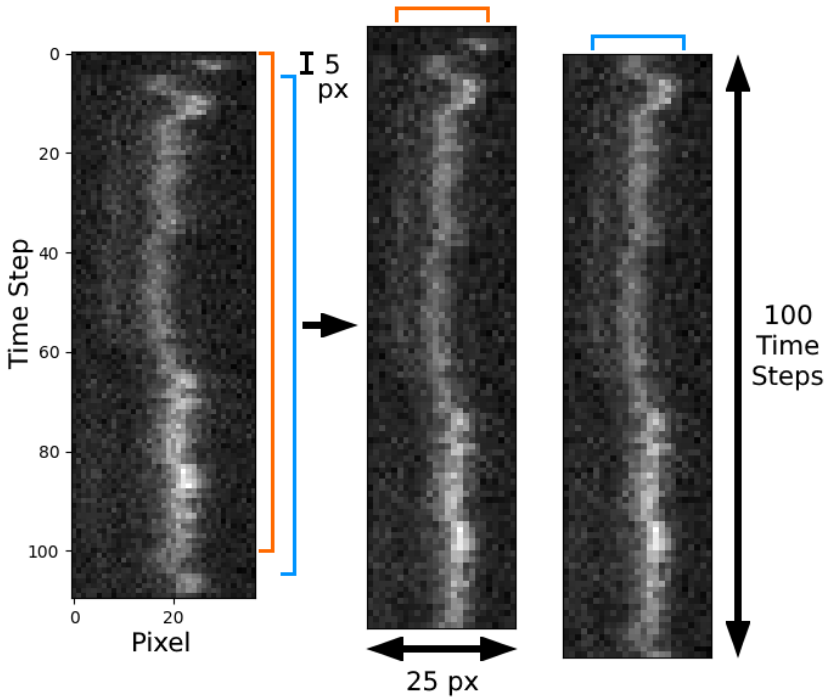


Figure 4.3: Images extracted from one Track. The Track, *left*, is split into two images that have been linearly interpolated to a width of 25 px. The sliding windows, marked with orange and blue brackets, each contain an image of shape 100×25 px at different perspectives of the Track.

4.1.2 Synthesising Correlated Images using Piecewise Affine Transformations

A labelled dataset is produced with the aid of a data augmentation process whereby each training image is warped twice, independently, to create a pair of correlated images. This is achieved through a piecewise affine transformation [228] on sets of Delaunay triangulated data points [229, 230]. The data points that are used to perform the Delaunay triangulation are sampled from a 5^{th}-order polynomial that is fit to the central wavenumbers of each transient peak detected within a Track. If one or more adjacent time steps within a Track do not contain a centroid, either due to an undetected peak or a physical gap within the scan, then centroids are linearly interpolated using known neighbouring centroids to fill those gaps for the polynomial fit. Alternatively, if any time steps contain multiple centroids - potentially due to a peak bifurcation - then the mean of those points are taken as the central peak for that time step.

Once the polynomial has been fitted to the peak centroids contained within the image, data points are sampled along that curve. The data points are located at the peaks, troughs, and rest positions - i.e. the points of maximum gradient between two extrema - of the polynomial. Additionally, the peak positions at the start and end of the image are used as data points, as well as pixels uniformly

spaced along the borders of the image; eight along the vertical edges, and four along the horizontal edges (corners shared).

The set of data points belonging to the peaks, troughs, and rest positions of the polynomial are then randomly shifted in wavenumber position, using a normal distribution, by up-to 15 % of each value from the central line of the image along the wavenumber axis. Figure 4.4 shows an example of the sampled data points, before and after horizontal noise-shifts. There are two sets of data points formed from this process: one set containing the randomly shifted positions, as well as the start, end, and border points; and the other set containing all original positions.

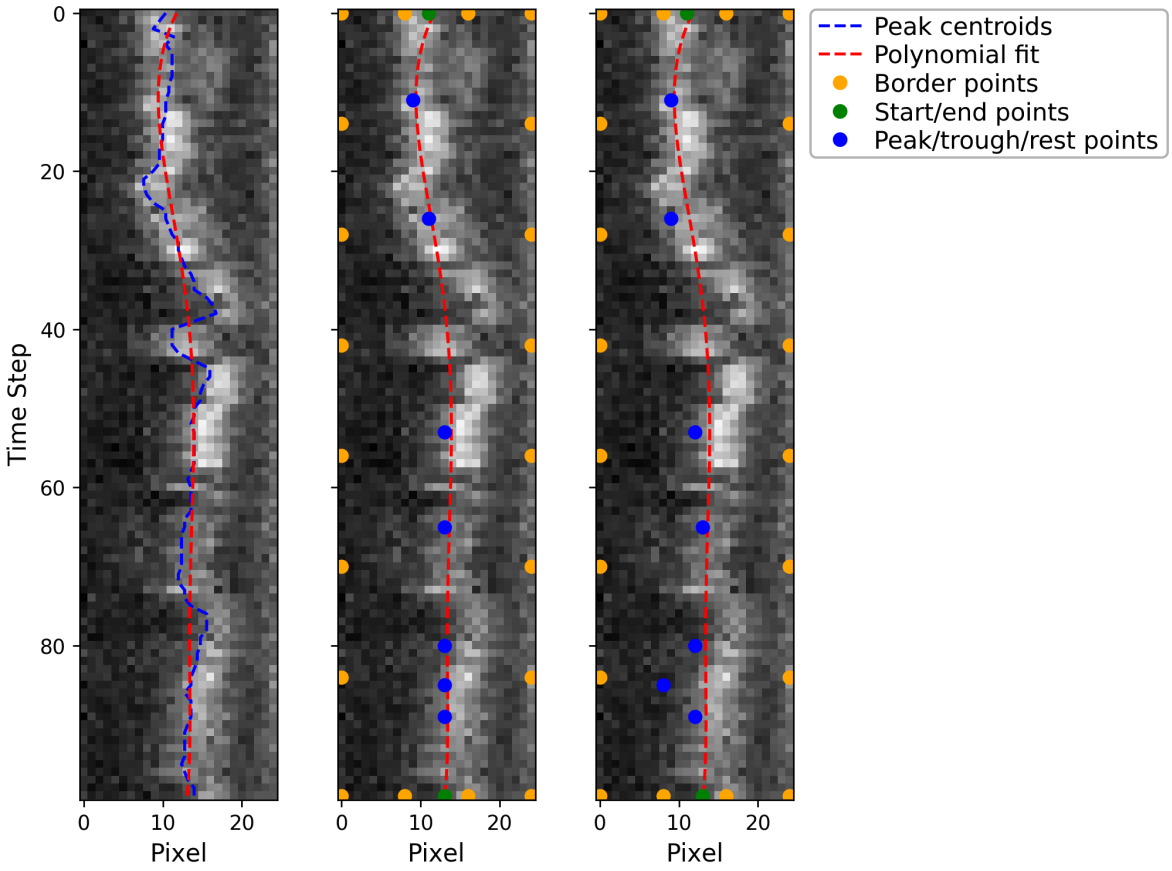


Figure 4.4: Example polynomial data point generation. *Left*, the 5^{th}-order polynomial is fit to the complete set of peak positions, overlaid onto the image; *middle*, the initial data points sampled from the polynomial, as well as the points along the borders; *right*, the noise-shifted data points. These data points form the vertices of the Delaunay tessellation in the next stage of data augmentation.

These two sets are then triangulated using the Delaunay method [228, 229, 230], and the resulting tessellations are used to warp the image from the original (source) tessellation, to the noise-shifted

(destination) tessellation, using a piecewise affine transformation (see Figure 4.5). This process was performed twice on each image, which produces a positively correlated augmented pair by default, as the points used in the tessellation, and the strength of the random horizontal shifts, were set to prevent an unintentional transformation of the synthesised peak correlation to negative. Additionally, to prevent the amount of noise added to *e.g.* a peak from falling below the position of a resting point, a cap was placed on the maximum allowed shift to one pixel above neighbouring data points. In order to produce a negative correlation, each synthesised pair had one member flipped horizontally at a 50 % chance. Lastly, the intensity values of the images were linearly normalised between the values [0, 1] in preparation for being processed by the neural network.

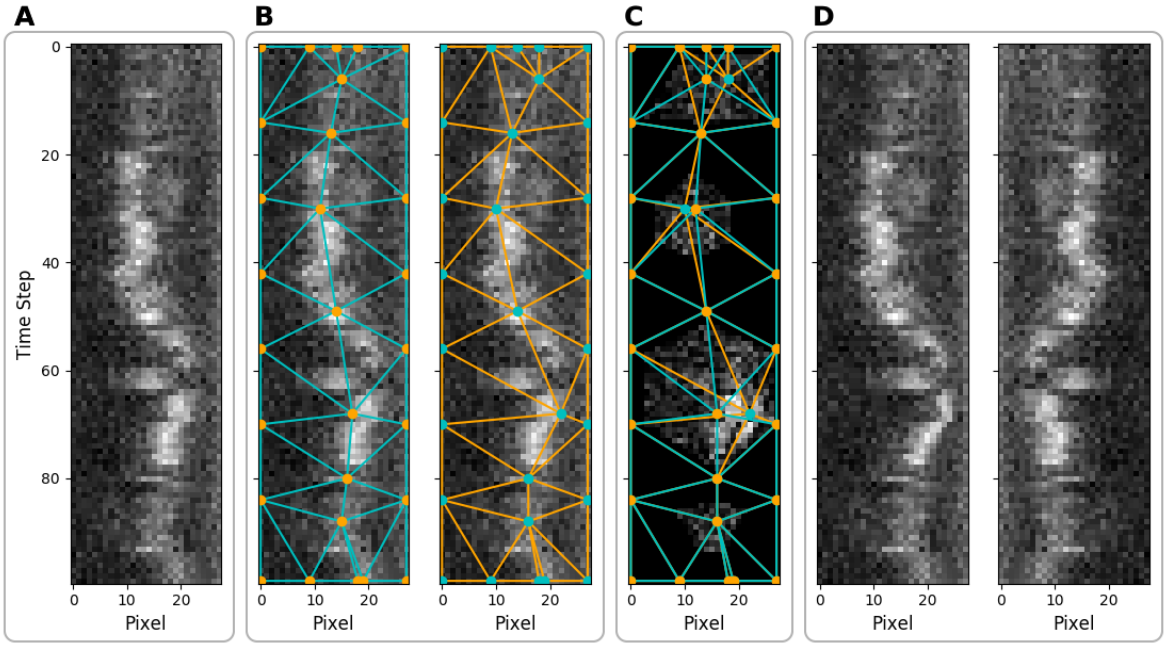
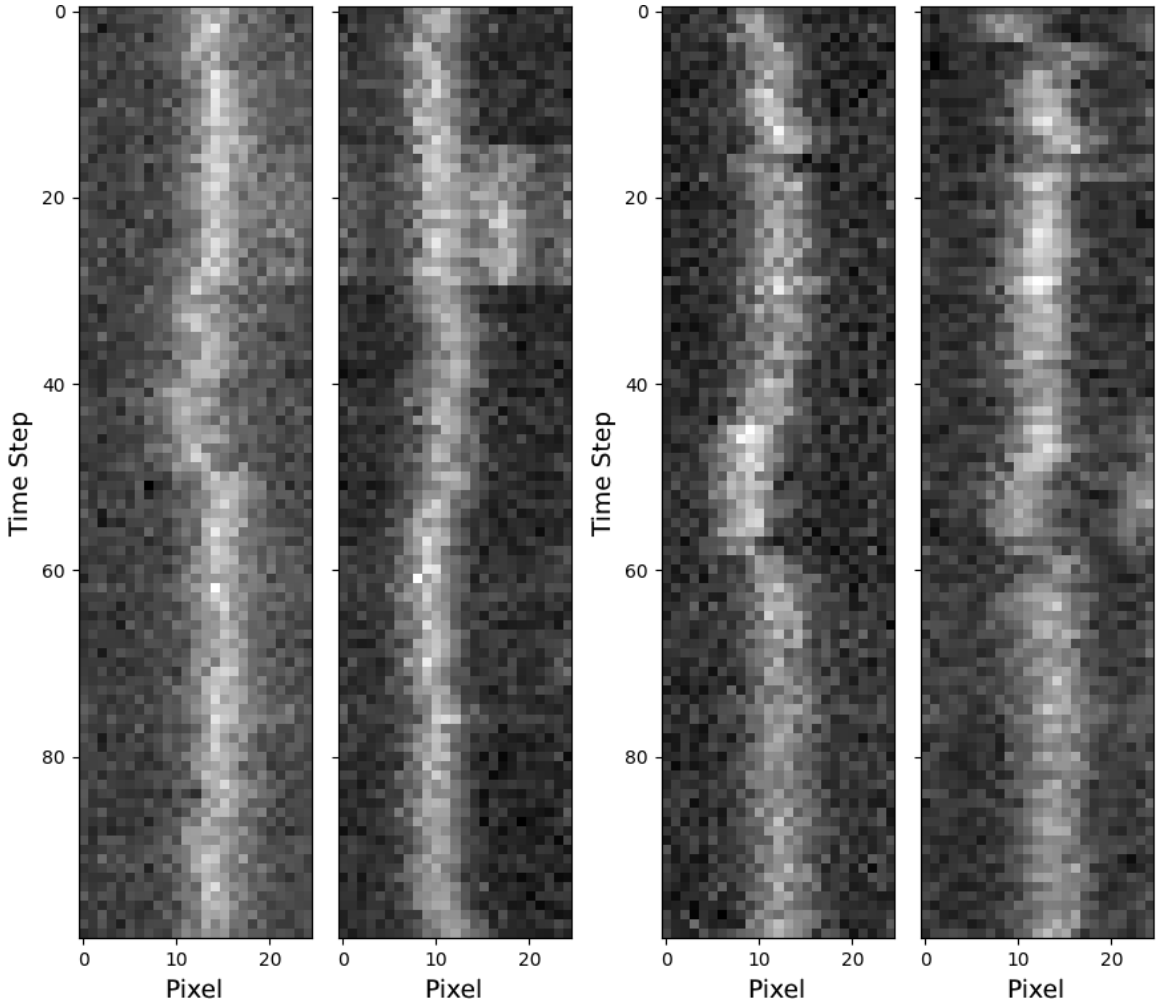


Figure 4.5: A) The original image. B) The two Delaunay triangulated tessellations - *left*, original positions; *right*, noise-shifted positions. C) The difference in pixel intensities between the original image in A and the *left* augmented image in D after the piecewise affine transformation. D) *left*, the augmented image; *right*, the accompanying augmented image in the pair (steps not shown) - this image has been horizontally flipped to create a negative correlation between the image pair.

From following the described data augmentation process on all eligible Tracks, the resulting dataset contained 3850 image pairs. The inference datasets are a fixed set of augmented pairs and labels, whereas the training dataset is used to generate a new set of augmented pairs every epoch during training. After the Siamese-CNN architecture, described in the following subsection, was pre-trained on this dataset, it was fine-tuned using a dataset of real Track pairs that were isolated within the same

Events, which were exclusively determined as having either positive or negative correlations. There were 455 pairs of real Tracks that were manually assigned correlation labels. Examples of positive and negative peak correlations between real Tracks is shown in Figure 4.6.



(a) Negative correlation.

(b) Positive correlation.

Figure 4.6: Examples of real correlated pairs used to fine-tune the Siamese-CNN.

4.1.3 Siamese-CNN Model Architecture

The CNN arms of the Siamese-CNN contain six layers, including the input layer. There are four convolutional layers, followed by a 128-unit FC output layer. The outputs of each CNN arm are combined into one vector, using the absolute difference distance metric between each unit. This

combined vector is processed by the 'decision head', a standard FC layer with a single unit. The output of each convolutional layer was normalised using instance normalisation [169, 170], which was initialised from a standardised random uniform distribution [231], followed by a Leaky ReLU activation function with slope coefficient, α, of 0.3, and maxpooling with a (2×2) stride and kernel size. The model depth and size for each layer was determined through a grid search optimisation, minimising the BCE loss.

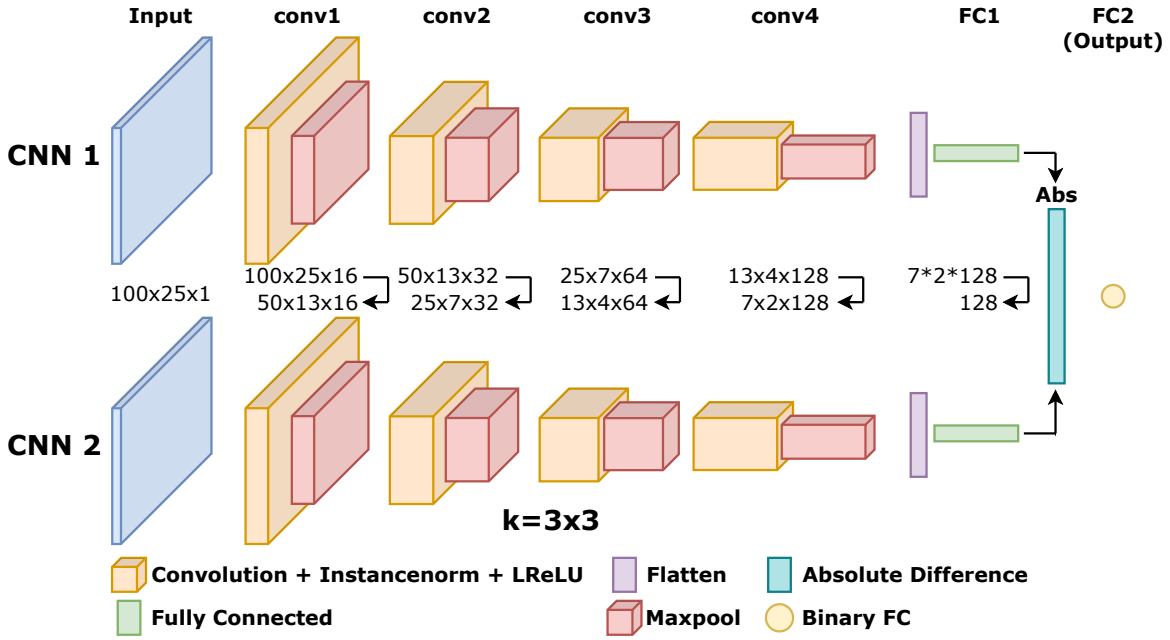


Figure 4.7: Block diagram of the Siamese-CNN. The dimension labels on each layer are in the format (Height, Width, #Filters); the upper values represent the output of each convolutional layer, and the lower values represent the output after maxpooling. Note that the batch size dimension is equal on all layers and is thus omitted. The k-value represents the convolutional kernel shape, which is the same for all layers, and has a unitary stride. The convolutional and maxpooling layers both use zero padding to capture the entire receptive field.

The model was pre-trained for 1000 epochs, with a static learning rate of 0.01, and a batch size of 64. The database underwent a 90:5:5 split, meaning that the validation and testing datasets each contained 190 fixed image pairs. The training dataset produced 3470 augmented image pairs every epoch - one pair per sample. The loss function used was the BCE loss between the predicted and true correlations, and the Adam optimisation algorithm was used - with parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-7}$ - to adjust the model parameters during training. All layers in both the CNN arms and the decision head were regularised using L2 weight decay with a regularisation factor, γ, of 0.1.

Clipnorm [177] was used to clip the calculated gradients to the maximum L2-norm value for each update step, to avoid the problem of exploding gradients.

The model was fine-tuned, with a reduced static learning rate of 0.001, for 13 epochs before early stopping [165] was applied to prevent the validation loss from diverging. The early stopping procedure involved averaging the validation loss curves from the 10 fine-tuning iterations (explained below), observing the mean epoch at which overfitting occurred - the point at which the averaged validation loss curve began to increase - and restoring the parameters of the best model back to the values they had at that epoch.

All other aforementioned hyperparameter values relating to pre-training were also used at the fine-tuning stage. The fine-tuning dataset was partitioned using k-fold cross validation, with a k-value of 10, meaning that each partition held approximately 410 training samples and 45 testing samples. The partitioning was performed based on each whole Track, meaning that, where multiple image pairs would together constitute an entire Track (see Figure 4.3), that set of image pairs would remain within the same partition. This method of partitioning the data was used to avoid creating a testing dataset that was too similar to the training pool, thus avoiding the formation of a trivial evaluation task. For model inference during both pre-training and fine-tuning, a threshold of 0.5 was specified to convert the output of the Siamese-CNN into positive (1) or negative (0) correlations.

4.2 Evaluation of Approach

This section describes and evaluates the decisions made in the development of the Siamese-CNN extension to the data analysis pipeline introduced in Chapter 3. Specifics of the design and implementation of the dataset assembly method used to train the Siamese-CNN are given, alongside a justification for the normalisation technique used to train the model; the method by which the performance of the Siamese-CNN is evaluated is detailed; and a visualisation tool is introduced to display the predicted peak correlations, which can assist in the rational design of heterogeneous catalysts. A more detailed description of the potential of the visualisation tool described in the final subsection is provided in Section 4.3, involving a discussion of potential future work based on an analysis of the temporal information present within the data.

4.2.1 Specifics of Data Processing Stages and Model Design

Regarding the design of the data preprocessing and augmentation steps for analysing correlated peaks, as well as the specifics of the model architecture, the physical aspects of the data were considered. The width of the input image (25 px) was selected in this manner based on the average separation, for all Tracks contained within the assembled dataset, between the left and right extrema wavenumbers of all detected peaks, plus the additional 5 px either side of these values to account for incomplete detections (as mentioned in Subsection 4.1.1). The height of the input image (100 px) was selected empirically,

motivated by an attempt to obtain stable, long-form Events whilst retaining a suitable dataset size resulting from all Events that matched this criterion. Smaller values for the image height would increase the number of samples within the assembled dataset, and allow for shorter duration Events to be incorporated and subsequently analysed, thus this is an avenue worthy of further investigation.

When the images extracted from Tracks are augmented into pairs for use in pre-training the model, the piecewise affine transformation only applies to changes in the horizontal (spatial) axis. This is due to the exclusively-horizontal shifts in the positions of each central node in the Delaunay point set when noise is added (see Subsection 4.1.2). The purpose of this is to minimise any transformations occurring along the vertical (temporal) axis, which would otherwise produce non-physical aberrations in the resulting image pairs.

Motivated by the statistical nature of each image pair being distinct from other image pairs within a batch, the Siamese-CNN was chosen to be normalised using instance normalisation [169, 170], rather than conventional batch normalisation [163]. Instance normalisation normalises each image (or instance) separately from the others within the same batch. As the Siamese-CNN is normalised after each convolutional layer with instance normalisation, the model learns a representation of the data that is robust to the stochasticity inherent to batch normalisation - more generally known as batch whitening, as shown by Huang *et al.* [232]. However, at the output layer of the model, it must produce an accurate prediction from this learned representation, which is difficult to achieve if the output from the final layer were to be normalised - hence the final FC layer in the decision head is left unnormalised.

4.2.2 Model Evaluation using Receiver Operating Characteristic (ROC) Curves

The area under curve (AUC) value of the receiver operating characteristic (ROC) curve was used to evaluate the performance of the Siamese-CNN architecture after pre-training, and the average performance of each partition after fine-tuning. An ROC curve is a graph that evaluates the performance of a classification model at all thresholds between 0 and 1, based on two parameters: the true positive rate (TPR) and the false positive rate (FPR). The TPR (also called sensitivity or recall) is defined as:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{4.1}$$

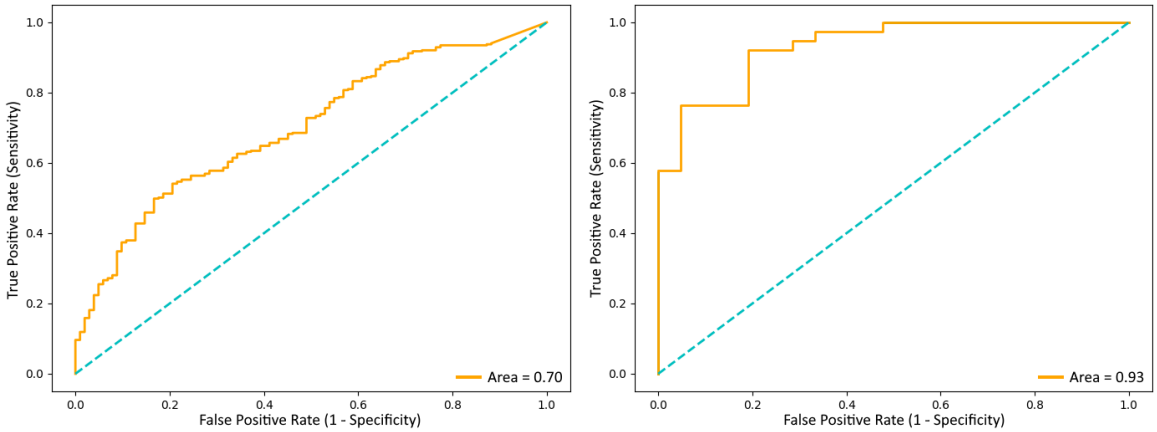
where TP is the number of true positive classifications, and FN is the number of false negative classifications. Similarly, the FPR (also known as '1 - specificity') is defined as:

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}, \tag{4.2}$$

where FP is the number of false positive classifications, and TN is the number of true negative classifications. As the classification threshold decreases, the number of correlated peaks identified by the model as positive increases (see Figure 4.8). The effect of misclassifying a sample as positive will be more or less severe depending on the domain application. For example, if a classifier was trained to

identify a particular disease and determine whether a subsequent treatment is provided to a patient, it may be decided that increased false classifications are permissible (i.e. a higher FPR) in order to maximise the number of treatments (i.e. a higher TPR), hence the classification threshold would be lowered.

In the case of SERS data analysis within the bounds of this research, the proportion of positively and negatively correlated peaks is unknown, hence a classification threshold of 0.5 was assumed to avoid bias. However, amongst all the tested model variations, which involved modifications such as adjusting static or adaptive learning rates and varying the number of epochs, the fine-tuned model was chosen based on the mean AUC value computed across all iterations. The average AUC value was calculated to be 0.8678, with a standard deviation of 0.1264. This method of selecting the most robust model would allow the chosen fine-tuned model to remain a suitable tool should the classification threshold require tuning. Figure 4.8a shows the ROC and associated AUC value for the synthesised testing dataset containing 190 fixed samples after pre-training, and Figure 4.8b shows the performance for one iteration of the fine-tuned Siamese-CNN on the partitioned testing dataset containing 45 real samples.



(a) Testing ROC curve of the Siamese-CNN pre-trained on synthesised correlated peaks.

(b) Testing ROC curve for one iteration of the Siamese-CNN fine-tuned on real correlated peaks.

Figure 4.8: ROC curves and associated AUC values used to evaluate the performance of the Siamese-CNN architecture on the peak correlation classification task.

4.2.3 Peak Correlation Matrix

The model iteration that achieved the best performance out of the 10 trials produced through fine-tuning, based on the highest AUC value of the ROC curve for the testing dataset, was used to perform inference on an extension of the same dataset containing real correlated pairs. There were 1405 total image pairs taken from Tracks belonging to detected Events within the BPT dataset - this includes

the 455 manually labelled Track pairs that were used to fine-tune the Siamese-CNN at each iteration. The classification of these correlations by the network was used to generate a correlation matrix plot, which serves as both a heatmap to identify the most common peak correlations for picocavities within the BPT dataset, and as a tool used to assign transient peaks to those predicted by the DFT. The resulting correlation predictions were plotted at x-y coordinates in the correlation matrix based on the mean wavenumber positions of each image pair - which are the constituents of Track pairs, broken down into 100-time step segments and then resized, as previously described.

Three separate correlation matrices were considered, each containing the same classified pairs, but using a different ruleset for how the images are partitioned. The first plots each image pair individually; the second plots one correlation result for each Track pair, representing the mean correlation of the constituent images (rounded to either a negative or positive correlation); and the third plots sets of image pairs that are grouped with adjacent pairs that share correlation signs, therefore a new set is formed whenever correlation switching occurs. As an example of the third ruleset, if a pair of Tracks contains five sets of image pairs that are initially positively correlated, but switch to negatively correlated after the third image pair - i.e. [1, 1, 1, 0, 0] - then two data points will be plotted on the correlation matrix for this set; one positive correlation representing the first three images, and another negative correlation representing the final two images. This third ruleset was chosen (see Figure 4.9) to account for the classified correlation shifts, which the second ruleset would ignore, whilst simultaneously attempting to reduce a bias in the results incurred by picocavities with longer durations, which would naturally produce more data points should the first ruleset be used.

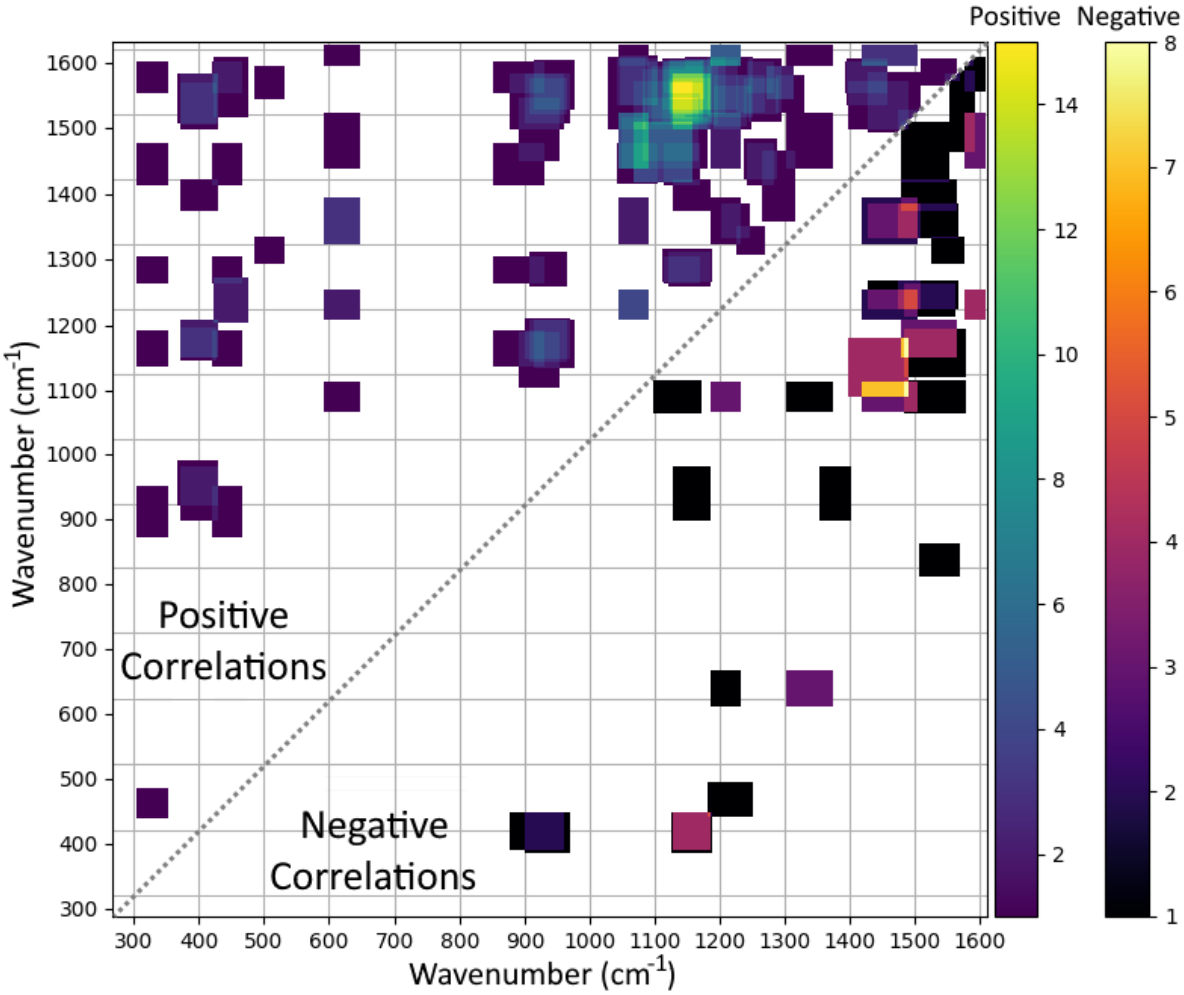


Figure 4.9: Correlation matrix heatmap showing the wavenumber positions of paired Tracks within the same Events, partitioned by shared correlations. The rectangular shape of each data point is based on the wavenumber ranges encapsulated by each member of a Track pair. The matrix can serve as a visualisation tool to assist in identifying regions of Raman activity, such as the density of Raman peaks occurring within the same Events above 900 cm^{-1}, and conversely the lack of Raman activity between approximately 650 to 850 cm^{-1}.

4.3 Discussion and Outlook

The work done in this chapter suggests exciting avenues for future research in the area of rational heterogeneous catalyst design. The aim of this section is to summarise what is currently possible with this research, introduce achievable goals through an extension of the analysis technique, and highlight desirable long-term goals that could significantly impact the field.

Through the use of the Siamese-CNN architecture and the production of a correlation matrix, this research is capable of achieving similar results to that of the previous chapter, through a comparison of DFT simulated Raman spectra and representative picocavity spectra obtained through the data analysis pipeline. Figure 4.10 demonstrates this possibility by overlaying modes predicted by the DFT onto a correlation matrix, featuring both positive and negative correlated peaks generated by the Siamese-CNN in the 1000 to 1600 cm^{-1} range (the region of highest BPT Raman activity).

By making a tentative peak assignment in the same way as described in Chapter 3, gradient field maps could be produced to determine, for example, the position of the adatom that produced this particular picocavity event. Although this chapter focuses on analysing picocavity information produced at the 'Event formation' stage, this technique could be trivially adapted to analysing specific Configurations obtained through clustering. This would allow for a focused analysis of common picocavity events types, creating an avenue for understanding atomic-scale interactions, and potentially influence modifications to catalyst surfaces to either encourage or discourage such events.

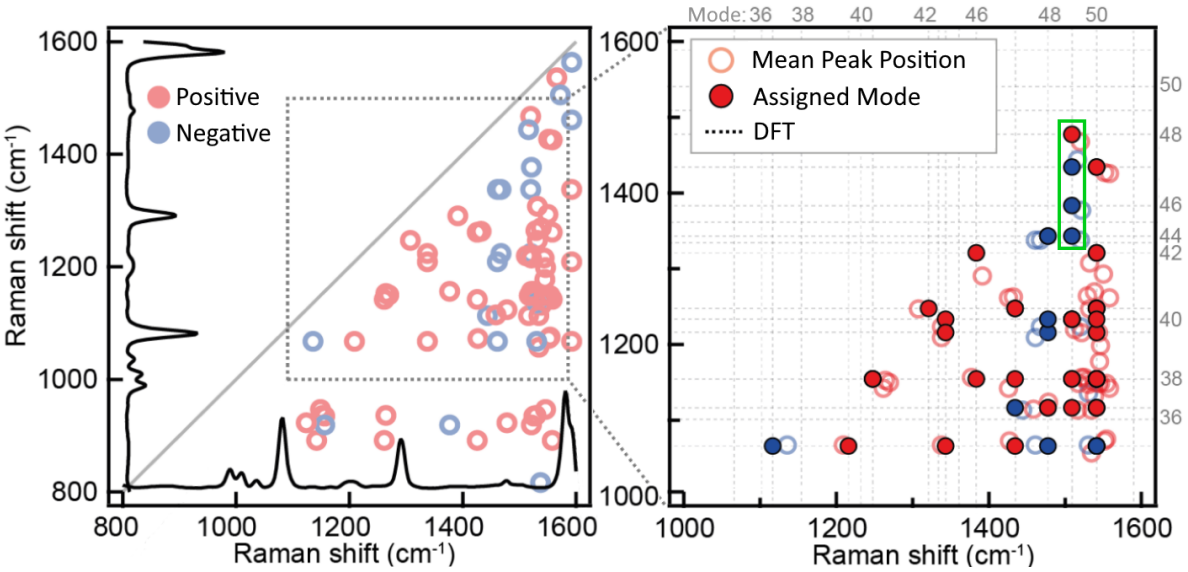


Figure 4.10: Tentative assignment of correlated BPT picocavity peaks to simulated DFT modes. The global resting state spectrum is plotted along each axis in the *left* plot for reference to the nanocavity peaks when making assignments in the magnified plot. The green box in the magnified plot shows a potential region of interest, featuring a single vibrational mode in comparison to three others.

Beyond similarities to the Configuration analysis technique introduced in the previous chapter, this research incorporates the temporal information extracted from time-series picocavity peaks. Fundamentally, this enables the targeted analysis of individual vibrational modes associated with single molecules on a catalyst surface. Through examination of predictions made by the Siamese-CNN and

leveraging the correlation matrix, this approach facilitates a comparison between a specific bond and others across the molecule. Such comparisons allow for determining the impact of one vibrational mode on others, based on the relative strengthening or weakening of the associated bonds - an example of which is shown in Figure 4.10 within the highlighted green box.

Extending this analysis technique may aid in the design of future catalysts to target specific interactions in order to improve yield (increase selectivity) and process efficiency, whilst simultaneously reducing unwanted by-products. This may be achieved through comparisons to DFT to identify which modes, through a relative change in bond strength with other bonds, correspond to a rate-limiting step in a chemical reaction that can be circumvented through targeted catalyst design [218]. Hence, this tool could be used to suggest modifications to a catalyst surface, such as doping one metallic surface in the NPoM geometry, that could weaken the energy barrier preventing a desirable interaction, thus increasing the rate of occurrence for the interaction in question [215, 233].

In addition to characterising the polarity of correlated peaks as explored by this technique, the temporal features within the SERS data offer insights into both the magnitude of each correlated shift and the displacement from the resting position of each peak. Since the strength of a bond perturbation is proportional to its proximity to the source of a local gradient field (refer to Figure 4.11), the relative amplitudes of these peak shifts can assist in finding the spatial location of the atomic feature responsible for generating the observed picocavity. Hence, incorporating this information into the data analysis pipeline is important, as it may aid in identifying the precise location of an adatom, which is crucial knowledge for tailoring catalyst modifications to achieve specific objectives. One such modification may be to adjust the size of the nanogap between two metallic surfaces to promote the dissociation of a particular bond, as evidenced by the vibrational mode above $1500\,\mathrm{cm}^{-1}$ in Figure 4.11 becoming weakest at a distance of approximately 2.25 Å between the BPT molecule and the adatom [73].

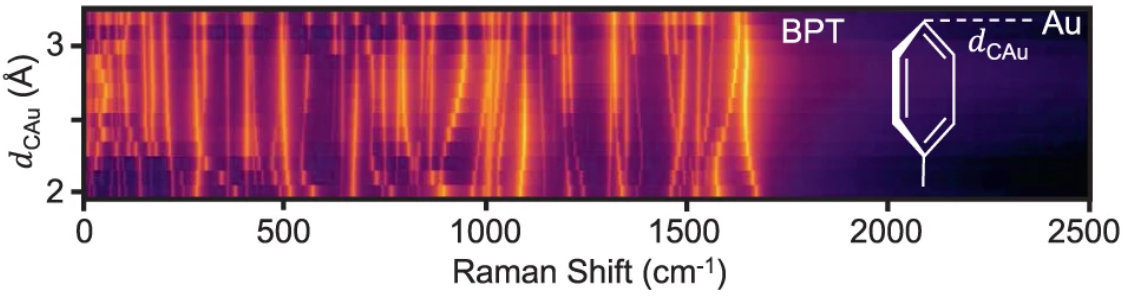


Figure 4.11: DFT Raman spectra of BPT with varying adatom distance demonstrating correlated shifts for each vibrational mode. Figure reproduced with permission [73].

After implementing any proposed modifications to a catalyst, correlation matrices and analysing Tracks in specific picocavity events can serve as valuable comparative tools to assess the impact and, ultimately, the success of these changes. Identifiable features may include: the rate of correlation

switching (e.g. from positive to negative), as this may indicate the stability of a picocavity event; or examining the conditions of Tracks before a desired chemical reaction occurs (i.e. bond dissociation), which could provide insight into further modifications to increase catalytic efficiency. This assessment would involve analysing SERS data taken both before and after modifications have been made. However, the processing and analysis of large datasets would likely be required to facilitate this comparison, in order to reproduce and identify the specific mode interactions under scrutiny.

As mentioned previously, the magnitude of each correlation is important to catalyst design, however, the extension to the data analysis pipeline developed within this chapter uses a Siamese-CNN as a binary classification tool. This means that correlated peaks are only identified as either positive or negative. Thus, whilst the magnitude of a correlation may factor into the prediction made by the Siamese-CNN, it is not output by the model. These detailed features incentivise changes to improve the technique, as two peaks might be weakly correlated, or not at all if the atomic distance between two modes is sufficiently large, which would influence any proposed modifications to a catalyst. Such changes may consist of modifications to the Siamese-CNN architecture to incorporate variable peak correlations, such as a multi-label output with 'pseudo-labels' that identify correlations as strong, weak, or uncorrelated; alternatively, the model could be adapted to a regression task with a single-label output referring to a continuous range of values between -1 to 1, scaling correlations from negative to positive based on the strength of that correlation. These suggested modifications would require a way to quantify (label) peak correlations, or to reconfigure the training routine to an unsupervised or self-supervised task if labelled data is infeasible.

The observed catalytic activity depends on the surface material chosen for the process as well as the physical surface structure [215]. A combination of these two factors could create a catalyst that is highly reactive to a chemical process, but binds too strongly to the desired product, which damages the catalyst as reaction sites would become blocked over time. This brings about an incentive to carefully design catalyst surfaces or sub-surfaces that are tailored to the specific catalytic reaction, maximising reaction efficiency whilst avoiding potential damage to the catalyst as a result of potential described blockages, or other factors including high atmospheric pressures and temperatures. Another important consideration is scaling up the catalyst process to meet industry requirements, as complex catalyst geometries and rare materials may introduce impractical costs. However, such considerations are highly dependent on the particular catalytic process, hence a combination of greater yields and reduced by-products may justify the use of more costly setups. For example, gold NPoM geometries - such as those presented within this chapter and the previous one (Chapter 3) - possess similar metal-molecule interactions to copper oxide in a CO_2 reduction process [224], hence further research into this field may provide results beneficial to increased catalyst efficiency.

To conclude, an extension to the machine learning and image processing data analysis pipeline has been developed to encapsulate the spatiotemporal information present in the BPT SERS data used in this study. A deep Siamese-CNN has been trained as a binary classification model using transfer learn-

ing, based on pre-trained parameters obtained through artificial correlated data synthesised through a data augmentation regime. The model identifies the polarity of correlated peaks based on a dataset extracted from the data analysis pipeline developed in Chapter 3. The aim of this foundational research is to aid in the fundamental understanding of catalysis, by creating a correlation matrix tool to visualise the output of the model, which can be used to analyse changes on single molecules interacting at surface interfaces on a gold catalyst. Further development of this data analysis tool, in particular on the improved differentiation of correlated changes based on the relative strength of spectral shifts, and through analysis of different catalysts to review the effect of proposed modifications, opens up a promising new technique to tailor the design of heterogeneous catalysts.

Chapter 5:

Regression Modelling of High-Concentration Raman Spectroscopy in the Nuclear Industry

Contents

THE content described in the following two chapters refers to work done in real-world industry applications of chemometrics, in collaboration with an industrial sponsor for this PhD, IS-Instruments Ltd., who design and manufacture bespoke remote sensing equipment, with a focus on Raman spectroscopy. Using this equipment, the Raman spectra of mixed substances were measured and analysed using joint machine learning feature extraction and linear regression models trained in this work. The performance of these models were compared with industry standard methods: PCR and PLS regression in optimised scenarios, and the efficacy of these methods are considered as potential complements to, or replacements for, existing analytical tools.

In reality, mixtures are commonplace, whether intentionally in the form of samples of interest suspended in a liquid medium, or unintentionally in the form of interference compounds. These may take the form of residual mineral components leftover by impurities in an arbitrary chemical reaction. Hence, there is a continual high demand for efficient and reliable methods of decomposing mixture spectra, which often form linear combinations of constituent chemical compounds, for the purpose of identifying and analysing target peaks. Within the nuclear sector, obtaining precise predictions of concentrations are essential for the safe, efficient execution of nuclear decommissioning projects, which contributes to cost reductions in necessary safety measures for workers in this area.

The primary goal in most chemometrics applications is to measure the properties of a chemical

system, such as concentration, which is a desirable feature to obtain accurate information for in an industrial environment, with both scientific and financial incentives. In this work, a database of Raman spectra was captured using spontaneous Raman spectroscopy. The dataset contains a known chemical compound (the analyte), dissolved into a known liquid medium at varying concentrations. With the aid of machine learning as a complex, non-linear feature extraction tool, a regression model is developed that can accurately predict unknown concentrations of these solutions, with a performance greater than that of industry standard methods: PCR (involving PCA and a linear regression model) and PLS regression.

Nuclear facilities undergo a process known as post operational clean out (POCO) at the end of an operational life cycle before decommissioning. This is done in order to reduce potential risks and hazards, whilst simultaneously reducing the running costs associated with dismantling the redundant facility. The goal of POCO in this study is to reduce the organic residue found on nuclear sites in vessels and pipework - which may be inaccessible to humans due to unsafe levels of radiation in the surrounding area - to an acceptable level so that they may be repurposed. If acceptable levels of reduction are not achieved, there is a heightened risk of potential fire hazards during the decommissioning process, for example in the plasma cutting of stainless-steel pipework. Therefore, having the ability to precisely identify and quantify concentrations of organic components holds immense importance for the nuclear sector during POCO. This knowledge can significantly enhance cost-efficiency in the safe cleanup of nuclear sites, with potential savings estimated in the range of £10 − £100 million per facility over the course of the lengthy POCO process, which can extend to upwards of 40 years.

Tributyl phosphate (TBP), see Figure 5.1, and odourless kerosene were chosen as a 'worse-case scenario' as by-products in POCO, in which these chemical compounds would exist within a wide range of pipes and vessels, and in a variety of forms such as bulk organics or films on aqueous surfaces, due to the use of these organic materials as 'reprocessing agents' during solvent extraction. Thus the nuclear database is composed of TBP dissolved in kerosene across a range of high concentrations. TBP can be used as an extractant in nuclear chemistry, which serves the role of one of the liquid components in the separation of compounds in a solution between two insoluble liquids. This separation is based on the difference in solubility between the compounds being separated.

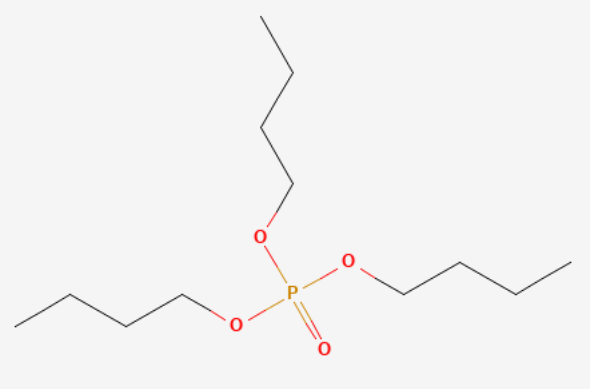


Figure 5.1: Molecular structure of TBP (image taken from PubChem [234]).

Kerosene, produced through the fractional distillation of crude oil, comprises of multiple categories of hydrocarbons, with each category comprising different products, such as: paraffins comprising of different alkanes (single-bond hydrocarbons, e.g. methane), and naphthenes comprising of cyclic aliphatics (compounds containing rings that may be saturated with hydrogen, e.g. cyclohexane). Hence, it is difficult to represent kerosene diagrammatically.

This chapter focuses on the design and implementation of a machine learning regression model for predicting the varying concentrations of a range of liquid sample solutions, which have been chosen based on common usages in nuclear industries, as well as in the biopharmaceutical industry - details of which are covered in the following chapter. As mentioned, industry standard methods are trained and tested alongside the machine learning model, which were optimised per task in order to set a high benchmark for the comparison.

A key theme explored in this chapter is the effect of a small dataset size on the design and implementation of machine learning models. Small datasets will be commonplace in many industrial settings, as companies will require a strong incentive to invest time and money into producing a large dataset for exploratory research. Such considerations include the choice of neural network architecture, the data preprocessing techniques, and the type and usage of data augmentation strategies.

5.1 Data Preparation and Model Design

The machine learning model chosen for this regression task was an FC autoencoder, which was trained to fulfil the role of a non-linear data compression and feature extraction tool for the input mixture spectra. Once the autoencoder was trained, the compressed spectra were fed through a regularised linear regression model (ridge regression) in order to make predictions on the analyte concentration within each mixture. As outlined at the start of this chapter, this work explores the effects of small database sizes on the choice, design and implementation of machine learning architectures, as well

as the data preprocessing and augmentation techniques implemented to attempt to overcome this limitation.

5.1.1 Data Acquisition and Preprocessing

The Raman spectra for the TBP chemical database was captured using a HES2000 spectrometer, an in-house setup from IS-Instruments. The spectrometer features an Andor iVac 316 FT detector, with 150 lines/mm blazed diffraction gratings from Richardson. The 500 mW laser had a central wavelength of 785 nm. The exposure time was set to 30 s for each spectrum. A schematic for the SHS configuration [140, 235] is shown in Figure 5.2.

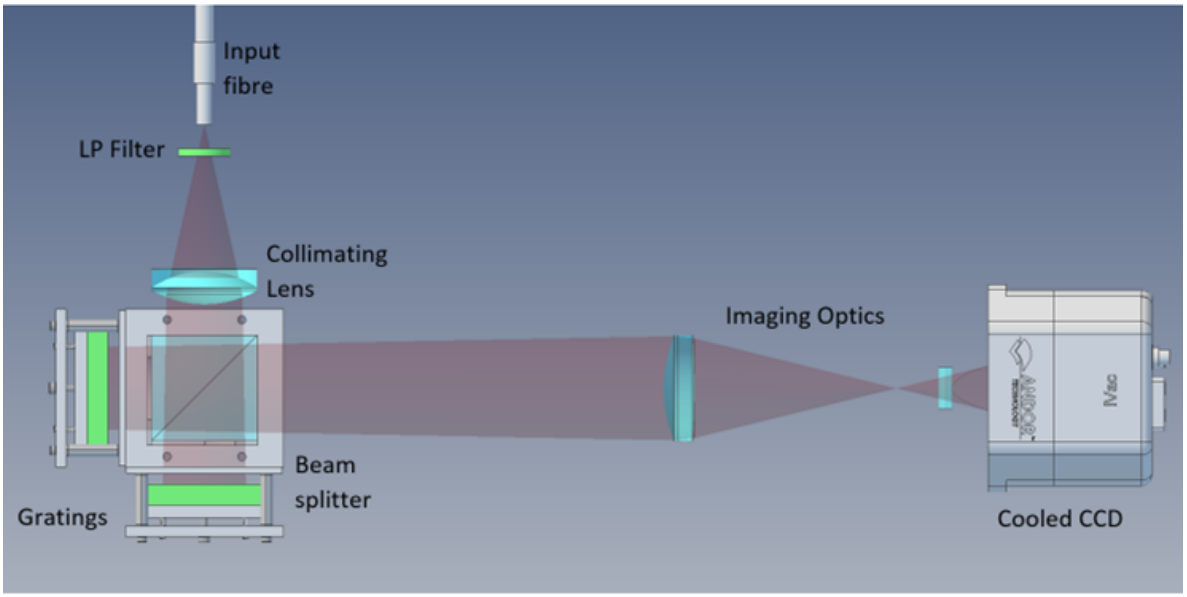


Figure 5.2: Schematic for the SHS used to capture Raman measurements within this chapter and the next. Image courtesy of IS-Instruments [235].

There were 9 concentrations measured for the TBP dataset, with each measurement having 128 repeats, ranging inclusively from 10% to 90% TBP dissolved in kerosene at uniform intervals. The repeat measurements were partitioned into the training, validation and testing datasets at a ratio of 6:2:2, which amounts to 76, 26 and 26 spectra per concentration, respectively. All spectra were interpolated to a wavenumber range as advised by IS-Instruments: 100 to 2200 cm^{-1}, at a constant wavenumber resolution of 4.102 cm^{-1} using a cubic spline interpolation, producing spectra containing 512 bins.

5.1.2 Data Augmentation Strategy

Linearly weighted data augmentation was applied to increase the number of samples, and therefore the variance, in each dataset. Interstitial concentrations could be synthesised using this augmentation method, which were created from neighbouring discrete concentrations sampled from the respective raw datasets. To synthesise an augmented spectrum, three spectra were chosen at random from a data pool combining two neighbouring concentrations, each of which were multiplied by a scaling coefficient, and then linearly combined to produce the spectrum. These three coefficients were sampled from a Dirichlet distribution [236], which satisfies the condition

$$\sum_{i=1}^{N} c_i = 1, \text{ where } c_i \geq 0 \; \forall \; i \in \{1, ..., N\}, \tag{5.1}$$

where values of c_i are the positive coefficients generated in the augmentation process, and N is the number of coefficients. The probability density function, P, of the Dirichlet-distributed vector is proportional to the product of each coefficient, raised to the power of a positive concentration parameter, α. These parameters were assigned the same value of 2.0 - the effect of varying this parameter is demonstrated in Figure 5.3. The probability density function is described by the proportionality

$$P(c) \propto \prod_{i=1}^{N} c_i^{\alpha_i - 1}. \tag{5.2}$$

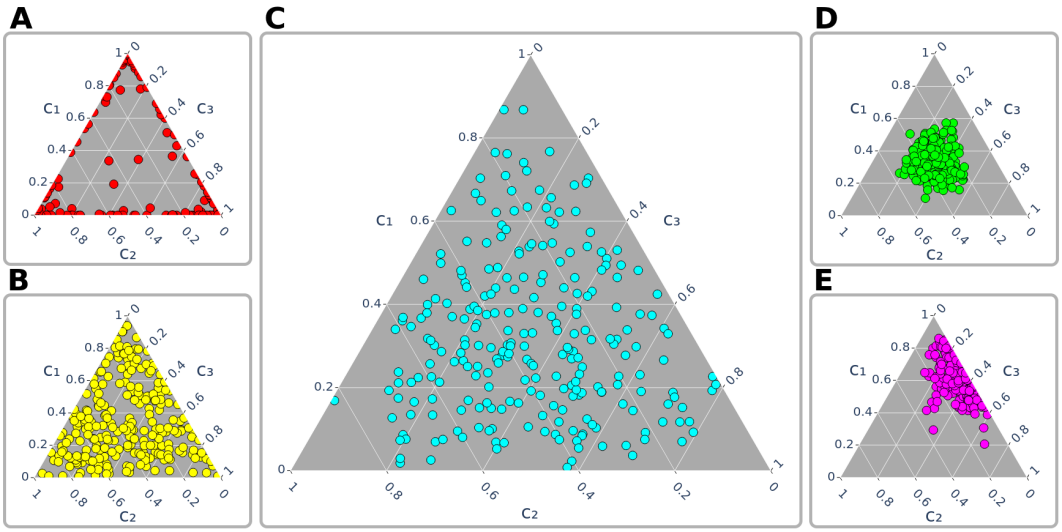


Figure 5.3: Ternary plot demonstrating the changes in the distribution of 250 randomly sampled Dirichlet coefficients, as a function of the positive concentration parameters (α_1, α_2, α_3): (**A**), (0.1, 0.1, 0.1); (**B**), (1.0, 1.0, 1.0); (**C**), (2.0, 2.0, 2.0); (**D**), (10.0, 10.0, 10.0); (**E**), (10.0, 1.0, 5.0). Larger α_i values assign a greater weight to their respective coefficients, c_i. Values for α_i were set to match the distribution in **C**, which produces coefficients that access a broad range, but avoid dominant values such as in **B**. More extreme values are obtained when $\alpha_i < 1$ as in **A**; coefficients of similar values are sampled with larger α_i values, therefore more often equally weighting the contributions of each constituent, as in **D**; and can be skewed to bias towards certain parameters as in **E**.

This data augmentation method allows for an arbitrary number of augmented samples to be synthesised. In total there were 16000 unique training samples synthesised every epoch, with 2000 fixed samples created before training for the validation and testing datasets.

Once the spectra were synthesised, noise was then added to further increase sample variance, followed by normalisation to remove intensity bias from the neural network during training. All pixels from the CCD were added together per column to form the one-dimensional interferogram used to produce each spectrum through an FFT - this process is referred to as full vertical binning. Due to the FT detector used by the SHS Raman system, which uses an FFT to convert the interferogram into an intensity spectrum, normally distributed noise was added to each training sample proportional to 10% of the mean intensity of each spectrum. This distributes the same level of noise amongst all spectral bins, which is more representative of the FFT than scaling the noise proportional to each wavenumber - as is appropriate for dispersive Raman instruments that do not convolve the spectral information. Note that any negative intensities produced by the noise added to the training samples were set to zero, as the instrumental noise that is being mimicked would not produce a negative count. After noise was added, the spectra within each dataset were rescaled using L2-normalisation. The normalisation term is given by the equation

$$T = \frac{1}{\sqrt{\max(\sum_{i=1}^{N} x_{i,s}^2,\ \epsilon)}}, \tag{5.3}$$

where $x_{i,s}$ is the intensity at each bin, i, for an arbitrary spectrum, s, and the value of $\epsilon = 10^{-12}$ is used as a lower bound in the divisor for numerical stability. Each dataset is then scaled by the normalisation term applied to each spectrum using the equation

$$X^d = x_i^d * T^d, \tag{5.4}$$

where X^d is the vectorised form of the normalised spectra for an arbitrary dataset d. Lastly, each dataset is divided through by the global maximum value in the training dataset generated for the first epoch, which linearly broadens the range of values that each dataset occupies (see Table 5.1). Training datasets generated for future epochs the normalisation terms specific to those datasets.

Table 5.1: The extrema values of each TBP dataset partition, before and after broadening their respective ranges by the global maximum value of the L2-normalised training dataset.

TBP Dataset	Narrow		Broadened	
	Min	Max	Min	Max
Training	0.0000	0.0013	0.0000	1.000
Validation	1.246×10^{-6}	0.0012	0.0009	0.9322
Testing	3.622×10^{-6}	0.0012	0.0027	0.9434

Figure 5.4 shows example outputs from the data augmentation process for all concentrations between 10% to 90%, inclusively. As the concentration of TBP increases with respect to kerosene, the main peaks between 500 to 1500 cm^{-1} become more intense, with the exception of the large peak before 1500 cm^{-1} that decreases with increasing TBP concentration - albeit to a non-zero count, as the both TBP and kerosene have a characteristic Raman response around this wavenumber.

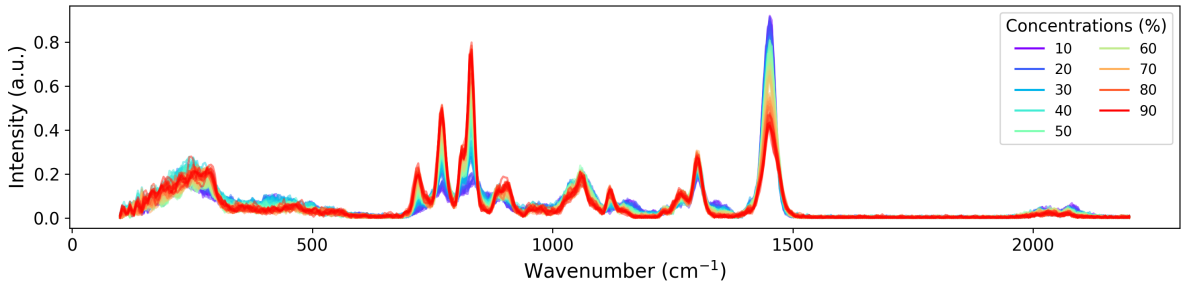


Figure 5.4: Synthesised mixture spectra of TBP through the entire inclusive range of concentrations.

5.1.3 Regression Model Architecture

To predict the concentration of a solution, salient features from each spectrum were extracted using an FC autoencoder trained to reconstruct input spectra. Using the embedding from this trained model, which contains the salient features required for reconstructing the data back to each input spectra, a separate regression model was trained for prediction. Ridge regression, a variation of linear least squares with an additional L2-normalised penalty term (typically used when data suffers from multicollinearity), was used as the regression model for this task. This regression model was fit to the same training data that was used to train the autoencoder. Similarly, the testing dataset partitioned for use by the autoencoder was used to evaluate the success of the ridge regression model, thereby evaluating the combined 'AE-Ridge' regression task.

The autoencoder used in this work contains 5 layers, including the input and output layers. There are two FC layers in the encoder, the last of which being the 128-unit embedding layer, which was used as an input for the decoder. The decoder mirrors the architecture of the encoder. The output of each hidden layer was normalised using batch normalisation, followed by a Leaky ReLU activation function with slope coefficient, α, of 0.3. Dropout was used as the last layer in each hidden block with a dropout rate of 0.25 to regularise the model during training. The model depth and size for each layer was determined through a grid search optimisation, minimising the MSE loss. A block diagram for the AE-Ridge model is shown in Figure 5.5.

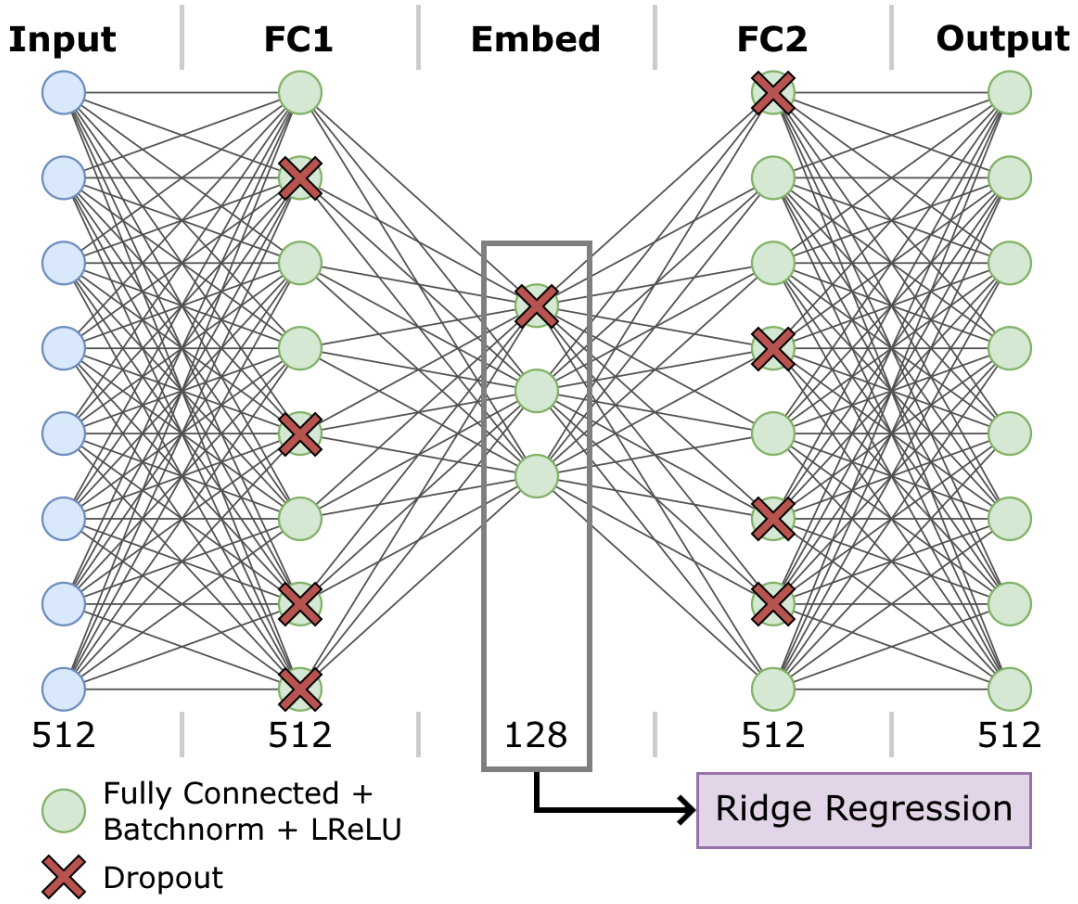


Figure 5.5: Autoencoder block diagram. The effective size of the hidden layers during training is a fraction of each respective size (384) based on the dropout rate of 25%. Note that the batch size dimension is equal on all layers and is thus omitted. The embedding hidden layer of the trained autoencoder is used as an input vector for the ridge regression task to estimate mixture concentrations.

The model was trained for 5000 epochs, with a static learning rate of 0.01, and a batch size of 200 spectra. The loss function used to train the autoencoder was the MSE loss between the input and reconstructed spectra, and the Adam optimisation algorithm was used - with parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-7}$ - to adjust the model parameters during training. All trainable layers were regularised using L2 weight decay with a regularisation factor, γ, of 0.1, and clipnorm [177] was used to clip the calculated gradients to the maximum L2-norm value. Once the autoencoder was trained, concentration estimates were made by training the ridge regression model, which used the embedding vector of the autoencoder as input data, and the synthesised concentrations as output labels, as shown in Figure 5.5.

5.2 Evaluation of Regression Model

Subsection 5.2.1 provides a qualitative discussion for a range of previous neural network architectures trained to carry out this regression task, with a focus on dataset size as a limiting factor for the choice of architecture. A theme explored in this area is model underfitting, which is an aspect of machine learning commonly found in such low-volume data scenarios in which a neural network fails to accurately learn a relationship, if at all, between input and output variables. In addition, the performance of the combined AE-Ridge model is evaluated against industry standard regression tools PCR and PLS regression in Subsection 5.2.2. This comparison is made using performance metrics commonly found in industrial settings: the coefficient of determination (R^2), the 95% prediction interval (PI), and the limit of detection (LoD), of which explanations are provided as to the role of these evaluation metrics alongside the equations that govern them. Similar descriptions are provided for the functionality and operation of PCR (in particular PCA) and PLS regression. Lastly, the complete set of results are provided for all regression tools trained both with and without data augmentation, demonstrating the increase in performance of the combined AE-Ridge regression tool, aided by the data augmentation technique designed for this regression task, over the industry standard used within this study.

5.2.1 Effects of Data Size on Model Selection

The objective of a regression task is to determine the relationship between a number of independent variables and some target dependent variable. From the outset of this study, alternative machine learning architectures were initially considered that attempted to make direct predictions on the target concentrations. Using a combination of FC and convolutional layers, multiple variants of DNNs and CNNs were designed and trained. These models aimed to encompass the complete processing pipeline; the training process would learn any beneficial data preprocessing steps (baseline subtraction, cosmic ray removal, etc.) normally executed in the earlier layers of the model, and progress into learning a relationship between dependent and independent variables through parameter updates that are conducive to making direct, accurate predictions on sample concentrations in the output layer.

There are a number of existing machine learning architectures trained for applications in the processing of the Raman spectra of mixtures. However, the focus of these architectures is primarily on the identification of mixture components [237], or through the use of large datasets used to pre-train neural networks for regression tasks [238] - which is a process in contravention to the small datasets typically available in an industrial setting. A machine learning regression model has also been trained on simulated gamma spectroscopy data for nuclear isotopes [239], which uses a multi-label output to form a probability distribution used to make concentration predictions with multiple analytes.

A collection of neural networks were trained to directly predict analyte concentrations in the TBP dataset, alongside biopharmaceutical data discussed in the following chapter. These varied in features including, but not limited to, the number of hidden layers (2 to 5), the depth of hidden layers (128 to

1024), the inclusion or exclusion of regularisation techniques such as dropout and weight regularisation, batch normalisation, and both ReLU and leaky ReLU activation layers. Neural networks featuring exclusively FC layers were categorised as DNNs, whereas variants that replaced some earlier FC layers with convolutional layers (ranging from 1 to 3 hidden layers) were part of the CNN set.

As previously discussed, spectroscopic datasets obtained in industrial settings are typically small in volume, as is the case with the TBP dataset used within this study (around 1000 spectra), and to an even greater extent in the following chapter focused on the biopharmaceutical sector. As such, the DNNs and CNNs that were trained using either MSE, or mean squared logarithmic error (MSLE), as the loss function failed to learn the relationship between individual Raman spectra and corresponding concentrations. Instead, the geometric mean concentration of each dataset was predicted, suggesting that these architectures were underfit to the training dataset. These results indicated that there was an insufficient volume of data required to train the model architectures in order to make direct predictions, as shown by characteristic features of the training and validation loss curves seen in Figure 5.6. The constant separation between training and validation loss curves, and the large variations seen in the validation losses, are suggestive of an unrepresentative training dataset too small to form a generalised model, despite the data augmentation strategy described in Subsection 5.1.2.

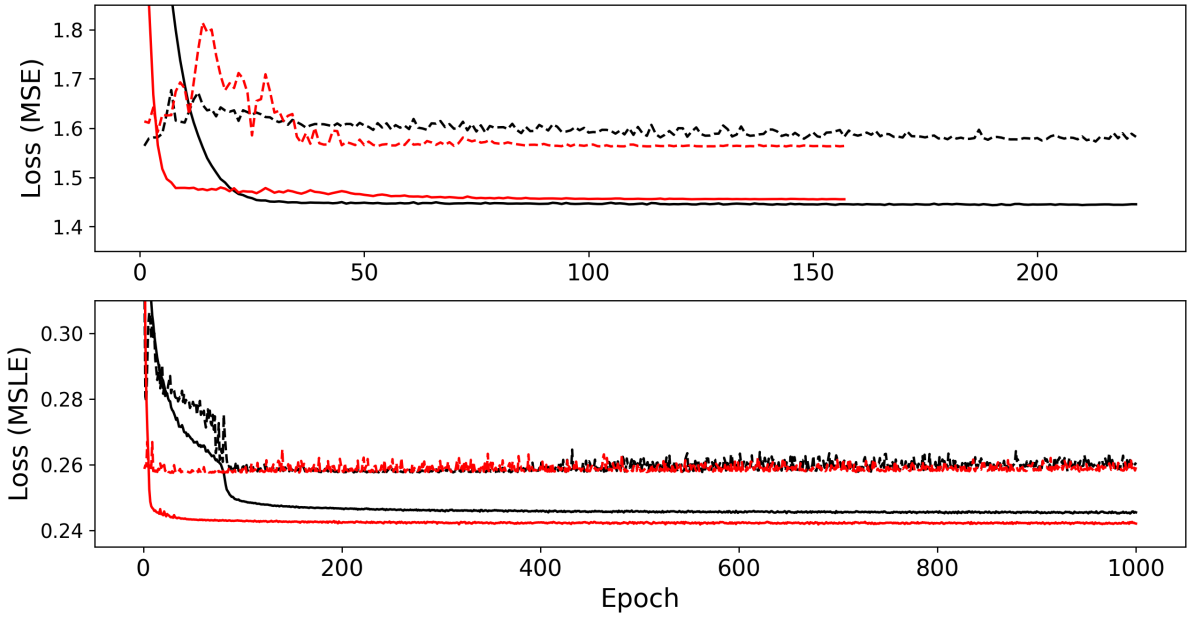


Figure 5.6: Example model performance for the DNN and CNN models trained to make direct predictions on sample concentrations for the TBP dataset. *Top*, two CNN architectures show sharp initial decreases in training losses (solid lines), followed by a negligible change at later epochs suggesting underfitting due to insufficient data required to train the model. *Bottom*, two DNN architectures show similar sharp decreases in loss as training begins followed by the same plateauing effect on the training loss. The validation losses (dashed lines) have notably higher variability in comparison to the training losses, indicating an unrepresentative validation dataset to evaluate the model.

Stemming from the underwhelming performance of the DNN and CNN architectures, and motivated by the success of the CAE trained in Chapter 3, the regression model was divided into two components: a feature extraction task handled by an autoencoder, chosen for the capacity of the architecture to learn salient features conducive to accurate data reconstruction (in this case, spectra); and a linear regression model (specifically ridge regression) to generate the desired concentration predictions once trained on the feature embeddings produced by the autoencoder.

The complexity of the machine learning regression model using this structure was at first reduced down to the most basic form, with a single fully connected hidden layer for the feature embedding and a ReLU activation function, but without any additional hyperparameters such as weight regularisation or dropout. At this stage, the combined autoencoder and ridge regression model proved capable of learning a relationship between the dependent and independent variables in the training dataset, in order to make accurate concentration predictions. Consequently, the complexity of the model was then iteratively increased in the same fashion as described for the DNNs and CNNs to maximise the predictive power of the AE-Ridge model. Thus the final autoencoder architecture, described in

Subsection 5.1.3, provided the best results on the regression task alongside the ridge regression model. The inclusion of convolutional layers was also investigated but was found to be detrimental to the performance of the regression model, hence the exclusive use of FC layers in the final autoencoder.

5.2.2 Results and Comparison to Industry Standard Methods

The performance of the AE-Ridge regression model was compared against two standard regression methods used for concentration prediction: PCR and PLS regression. The latter of which is widely used in chemometrics and other similar areas of spectroscopic data processing [99, 106, 107]. The metrics used to evaluate the performance of each regression model in the concentration estimation task were the R^2, 95% PI, and LoD metrics.

The Coefficient of Determination. The R^2 metric measures the proportion of variance in a dependent variable that is explained by an independent variable, which are the true concentrations of each mixture and the concentrations predicted by the regression task, respectively. It is described by the equations

$$R^2 = 1 - \frac{RSS}{TSS} \tag{5.5}$$

$$RSS = \sum_{i=1}^{n} (y_i - \tilde{y}_i)^2 \tag{5.6}$$

$$TSS = \sum_{i=1}^{n} (y_i - \mu)^2, \tag{5.7}$$

where RSS is the sum of squares of residuals, defined using y_i as the true concentrations and $\tilde{y}_i$ as the predicted concentrations, for all n samples; and TSS is the total sum of squares, using the same variable definitions as in Equation 5.6, in addition to the mean concentration of all samples, μ.

Prediction Interval. The PI metric estimates an interval, based on existing data, within which a given observation will fall at a certain probability. A prediction interval of 95% was selected for this evaluation. The formula is as follows:

$$PI = \hat{y}_i \pm z\sigma, \tag{5.8}$$

where $\hat{y}_i$ is the i^{th} predicted value (concentration), σ is the standard deviation of the predicted concentration distribution. The parameter z defines the standard score of the prediction interval, which is the number of standard deviations by which a raw value succeeds or precedes the mean. A 95% level of confidence was specified, hence z is set to a value of 1.960, which is obtained from a z-score of

0.9750 located on a z-table. The value of 0.9750 is used rather than 0.9500 on a z-table because the 95% PI is a two-tailed test that defines a range inclusive of both lower and upper bounds, as shown in Figure 5.7, therefore the combined percentages equate to the 95% PI value.

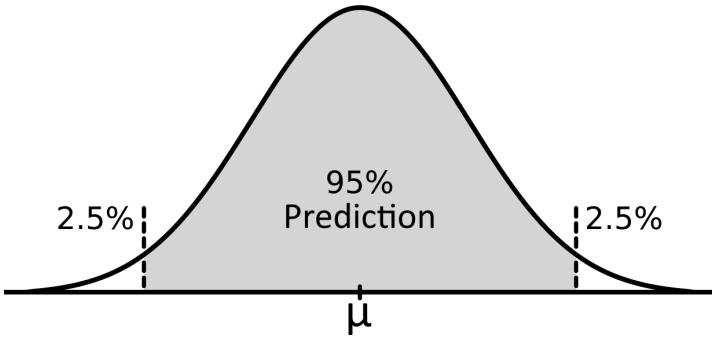


Figure 5.7: Illustration of the 95% Prediction interval.

Limit of Detection. The LoD metric establishes the lowest analyte concentration that can be reliably detected by an analytical technique or instrument. This metric holds significant industrial relevance as it represents a threshold at which target signals can be distinguished from background noise, with a typical confidence level of 99% [240]. A low LoD value enhances quality control in impurity detection, and strengthens safety and risk assessments by increasing confidence in the sufficient absence of harmful substances. These attributes directly relate to POCO, in which the reliable detection of organic residues is critical to the industrial process. LoD relates to the SNR of the instrument and, in this context, reflects the sensitivity of the regression method. It is described by the equation

$$\mathrm{LoD} = 3.3\,\frac{\sigma_{\mathrm{SD}}}{\mathrm{m}}, \tag{5.9}$$

where the 3.3 is a constant corresponding to a 99% confidence level in a normally distributed dataset, and m is the slope of the linear regression line obtained from a calibration curve. The variable σ_{SD} is the standard deviation of the response of the model to the analyte signals. This value can be obtained using the calibration curve, either by calculating the standard deviation of the y-intercepts from multiple regression lines, the standard deviation of blank signals (those that do not contain the analyte), or from the residual standard deviation of a single regression line. As the calibration curve used in this study relates to the linear relationship between predicted and true sample concentrations, the residual standard deviation of the regression line is used to calculate the LoD for each model.

Principal Component Regression. PCR can be separated into two parts: PCA and a regression task - typically a least squares model. PCA is a data reduction technique that aims to reduce the dimensionality of complex data by calculating the 'principal components' of that data, which are sets

of line vectors fit to minimise the squared distance to that line, whilst remaining orthogonal to all previous vectors. Using this strategy, an i-dimensional dataset undergoes a change of basis to the same number of dimensions, but where the i^{th} dimension describes a portion of the total variance in the dataset that is smaller than the previous $i-1$ dimensions. It is common to discard a number of principal components, leaving only the first N which describe a dominant share of the total variance in the dataset; this strategy can therefore remove much of the noise within a dataset, as the more structured, salient features are typically contained within the principal components that are retained. The regression task that follows PCA typically uses a linear regression model, such as ordinary least squares regression, applied to a number of high-variance principal components. However, low-variance components have also been shown to be of a similar, if not greater importance, as regressors [241].

The TBP dataset was standardised as an initial preprocessing step, followed by a dimensionality reduction using PCA, as previously described. As the data is standardised, each principal component represents the eigenvectors of the data correlation matrix, which describes the joint-variability, or causal relationship, between the complete set of data variable pairs. Hence, the eigenvectors represent directions of variance within the dataset in descending order, where each eigenvalue defines the amount of variance of the corresponding eigenvector. Multiplying each eigenvector by the square root of the respective eigenvalue obtains the loadings of that principal component. PCA loadings are the coefficients of the original variables that are used to form each principal component describing the total variance within a dataset, as shown:

$$\text{Total Variance} = \sum_{i=1}^{N} PC_i = \sum_{i=1}^{N}\sum_{j=1}^{M} w_{ij}X_j, \tag{5.10}$$

where PC_i is the i^{th} principal component of the N total principal components describing the complete dataset variance; M is the number of original variables; and w_{ij} is the coefficient representing the loading of the j^{th} original variable, X_j, for the i^{th} principal component. A combined vector, W_i, containing all loadings from Equation 5.10 represents the eigenvector of the i^{th} principal component. PCA loadings are therefore useful because they allow for the interpretation of each principal component; a loadings matrix can be computed that shows the correlation between each principal component and the original variables, which provides information on which of the original variables describes the largest portions of variance in the dataset (i.e. which of the original variables are most relevant).

Following dimensionality reduction through PCA, a ridge regression model was trained to make predictions on the concentrations of each sample. Ridge regression was chosen for this task to match the AE-Ridge regression model for a more comparable test. Although it is typical to determine an appropriate number of principal components to retain based on a user-specified threshold, such as the first N principal components to contain at least 95% of the explained variance, the AE-Ridge regression model was compared to the best-case scenario from the PCR regression task to set a high benchmark. A set of PCR models was trained based on the full range of retained principal components, meaning

from 1 to the maximum number of original components (512), and the model that achieved the lowest 95% PI score was selected as the best model for the TBP dataset to be compared to the AE-Ridge model.

For completeness of the evaluation, in order to determine whether the increase in performance in the AE-Ridge model was due to the non-linear features extracted by the neural network, or through the data augmentation strategy, the PCR method was tested on the TBP dataset both with and without data augmentation applied. This was done in addition to selecting the optimal number of principal components to retain for each regression model. The results of this test are shown in Table 5.2.

Partial Least Squares. The PLS regression method is similar to PCR in that it is a model trained to make predictions based on dimensionally-reduced data. However, where PCA acts to maximise the variance of orthogonal principal components based solely on the independent variables, PLS regression calculates a linear regression model by finding a relationship in the independent variables that maximises the variance in the dependent variables. Because of this difference the components, hereby referred to as latent components, obtained through an iterative decomposition of sample data are not the same as principal components. PLS regression describes the sample data, X, and the target variables, Y, using the equations

$$\mathrm{X} = \mathrm{TP}^{\mathrm{T}} + \mathrm{E} \tag{5.11}$$

$$\mathrm{Y} = \mathrm{UQ}^{\mathrm{T}} + \mathrm{F}, \tag{5.12}$$

where X is an n $\times$ m matrix of n samples and m independent variables; and Y is an n $\times$ p matrix of n samples and p dependent variables - this represents concentration, hence in this case it is equal to 1. Both T and U are n $\times$ l matrices of n samples and l latent components, with T representing the X scores, and U representing the Y scores - these scores are the values of observations that have been projected into the transformed coordinate system. P and Q are m $\times$ l and p $\times$ l matrices of loadings, respectively. Lastly, E and F are residual error terms. By using these descriptions of X and Y, PLS regression iteratively maximises the covariance, or joint-variability, between T and U in order to make accurate predictions. The optimisation algorithm loops a number of times equal to the number of latent components specified.

As with the PCR test performed, separate PLS regression models were trained on the TBP dataset, both with and without data augmentation, and with a number of latent components for each model that were dictated by the lowest 95% PI value achieved in the range of 1 to the maximum number of components (512) - in order to produce a best-case scenario to compare with the AE-Ridge regression model.

Regression Results. To thoroughly evaluate the regression methods, the complete set of permutations (regression methods and data processing procedures) were tested. To achieve this, the AE-Ridge model was retrained and re-evaluated on all discrete mixtures datasets (*i.e.* without data augmentations). The results of these tests are shown in Table 5.2.

Table 5.2: Results of evaluating the three regression methods trained on the TBP dataset, both with (augmented) and without (discrete) data augmentation. The value N represents the number of components each dataset had after dimensionality reduction. The best results are shown in bold.

Method	Discrete				Augmented			
	N	R^2	95% PI (%)	LoD (%)	N	R^2	95% PI (%)	LoD (%)
PCR	321	0.9909	±4.92	1.59	284	0.9964	±3.11	0.80
PLS	8	0.9918	±4.67	1.55	17	0.9963	±3.13	0.79
AE-Ridge	128	**0.9924**	**±4.51**	**1.48**	128	**0.9984**	**±2.04**	**0.75**

The results of the concentration predictions on the TBP dataset using the three regression methods showcases the increase in performance of the AE-Ridge machine learning regression model over the industry standard alternatives when combined with data augmentation. Without the use of data augmentation, the AE-Ridge regression model still outperformed the alternative approaches in that category, although the results are closely comparable. It should be noted that all regression models benefitted from the use of data augmentation, however the AE-Ridge model saw the greatest gain in performance. In particular, the value of the 95% PI metric for the AE-Ridge model trained with data augmentation achieved approximately 50% better performance over both PCR and PLS regression, which achieved similar results.

5.3 Conclusions

The machine learning regression model developed in this chapter surpasses industry standard tools PCR and PLS regression with a 50% improvement in the 95% PI metric, achieved through a data augmentation strategy that increases sample variance during training. A linear relationship was assumed between neighbouring sample concentrations, which was effective given the linear Raman response of the TBP and odourless kerosene mixture dataset used in this research for high concentration ranges. All regression model permutations consistently achieved R^2 performance above 0.99 for discrete data and above 0.995 for synthesised samples. Each model similarly demonstrated strong LoD performance, with the AE-Ridge model improving upon the other methods by approximately 5%. As discussed in Subsection 5.2.2, this improvement would translate to tangible benefits in the context of risk assessments, by ultimately leading to cost reductions during POCO. Due to the high SNR of each spectrum, uniformly distributed concentration measurements, and linear Raman response, all models could accurately predict TBP concentrations. However notably, the AE-Ridge model outperforms both PCR

and PLS regression in estimate precision when trained with data augmentation.

The regression model was trained on a dataset of organic compounds present in the nuclear decommissioning process, POCO. Nevertheless, there exists other significant molecules that warrant inclusion, such as dibutoxydiethylether (an organic solvent) and breakdown by-products of TBP: dibutyl and monobutyl phosphate. Furthermore, it should be noted that the dataset employed in this study comprises liquid samples, although it is common to encounter residual organics in the form of bulk substances or vapours. Consequently, there is a need for future research aimed at adapting to a multi-class regression task, and to test the accuracy of model predictions across different phases of matter.

Amongst the possible avenues of future research, those worthy of investigation are primarily ones that take into consideration practical hurdles and limitations of real-world applications, particularly in relation to data collection. As a consequence, modifications to data processing stages and the neural network architecture would be at the forefront of future work. Such changes include: implementing either multi-class regression to predict quantities of multiple mixed sample spectra; transfer learning to leverage learned relationships between input variables and output concentrations to adapt to new samples (should retraining a multi-class model be impractical); or replacing the Ridge regression model with a neural network-based regression tool to further improve the predictive performance.

Expanding upon the last point, a neural network-based regression tool could be achieved by retaining only the encoder half of the FC autoencoder, freezing its parameters (i.e. preventing them from being updated through gradient descent), and connecting a simple multilayer perceptron (MLP), which is a neural network exclusively containing fully connected layers, or a CNN with a single output node representing the sample concentration. This proposed method may hold a greater chance of overcoming the underfitting issue, mentioned in Subsection 5.2.1, due to the features learned in the embedding of the autoencoder. In addition, model underfitting may be ameliorated through expanding the diversity of the training dataset to contain multiple classes, each with their own varying concentrations.

The linear Raman response in the nuclear dataset used in this chapter can be further taken advantage of with the inclusion of additional analyte datasets, as the data augmentation technique would be capable of synthesising samples from a mixture of multiple nuclear analytes using a linearly weighted combination of single analyte datasets. This process could be achieved with minimal modifications to the data processing pipeline, and would prove a desirable aspect from an industry standpoint, as the need for measuring a full suite of concentration permutations would be circumvented.

The regression methodology established in this chapter will be extended into the following chapter focused on the biopharmaceutical industry, demonstrating the versatility of this approach.

Chapter 6:

Transferring Success: Low-Concentration UVRRS in the Biopharmaceutical Industry

Contents

The work done in this chapter extends the evaluation of the joint machine learning feature extraction and linear regression tool to real-world chemometrics applications in the biopharmaceutical industry. Drug manufacturing represents a costly undertaking, with estimates placing the expense of establishing manufacturing facilities in the growing market at over £150 − £400 million, creating a strong financial incentive to improve manufacturing efficiency for drugs essential across numerous application areas. Such enhancements result in a decrease in waste products and an increase in drug quality due to the heightened assurance of component concentrations.

Protein biologics currently hold a position of focus in biopharmaceutical research. This field encompasses enzymes, monoclonal antibodies (mAbs), which are specialised drug therapies designed to target specific proteins in cancer cells, and other small proteins. These are essential components in the research and development of new medicines, particularly in key health domains like anti-cancer and immunomodulation [242]. However, the manufacturing of these drugs is a difficult and costly endeavor. Consequently, there is a need to decrease these expenses by introducing biosimilars - near-identical products to an original medicine - which would result in reductions to costs for patients receiving these therapeutic drugs.

The production of mAbs is achieved through a downstream processing tool, liquid chromatography, in which raw biological materials such as cells or tissue are used to produce pure proteins for future use in drug manufacturing. A major challenge present throughout this downstream process is a risk of protein aggregation [243, 244, 245], which affects the yield of the resulting protein products. Besides that,

protein aggregates are connected to adverse immunogenicity [245] - the ability for a drug introduced into the body to produce an undesirable immune response - thus further emphasising the need for quality control during the manufacturing of mAbs. By designing monitoring tools to quantify aggregation levels, adjustments could be made to liquid chromatography processes to assist in maximising drug yield and quality.

Raman spectroscopy has been recognised as a potential monitoring tool. Nevertheless, conventional spontaneous Raman spectroscopy in the visible and near-IR wavelength regions proves impractical for gathering high-quality data, primarily due to significant levels of fluorescence that convolve with the Raman response of analyte proteins. In this work a UVRRS system [246, 247, 248], a form of RRS utilising a deep ultraviolet Raman probe laser, is used to overcome the challenge of fluorescence by operating at low wavelengths (below 250 nm). In this wavelength range the analyte Raman and fluorescence responses become spectrally separated (see Figure 6.1). In addition to this, the SNR of the resulting spectrum is enhanced by two complementary factors. Firstly, as the Raman scattering intensity is proportional to λ^{-4}, the shorter wavelength laser used in the UVRRS system provides an increase to peak intensity. Secondly, as the laser wavelength lowers, the electronic transition of many organic molecules with conjugated structures are approached [248], which produces a resonant effect that amplifies the Raman response by several orders of magnitude [41, 246].

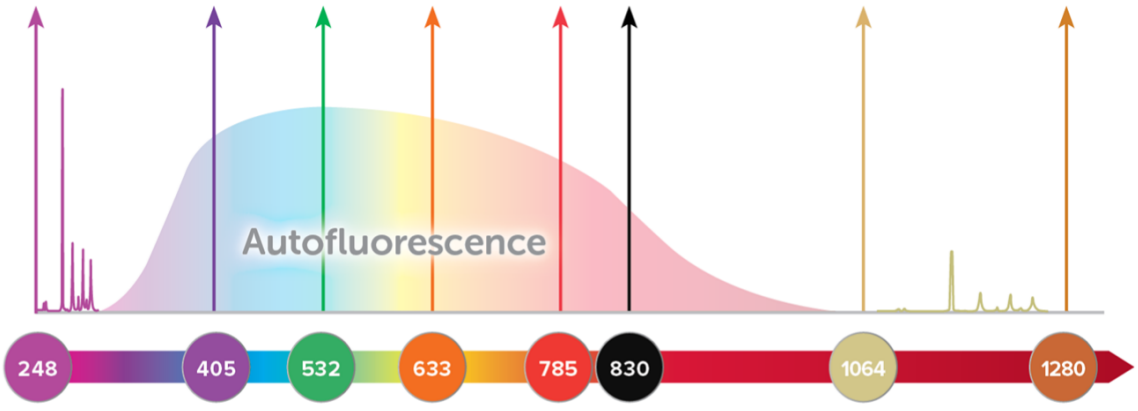


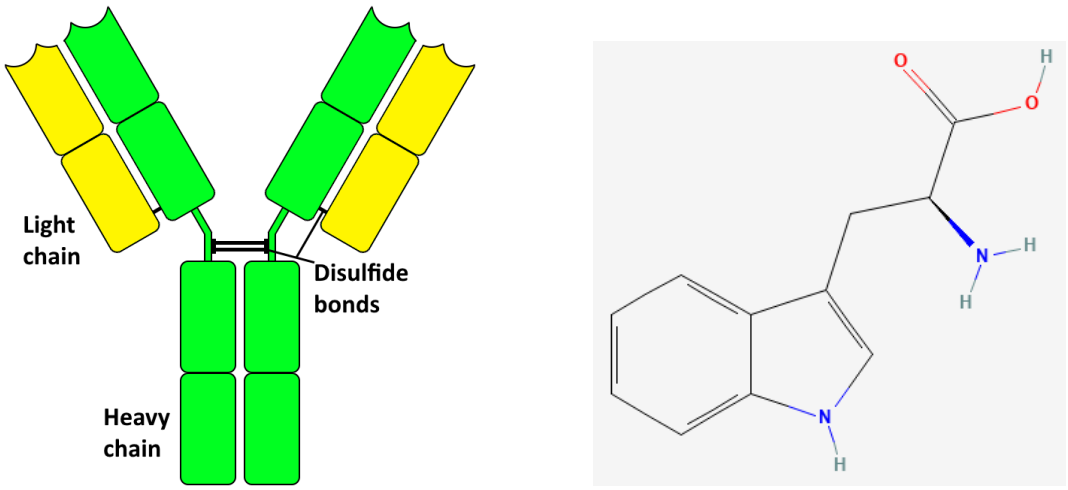
Figure 6.1: Generalised illustration of how fluorescence affects a sample based on the laser wavelength employed. When utilising a low-wavelength laser, as in the UVRRS system (228.5 nm), a spectral separation occurs between the Raman and fluorescence responses. This separation arises from the respective dependence, or lack thereof, on the laser wavelength. The term 'autofluorescence' denotes the fluorescence response inherent to an organic molecule. Cyclohexane is used in this example, which is a type of saturated hydrocarbon used to calibrate the UVRRS spectrometer due to its well-known characteristic Raman peaks.

This chapter investigates two datasets of biopharmaceutical data at low concentrations (mg mL^{-1}),

which is made possible by the described aspects of the UVRRS system that are advantageous to protein biologics analysis. As in Chapter 5, a theme explored in this chapter is the effect of small dataset sizes, common to industry applications, on the design of the regression model. An alternative data augmentation strategy is considered, from which a qualitative explanation is provided for the non-linear Raman response of protein macromolecules due to their size and the resonant Raman effect, which are aspects crucial to the design of data processing stages. Lastly, the limits of the selected data augmentation strategy are investigated by analysing the performance of the regression model based on selected modifications to the augmentation process. This test is carried out on a database of amino acid measurements with a non-linear Raman response, taken at non-uniform concentration intervals.

6.1 Data Preparation and Modifications to Processing Stages

Two databases were created from bio-molecular compounds that had been dissolved in aqueous (water) solutions, as a result of the liquid chromatography manufacturing process. The first being a macro-molecule, immunoglobulin G (IgG), which is a common type of antibody found in humans (see Figure 6.2a). The other molecule chosen was tryptophan, which is an amino acid used in protein synthesis (see Figure 6.2b). As both of these molecules serve important roles in the human body, they are thus relevant areas of study for biopharmaceutical industries in protein biologics research. It is important to accurately predict the quantities of these substances, as the goal is to increase the purity of these solutions due to the role of these organic compounds as common constituents in modern drug design.



(a) IgG (image adapted from Janeway *et al.* [249]). (b) Tryptophan (image taken from PubChem [250]).

Figure 6.2: Molecular structures of IgG and Tryptophan. A simplified schematic representation of IgG is given showing light and heavy chains, which determine the functionality of the antibody.

Descriptions of the data preprocessing and augmentation stages are largely the same as described in Section 5.1 of the previous chapter. Therefore, the subsequent subsections will highlight notable distinctions in the preparation of protein datasets for training and evaluating the performance of the AE-Ridge regression model within the context of the biopharmaceutical sector featured in this chapter. Wherein emphasis will be placed on the acquisition of spectra from low concentration mixtures using deep UVRRS. The design of the FC autoencoder architecture and hyperparameter choices, used as a non-linear feature extraction tool within the machine learning regression task, are identical to the previous chapter, demonstrating (as per the results in Subsection 6.2.3) the ability for the combined AE-Ridge regression tool to handle different types of Raman spectroscopy data (spontaneous and resonance), at both high and low concentrations, across different industrial areas. The training method employed in this chapter mirrors that of Chapter 5. In addition, two distinct versions of the AE-Ridge model were trained, one for each biopharmaceutical dataset.

6.1.1 Data Acquisition and Preprocessing

The Raman spectra for the IgG and tryptophan mixture databases were captured using a deep UV spectrometer called Odin, which is an in-house setup developed by research collaborators at IS-Instruments. The spectrometer featured an Andor iDus 420 FT detector, with 400 lines/mm blazed diffraction gratings from Richardson. A low-pass filter was used to exclude some noise by attenuating high frequency signals from the fibre core. A 9 mW laser was used at a central wavelength of 228.5 nm. The exposure time was set to 30 s for each spectra. The arrangement of this SHS spectrometer is the same as the Odin spectrometer, as seen in Figure 5.2 in the previous chapter.

There were 11 concentrations measured for IgG ranging from 0.1021 to 2.0173 $\mathrm{mg\,mL^{-1}}$, and 17 concentrations measured for tryptophan ranging from 0.0127 to 5.0971 $\mathrm{mg\,mL^{-1}}$. Table 6.1 contains the full list of all concentrations measured for both datasets. For both the IgG and tryptophan datasets, each concentration had 10 repeat measurements that were distributed into respective training, validation and testing datasets at ratios of 6:2:2. Consequently, each dataset contained a total of 110 spectra for IgG and 170 for tryptophan. Therefore, due to the limited size of the datasets available, the application of data augmentation becomes crucial in order to introduce enough sample variability required for training a deep neural network effectively in the context of this regression task.

Table 6.1: Full range of measured concentrations for both IgG and tryptophan in aqueous solutions.

IgG Concentrations ($\mathrm{mg\,mL^{-1}}$)					
0.1021	0.2028	0.4031	0.6064	0.8117	1.016
1.2195	1.4072	1.6156	1.792	2.0173	
Tryptophan Concentrations ($\mathrm{mg\,mL^{-1}}$)					
0.0127	0.0238	0.0409	0.0639	0.0887	0.1027
0.1962	0.3986	0.5922	0.8405	1.0150	1.2200
1.4409	1.6179	1.8190	2.0548	5.0971	

The datasets were interpolated to a fixed wavenumber range as advised by IS-Instruments, which was tuned to encapsulate the full range of characteristic Raman features whilst maximising the respective resolutions by discarding redundant wavenumbers. This resulted in the IgG dataset having a wavenumber range of 600 to 2250 $\mathrm{cm^{-1}}$ at a constant wavenumber resolution of 3.223 $\mathrm{cm^{-1}}$, and the tryptophan dataset having a wavenumber range of 570 to 2270 $\mathrm{cm^{-1}}$ at a constant wavenumber resolution of 2.969 $\mathrm{cm^{-1}}$. Both datasets were interpolated using a cubic spline interpolation, which produced spectra containing 512 bins.

6.1.2 Influence of Non-Uniform Concentrations on Dataset Design

As stated at the beginning of this section, the data augmentation process applied to the IgG and tryptophan datasets were the same as those applied to the TBP dataset described in Subsection 5.1.2 of the previous chapter. However, due to the assumption of linear scaling between neighbouring samples with small differences in concentration, two tryptophan datasets were defined: one retaining the 5 $\mathrm{mg\,mL^{-1}}$ sample, termed Trypt-5; and the other discarding it, termed Trypt-2. This resulted from the closest neighbour to the 5 $\mathrm{mg\,mL^{-1}}$ sample being sufficiently separated in concentration as to reduce model performance should the former sample be retained. Figures 6.3 and 6.4 show examples of synthesised spectra produced by the data augmentation process for the IgG and Trypt-5 datasets. The performances of these models are evaluated in Subsection 6.2.3, followed by an exploratory test in Subsection 6.2.4 to overcome performance issues in the AE-Ridge model trained on the Trypt-5 dataset - owing to the heightened non-uniformity of measured concentrations - through modifications to the data augmentation strategy.

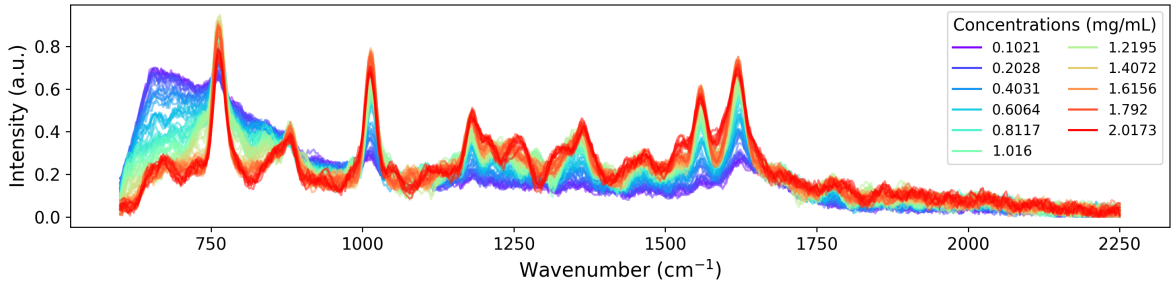


Figure 6.3: Synthesised mixture spectra of IgG through the entire range of concentrations between approximately 0.1 mg mL^{-1} to 2 mg mL^{-1}. The lower concentrations feature a large, broad water peak before 750 cm^{-1} that is suppressed as the concentration increases. Conversely, the intensity of the main IgG peaks increase as the concentration increases.

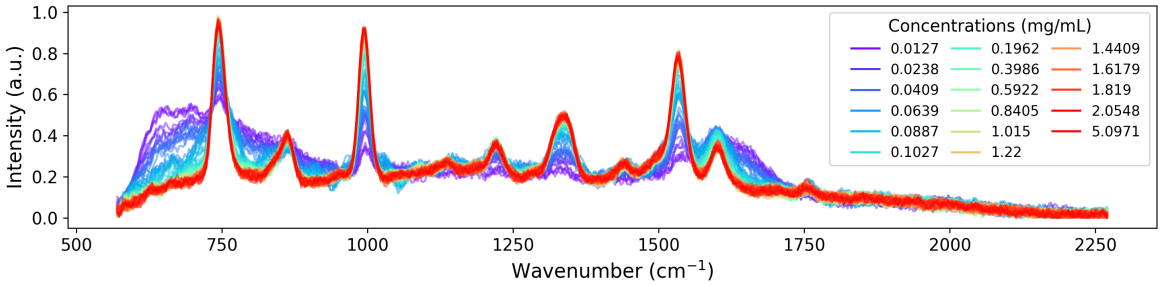


Figure 6.4: Synthesised mixture spectra of tryptophan through the range of concentrations in the 'Trypt-5' dataset, between approximately 0.01 mg mL^{-1} to 5 mg mL^{-1}. As with Figure 6.3, the lower concentrations feature a broad water peak before 750 cm^{-1}, with a similar number of counts, which is suppressed as the concentration increases.

6.2 Evaluation of Regression Model

In this section, attention is drawn to the non-linear Raman responses of both the IgG and tryptophan datasets as the concentrations of each sample vary, due to a combination of factors including: large Raman scattering cross-sections, the large size of the macromolecules and a resulting attenuation effect, and the resonance Raman response of the proteins caused by the UVRRS system. An explanation is provided for the method by which synthesised spectra are normalised to account for this non-linear effect in Subsection 6.2.1. In Subsection 6.2.2, an alternative data augmentation strategy is investigated in an attempt to synthesise more accurate interstitial concentrations that are sampled independently of the measured sample distribution. However, as this alternative method yields monotonically increasing Raman peak heights as sample concentrations increase, contrary to the expected non-linear pattern between Raman peak heights and sample concentrations described by the LIDAR (light detection and ranging) equation [251, 252], which governs signal intensity based on factors such as transmitted laser

power, optical system properties, and analyte characteristics, it was concluded that this alternative approach was not viable.

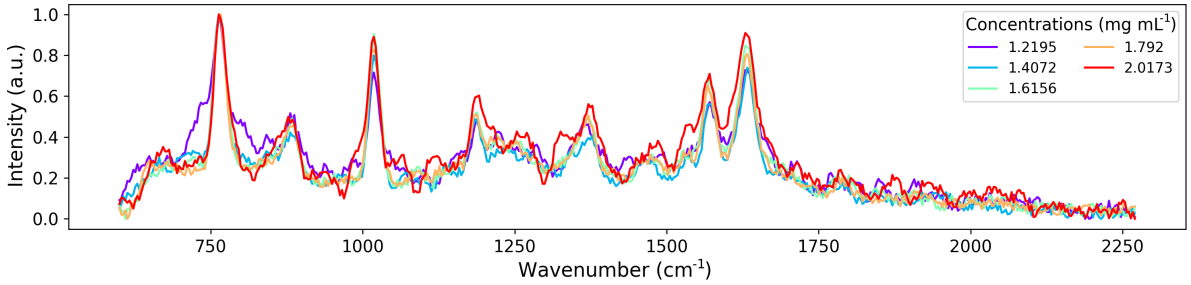
As in the previous chapter, Subsection 6.2.3 provides a comparison between the model performances of the AE-Ridge regression model and the industry standard regression tools, PCR and PLS regression. Lastly, Subsection 6.2.4 investigates a modification made to the data augmentation strategy to account for the non-uniform nature of sampled concentrations in the tryptophan dataset, specifically by taking the natural logarithm of the discrete (unaugmented) concentrations before synthesising interstitial spectra, showing a drastic increase in the performance of regression models trained on both the Trypt-2 and Trypt-5 datasets.

6.2.1 Considerations for Dataset Normalisation

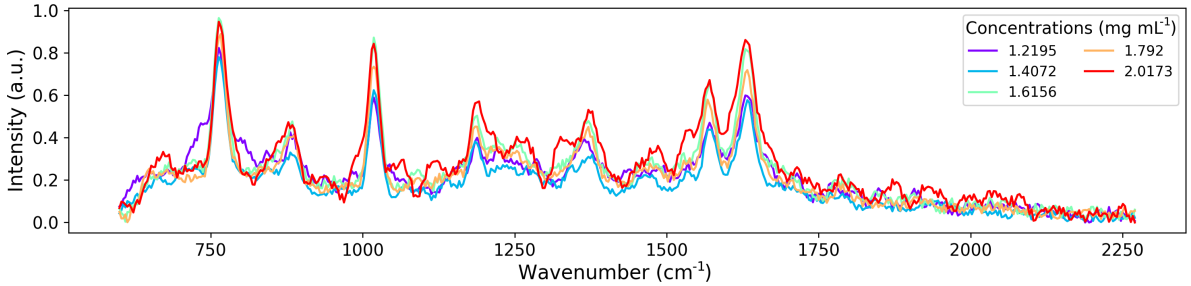
As mentioned in Subsection 5.1.2 of the previous chapter, the datasets for each molecular compound were normalised using L2-normalisation. This choice of normalisation was based on the non-linear Raman response of the IgG and tryptophan datasets evaluated in this work. This section compares the chosen L2-normalisation procedure against a linear normalisation based on a comparison of the resulting spectra, focusing in particular on the Raman peaks of both analytes.

Beyond the discrete sample concentrations of $1.2195\,\mathrm{cm}^{-1}$ for IgG (see Figures 6.3 and 6.5), and $0.3986\,\mathrm{cm}^{-1}$ for tryptophan (see Figures 6.4 and 6.6), contributions to the Raman spectra from water peaks become insignificant in comparison to each analyte. Thus, the only noticeable changes to spectra at higher concentrations are the relative intensities of each analyte peak - assuming all other conditions, such as acquisition time, remain constant. However, at high enough concentrations the peak ratios no longer vary, at which point the distinguishing factor is the ratio between peak height and baseline. This is due to the Raman scattering cross-sections of both IgG and tryptophan being significantly greater than that of water. An analysis of the non-linear nature of the Raman responses of these organic molecules, and how they affect the data augmentation procedure, is given in Subsection 6.2.2.

Because of the change in behaviour of relative peak heights and ratios at higher concentrations, spectra were normalised using L2-normalisation after new concentrations were synthesised (with noise added to training samples), rather than a linear normalisation between the range [0, 1], to attempt to prevent high concentration samples from becoming degenerate. Linear normalisation was also avoided as higher concentration signals would have an amplified level of noise; although the noise present in each signal is not a determining feature for the concentration of an unknown sample, there is a possibility that a neural network could learn a representation of the data that incorporates this as a deciding feature. All spectra were then rescaled, as described in 5.1.2, to simplify the process of hyperparameter optimisation between the multitude of neural networks trained throughout this thesis.

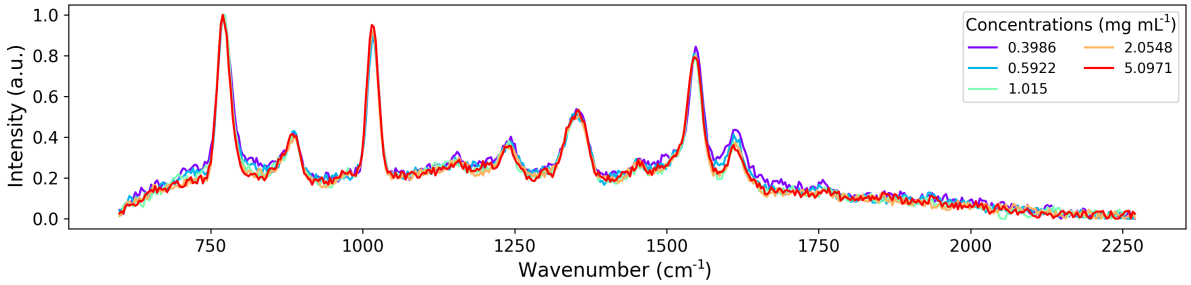


(a) IgG spectra linearly normalised between the range [0, 1].

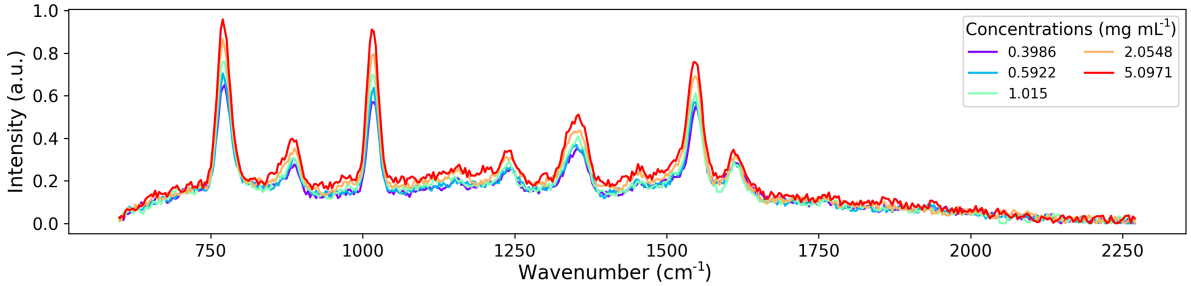


(b) IgG spectra L2-normalised, then broadened between the range [0, 1].

Figure 6.5: Five IgG spectra synthesised at concentrations contained within the discrete dataset. Both normalisation methods mostly maintain distinctions between concentrations. However, the peak at $750\,\mathrm{cm}^{-1}$ is degenerate in **a**, whereas **b** produces an overall better separation of peak intensities with respect to concentrations.



(a) Tryptophan spectra linearly normalised between the range [0, 1].



(b) Tryptophan spectra L2-normalised, then broadened between the range [0, 1].

Figure 6.6: Five tryptophan spectra synthesised at concentrations contained within the discrete dataset. Peaks heights in **a** closely match, making those concentrations more degenerate, whereas in **b** they remain distinct, with higher concentration mixtures having taller peaks.

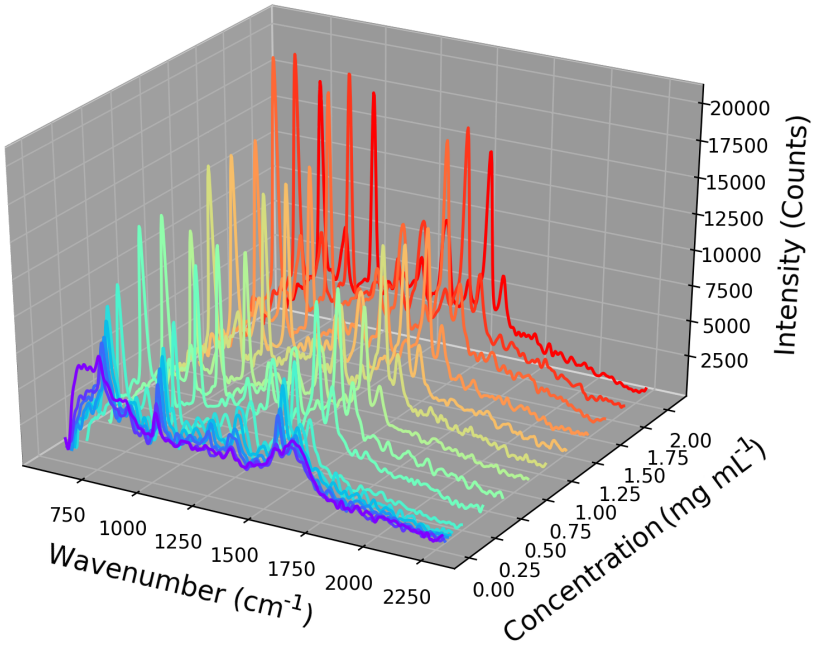
6.2.2 Effects of Non-Linear Raman Response on Data Augmentation

Whilst the performance of the regression model trained on the three datasets (IgG, Trypt-2 and Trypt-5) benefitted from the use of data augmentation, there are shortcomings to the linear data augmentation method - namely that uniformly-spaced concentrations are required in order to accurately synthesise a uniform distribution of concentrations, and the inaccurate assumption of a linear relationship at small differences in concentration between peak intensity and sample concentration. From this, an alternative data augmentation method was investigated.

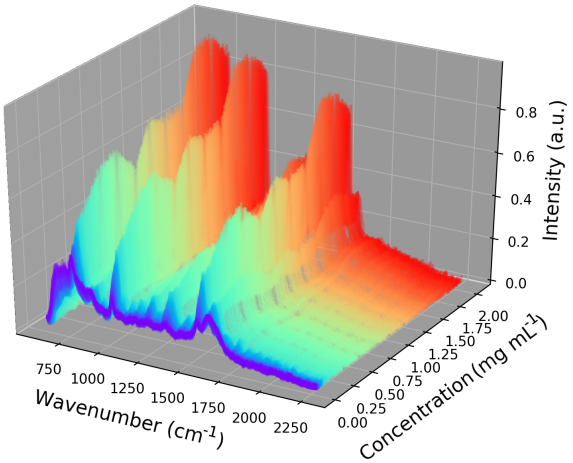
Ideally, an alternative method would be capable of synthesising interstitial concentrations that are more accurate than those created under the assumption of a linear relationship. It would also be capable of producing a uniform distribution of concentrations from a set of non-uniform distributions sampled from discrete sample concentrations - which, through knowledge of the non-linear Raman scattering intensity of a sample, may have necessitated an increased number of measurements within a particular range of concentrations - thereby removing model bias towards more densely populated concentrations. Based on these considerations, the Trypt-2 dataset was used to test the alternative method.

An alternative data augmentation method was tested that fit a spline, per wavenumber, across

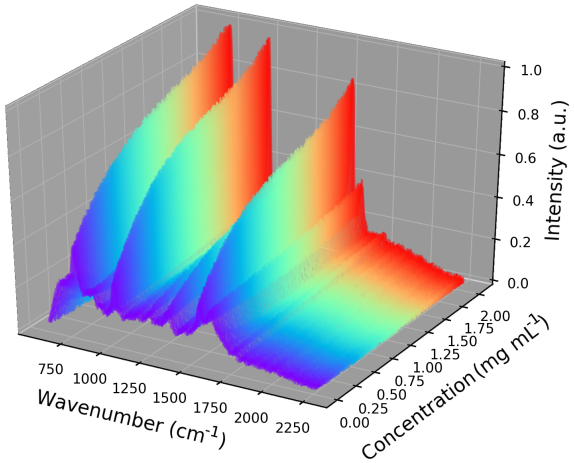
the discrete concentrations. The 10 spectra measured at each sample concentration were partitioned into the training, validation, and testing datasets with a ratio of 8:1:1. The training dataset was then averaged, increasing the SNR by an approximate factor of 2.8 - calculated from the signal noise that scales proportionally to the square root of the number of frames. Once all splines were fit to each wavenumber, new spectra could then be synthesised through a continuous range of concentrations by sampling at the same point along each spline for every wavenumber (see Figure 6.7). The concentration of each synthesised spectrum can be trivially calculated based on the distance between neighbouring discrete concentrations.



(a) Averaged discrete concentrations.



(b) Linearly-synthesised concentrations.



(c) Spline-synthesised concentrations.

Figure 6.7: Examples of the continuous distribution of concentrations synthesised from the Trypt-2 dataset via the linear (**b**) and spline (**c**) data augmentation methods. The discrete concentrations are independently averaged and shown in (**a**) for reference.

Degenerate peak heights in the discrete dataset (Figure 6.7a) are mimicked in the linear method (Figure 6.7b), as can be seen around 1.0 to 1.5 mg mL^{-1} on the concentration axis (the orange region).

There were a greater number of measurements taken at lower concentrations of tryptophan, as seen in 6.7a, which causes a greater number of low concentration samples to be synthesised by the linear method in comparison to the spline method. This is visualised by the increased density of coloured spectra in the blue-purple range for the linear method (Figure 6.7b) in comparison to the spline method (Figure 6.7c).

The spline data augmentation method, though capable of creating a uniform distribution of concentrations, was not selected to replace the linear data augmentation method. One reason for this is the reduced variance in synthesised spectra in comparison to the linear method - described in Subsection 5.1.2 of the previous chapter - resulting from the set of single splines fit across each wavenumber from which to generate spectra, which is an undesirable trait due to the small amount of available data. Repeated spectra synthesised at the same concentration would be identical, before the addition of noise, where the linear method is capable of producing different spectra due to sampling from individual, non-averaged frames. This was considered for the spline method, however it would drastically increase computation times due to having to fit 512 splines (the amount of wavenumber bins) for each application of the method, along with reducing the SNR of the resulting data.

Note that the curve of the spline-synthesised data in Figure 6.7c monotonically increased, proportional to increasing sample concentration, based on the smoothed cubic spline. This smoothing factor was specified to model an increasing relationship between peak intensity and sample concentration, and is satisfied when the following condition is met:

$$\sum_{i=1}^{N}(y_i - f(x_i))^2 \leq s, \tag{6.1}$$

where y_i is the intensity at each of the N concentrations for the specified wavenumber, $f(x_i)$ is value at each concentration on the fitting spline, and s is the smoothing factor.

As mentioned previously, there is a non-linear relationship between peak height and sample concentration that exists for each of the three datasets (see Figure 6.7a). The source of this non-linearity is a result of the differences in Raman cross-section between the mixture components and the size of the analyte molecules. This behaviour is described by the LIDAR equation [251, 252], which governs the signal intensity, S, from an analyte across all wavenumbers:

$$S = N\lambda\alpha e^{-2\tau}, \tag{6.2}$$

where N is the number of scattering centres, α is the Raman scattering cross-section of the analyte, τ is the optical depth of the medium between instrument and analyte, and λ represents additional constant variables, defined as

$$\lambda = \frac{L_p\, V(R)\, A\, O_e\, D_{QE}\, I(R)\, \Delta R}{\pi R^2}, \tag{6.3}$$

where L_p is the laser power, V(R) is the overlap integral of the outgoing laser, A is the collecting area, O_e is the optical efficiency, D_{QE} is the quantum efficiency of the detector, I(R) is the overlap integral between incoming and outgoing beams, ΔR is the sample depth, and R is the distance to the analyte.

For smaller molecules measured via spontaneous Raman spectroscopy, based on a conventional spectrometer setup with a typical central wavelength laser, the exponential term in Equation 6.2 is negligible. As such, doubling the sample concentration would result in a doubling of the signal intensity. However, as the IgG and tryptophan databases used in these experiments were measured using a deep UV spectrometer with a low central wavelength laser under resonance conditions, which amplifies the size of the α term, the exponential term becomes important. This results in the signal intensity from these samples increasing non-linearly with concentration, hence a monotonically increasing fit between concentrations becomes inviable.

Additionally, where the number of scattering centres, N, in smaller molecules would mutually increase with higher α values in increasing concentrations, the larger size of the biomolecules causes N to decrease due to attenuation from the sample. The effect of this, in combination with the relevant exponential term for IgG and tryptophan, results in a decrease in signal intensity as the concentration increases to a certain point. Figures 6.7a and 6.7b demonstrate this beyond the averaged $1.8190\,\mathrm{cm}^{-1}$ sample concentration in tryptophan. It should be noted that the reduction in average peak height around the $1.0150\,\mathrm{cm}^{-1}$ sample in the same figures is due to an error in measurement, which would result in the same inherent, non-physical flaw in either data augmentation method.

Therefore, as a result of the non-linear behaviour of the signal intensity as described by the LIDAR equation, the spline method for data augmentation becomes an inviable alternative to replace the linear data augmentation method due to the monotonically increasing nature of the resultant spectra. Beyond that, the augmented spectra seen in Figure 6.7c represent data contributions from a smoothed fit of the peak heights, rather than directly from individual frames as in the linear method. Even should the spline be used without smoothing, there would remain a dubious, non-physical assumption of the behaviour of Raman spectra at interstitial concentrations. To resolve this, more measurements must be taken at a higher concentration resolution, which is both impractical and time consuming from an industry standpoint, and thus should ideally be avoided. Because of these drawbacks, the linear method remained as the selected data augmentation strategy.

6.2.3 Results and Comparison to Industry Standard Methods

As in the previous chapter, the complete set of permutations (mixtures datasets, regression models, and data processing procedures) were tested. As before, the AE-Ridge model was retrained and re-evaluated on the unaugmented mixtures datasets to fulfil this requirement.

Table 6.2: Results of evaluating the three regression methods trained on the three mixtures datasets without data augmentation. The value N represents the number of components each dataset had after dimensionality-reduction. The best results are shown in bold.

Dataset	Method	Discrete			
		N	R^2	95% PI ($mg\,mL^{-1}$)	LoD ($mg\,mL^{-1}$)
IgG	PCR	5	**0.9872**	**±0.1404**	0.1148
	PLS	3	0.9798	±0.1763	0.1326
	AE-Ridge	128	0.9804	±0.1738	**0.0838**
Trypt-2	PCR	66	0.7998	±0.6216	0.2790
	PLS	3	0.7917	±0.6340	0.2689
	AE-Ridge	128	**0.8383**	**±0.5585**	**0.1563**
Trypt-5	PCR	86	**0.5263**	**±1.6940**	0.8634
	PLS	3	0.5161	±1.7122	0.9000
	AE-Ridge	128	0.4253	±1.8658	**0.7834**

Table 6.3: Results of evaluating the three regression methods trained on the three mixtures datasets, both with data augmentation. The value N represents the number of components each dataset had after dimensionality-reduction. The best results are shown in bold.

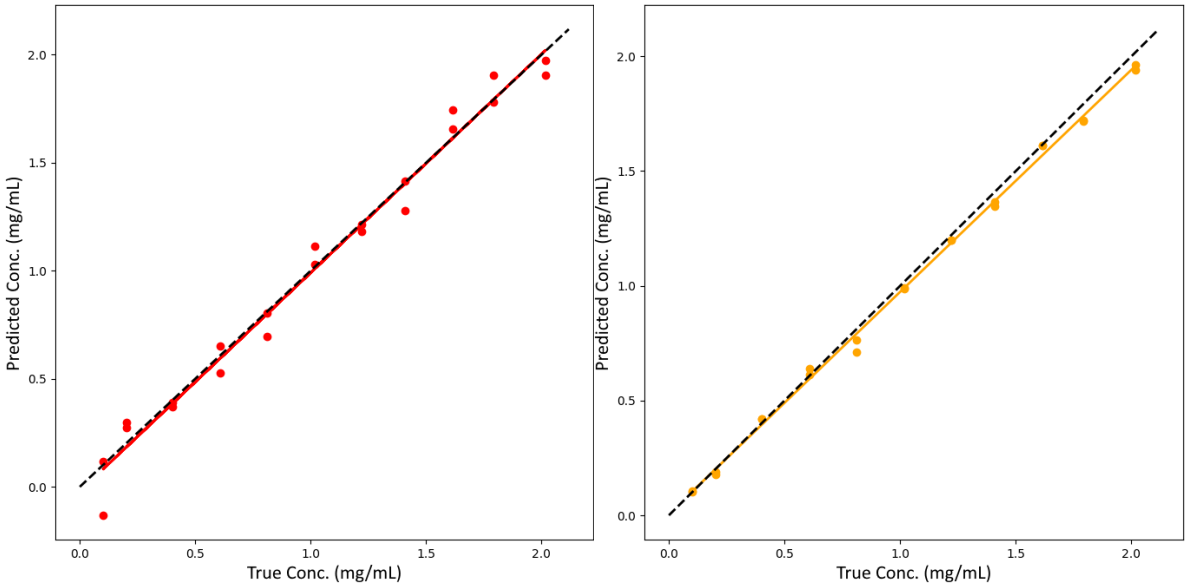
Dataset	Method	Augmented			
		N	R^2	95% PI ($mg\,mL^{-1}$)	LoD ($mg\,mL^{-1}$)
IgG	PCR	6	0.9890	±0.1303	0.0340
	PLS	14	0.9872	±0.1406	0.0287
	AE-Ridge	128	**0.9951**	**±0.0871**	**0.0179**
Trypt-2	PCR	311	0.9162	±0.4007	0.1119
	PLS	16	0.8588	±0.5219	0.1176
	AE-Ridge	128	**0.9758**	**±0.2126**	**0.0534**
Trypt-5	PCR	11	0.6503	±1.4555	0.3945
	PLS	29	0.4821	±1.7713	0.4470
	AE-Ridge	128	**0.7221**	**±1.2975**	**0.3090**

Tables 6.2 and 6.3 demonstrate the advantage of machine learning in the ability to learn complex, non-linear relationships. The necessity for a representative data augmentation procedure is also highlighted, as the AE-Ridge model substantially outperforms the PCR and PLS regression models on all three datasets. Interestingly, data augmentation improved the performance of all regression models across all datasets, with the exception of PLS regression trained on the Trypt-5 dataset that decreased in R^2 and 95% PI performance. This could be attributed to the inability of the data augmentation procedure to accurately synthesise interstitial concentrations between measured concentrations with a large separation, as in the case for the 2 and 5 $mg\,mL^{-1}$ samples, but retaining an improvement to the LoD alongside the other regression models.

For the IgG dataset, the AE-Ridge model was out-performed by PCR on the discrete data with the exception of LoD, but succeeded PCR in all metrics when trained on augmented data - particularly on the 95% PI metric, which shows an approximate 50% improvement. With regards to the tryptophan datasets, the performance of all regression models were lower than the IgG counterparts owing to the increased non-uniformity of the measured concentrations. When trained on the Trypt-2 dataset, the AE-Ridge model outperformed both PCR and PLS regression regardless of the use of data augmentation - though the performances of all models were improved by its inclusion. Whereas, when trained on the Trypt-5 dataset, the AE-Ridge model only surpassed the other methods with the inclusion of the data augmentation procedure. This suggests that the AE-Ridge model is limited in the ability to learn non-uniform separations in data. The results demonstrate that data augmentation enables machine learning models to gain a more significant improvement in performance than both PCR and PLS regression. Additionally, the lack of data augmentation on the Trypt-5 discrete dataset, and the subsequent poor performance, emphasises the fact that most machine learning models - in particular deep learning models - are data hungry.

6.2.4 Effects of Modifying Data Augmentation Process on Model Performance

The effect of data augmentation when applied to machine learning are showcased in Figure 6.8, in combination with the results in Tables 6.2 and 6.3 in the previous section. The range of predicted concentrations is both more accurate and precise due to data augmentation, which benefits from the near-uniform separation of measured concentrations.

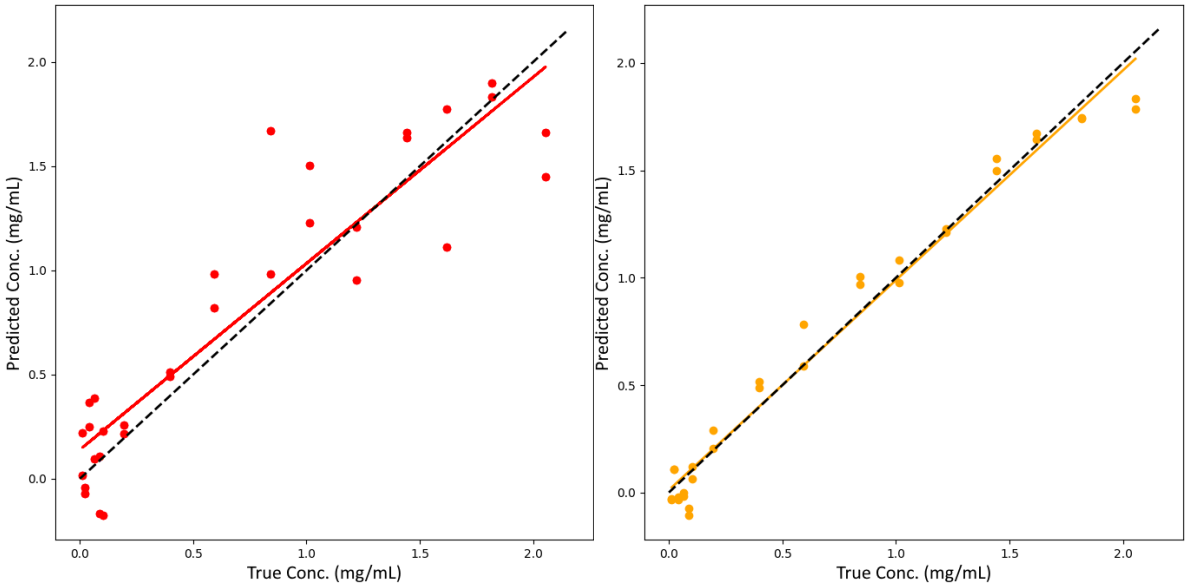


(a) IgG discrete model test predictions.

(b) IgG augmented model test predictions.

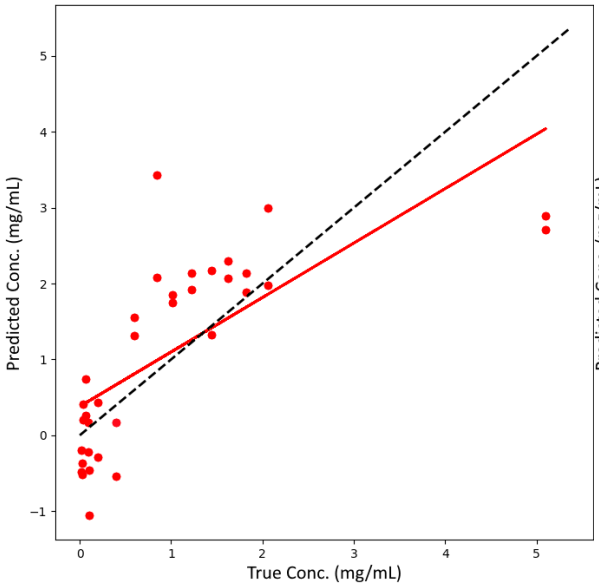
Figure 6.8: Concentration predictions using the AE-Ridge regression model on the IgG dataset both with and without data augmentation. The dashed black lines shows the identity line, and the orange and red lines show the trend lines for the plotted data points. Regression metrics are shown in Tables 6.2 and 6.3.

As mentioned, the data augmentation procedure benefits from samples being measured at uniformly distributed concentrations. The trend lines plotted for both the discrete and augmented variant models highlight the improvement to each regression model with the exclusion of the outlier sample at around $5\,\mathrm{mg\,mL^{-1}}$ (see Figure 6.9). Additionally, regardless of the data augmentation procedure implemented, the raw data must be faithful to the concentration that it is labelled to represent. As the data augmentation procedure synthesises interstitial concentrations in a linear fashion, any flaws in the raw data are represented alongside the desired distinguishing features. Figure 6.7b in Subsection 6.2.2 showcases the effect of this, in which synthesised spectra around the $1.0150\,\mathrm{mg\,mL^{-1}}$ sample have a downwards trend in the intensity of the three main peaks as a result of measurement errors. This effect is seen in Figure 6.9b as the data points around that concentration feature a low frequency oscillation. The AE-Ridge model trained on the Trypt-2 augmented dataset displays a minor plateauing of predictions beyond approximately $1.5\,\mathrm{mg\,mL^{-1}}$, suggesting that the regression model has learned a data representation based partially off of a simple peak-height estimation alongside the non-linear behaviour of the signal intensity (as explained in Subsection 6.2.2).

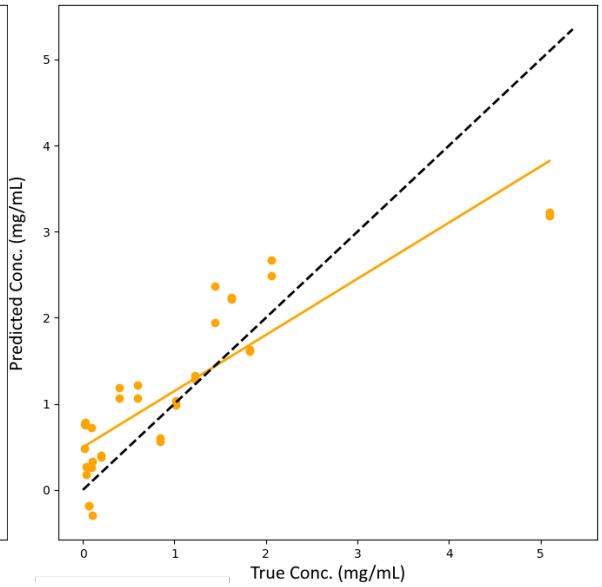


(a) Trypt-2 discrete model test predictions.

(b) Trypt-2 augmented model test predictions.



(c) Trypt-5 discrete model test predictions.



(d) Trypt-5 augmented model test predictions.

Figure 6.9: Concentration predictions using the AE-Ridge regression model on the Trypt-2 and Trypt-5 datasets both with and without data augmentation. The dashed black lines shows the identity line, and the orange and red lines show the trend lines for the plotted data points. Regression metrics are shown in Tables 6.2 and 6.3.

As mentioned in Subsection 6.2.2, the data augmentation method employed benefits greatly from the raw dataset being measured at uniformly-spaced concentrations. The effect of this is demonstrated in Subsection 6.2.3, Tables 6.2 and 6.3 by the improved model performance of IgG dataset in comparison to the two tryptophan datasets that possess notably worse data uniformity. An exploratory test was performed to improve model performance on both the Trypt-2 and Trypt-5 augmented datasets through a modification to the data augmentation process. The respective models were retrained on the same data as before, but with the natural logarithm applied to all associated labels (concentrations) at the data augmentation stage. The logarithmic form of all spectra were used throughout the training and inference processes, and only exponentiated back to the true values in order to convert concentration predictions into the desired units. This conversion shifted the measured concentrations closer towards uniform intervals (see Figure 6.10), which had the effect of greatly improving model performance, owing to the benefit of uniform data sampling to the data augmentation strategy.

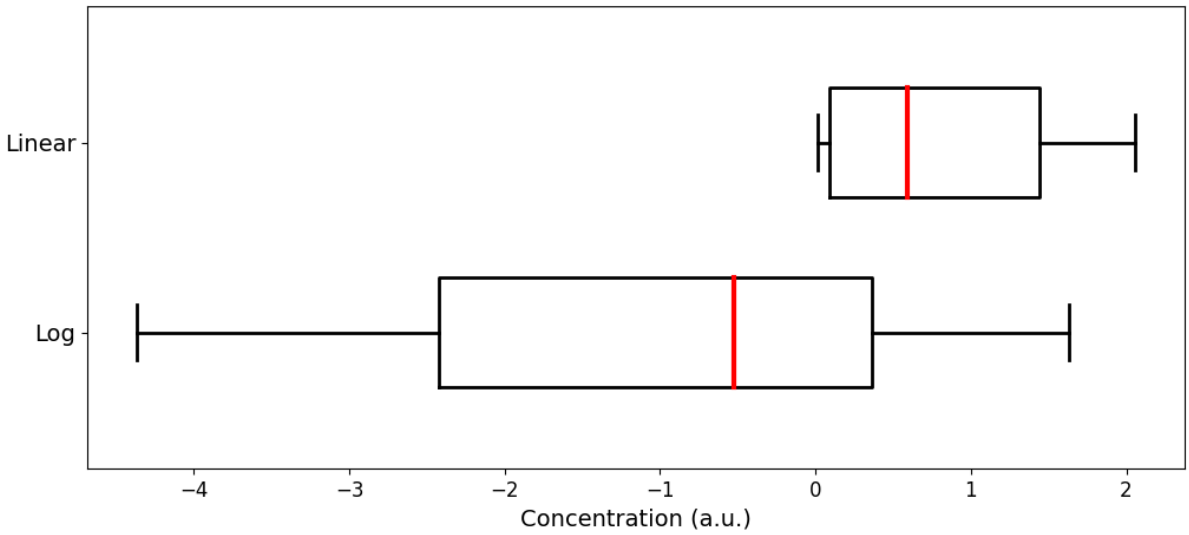


Figure 6.10: One-dimensional representation of the Trypt-2 concentrations before and after taking the natural logarithm, showing the improved uniform-spacing of discrete concentrations as a result of the transformation. The concentration of the linear values are in units of $\mathrm{mg\,mL^{-1}}$, thus the log values are in units of the natural logarithm of $\mathrm{mg\,mL^{-1}}$.

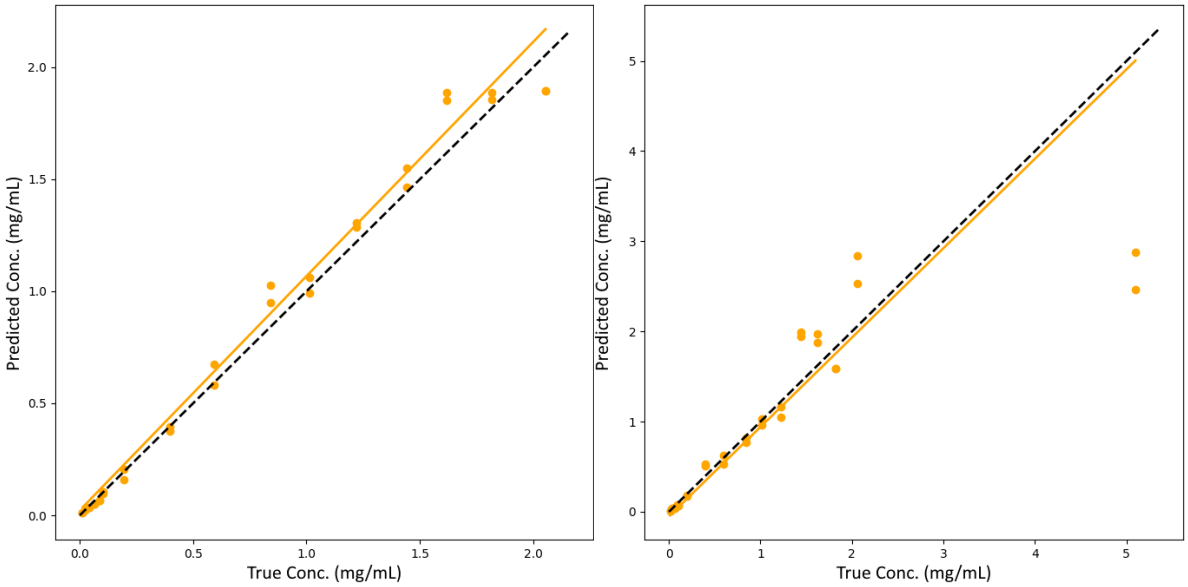
Table 6.4 shows an improvement to the Trypt-2 augmented dataset, and drastic improvement to the Trypt-5 dataset, in terms of the R^2 metric, by utilising the natural logarithm of the data labels at the data augmentation stage. The LoD performance saw an approximate 20% improvement on the Trypt-2 dataset, and a 30% improvement on the Trypt-5 dataset, due to the increased data uniformity. However, the 95% PI metric remained in close proximity between 'standard' and 'logarithmic' versions of each tryptophan regression model, despite the large improve to the R^2 value. Due to the math-

ematical nature of this procedure, and the increased density of low concentration samples that were measured, the resulting models are more accurate at predicting samples with a lower concentration.

Table 6.4: Results of training the Trypt-2 and Trypt-5 datasets by taking the natural logarithm of the labels (concentrations), in comparison to the standard values. The values for each metric were exponentiated back to $\mathrm{mg\,mL^{-1}}$ for the comparison. The value N represents the number of components each dataset had after dimensionality-reduction. The best results are shown in bold.

Dataset	Standard Labels				Logarithmic Labels			
	N	R^2	95% PI ($\mathrm{mg\,mL^{-1}}$)	LoD ($\mathrm{mg\,mL^{-1}}$)	N	R^2	95% PI ($\mathrm{mg\,mL^{-1}}$)	LoD ($\mathrm{mg\,mL^{-1}}$)
Trypt-2	128	0.9758	±0.2126	0.0534	128	**0.9932**	**±0.1835**	**0.0456**
Trypt-5	128	0.7221	±1.2975	0.3090	128	**0.9764**	**±1.2666**	**0.2346**

As the concentration of a sample increases, the model begins to diverge in its predictive capability, as can be seen in Figure 6.11, in which the line of best fit diverges from the identity line at higher concentrations (though this effect is harder to see in 6.11b owing to the difference in scale). The nature of this divergence would explain the retention of the high 95% PI values. Despite this, the results demonstrate a clear improvement in the performance of the tested regression models owing to the increased uniformity in the sample data - placing the performance of the 'logarithmic' Trypt-2 model, with data augmentation, closer in line with that of the IgG counterpart model (see Table 6.3). This method may also allow for the incorporation of the outlier sample around $5\,\mathrm{mg\,mL^{-1}}$, providing that the reduction in model performance is acceptable in the bounds of the particular regression task, which heavily depends on the importance of the outlier samples.



(a) Trypt-2 logarithmic model predictions.

(b) Trypt-5 logarithmic model predictions.

Figure 6.11: Concentration predictions using the AE-Ridge regression model on the Trypt-2 and Trypt-5 augmented datasets, trained with logarithmic concentration labels. The dashed black lines shows the identity line, and the orange lines show the trend lines for the plotted data points. Regression metrics are shown in Table 6.4.

This result emphasises the advantage to the data augmentation strategy, and by extension, its positive impact on the performance of regression models, of measuring samples at uniform concentration intervals. Alternatively, specialised adaptations to online processing techniques, like the one outlined here, can alleviate the need for such rigid data collection prerequisites, which would otherwise impose impractical constraints from an industrial standpoint.

6.3 Conclusions

The goal of this chapter was to apply the machine learning regression technique, originally developed in Chapter 5 based on data relevant to a nuclear decommissioning process, to mixed spectra datasets critical in biopharmaceutical drug manufacturing. The two datasets processed within this chapter consisted of analyte proteins dissolved in aqueous solutions at low concentrations. A good SNR was obtained from the samples by leveraging the UVRRS system to amplify the Raman response. This amplification was a result of multiple factors, including: the shorter wavelength Raman laser enhancing the Raman scattering intensity; the electronic transition of the organic molecules being approached, causing a resonant effect; and a spectral separation of the Raman signal from sample fluorescence.

As in Chapter 5, the AE-Ridge regression model was evaluated against industry standard regression

tools, PCR and PLS regression, and was found to greatly exceed the precision of predicted concentrations on both datasets through the use of data augmentation. Using the next best model as the base for comparisons in all cases, the AE-Ridge model outperformed the next best model on the 95% PI metric by 50% for IgG, 88% for Trypt-2, and 12% for Trypt-5. All IgG regression models performed well on the R^2 metric, with the AE-Ridge model having a 0.6% increase in performance over both PCR and PLS regression. This small difference owes to the high SNR of the IgG data, and the uniformity of sample measurements. With regards to the tryptophan datasets, there were more substantial improvements to the accuracies of both AE-Ridge models, possessing 7% and 11% increases in the R^2 metric for Trypt-2 and Trypt-5, respectively. The LoD was substantially improved by the AE-Ridge model on all datasets: 90% for IgG, 110% for Trypt-2, and 28% for Trypt-5, demonstrating both the benefit of machine learning and data augmentation processes in increasing the confidence level in handling low concentration data samples, and the detrimental effect of non-uniform data in the Trypt-5 dataset.

By comparing the results of the modifications to the data augmentation technique (as proposed in Subsection 6.2.4) to the unmodified processes on the AE-Ridge models, the benefit of the modification becomes evident. With regards to the R^2 metric, the AE-Ridge model performance improved on the Trypt-2 model by only 2%, however the Trypt-5 model improved by 35%, aligning with the goal of improving model performance through increased data uniformity. The LoD further improved by 17% on Trypt-2 and 32% on Trypt-5. However, as shown in Figure 6.11, careful consideration must be given to the nature of any modifications made to the sampling strategy used by the data augmentation technique, as the accuracy of the trendline begins to diverge at greater concentrations, as a result of the exponential behaviour. Such modifications, though maintaining poor outlier sample predictions on the Trypt-5 model (having only increased by 2% on the 95% PI metric), allowed for a performance on the AE-Ridge model that surpassed the R^2 metric for both PCR and PLS regression trained on the Trypt-2 dataset - a considerably easier regression task. This result shows potential for the inclusion of outlier samples within an arbitrary dataset without adversely affecting the performance of standard samples. However, modifications to either the regression model or the data augmentation technique would be required in order to improve predictions made on outlier samples. Although, the non-linear nature of the Raman response from the organic macromolecules used in this chapter may place a limit on such potential.

To conclude, the results presented within this chapter - in combination with the previous chapter - demonstrate the capacity of the AE-Ridge regression model to produce both accurate and precise predictions on a range of data from different industries, at both low and high concentrations, and measured using different Raman spectroscopy techniques. By considering the non-linear Raman response of the organic macromolecules used within this chapter, substantial gains in model performance are achieved through a simple modification to the sampling strategy at the data augmentation stage. However, there is interest in making further modifications to the data augmentation technique with the goal of improving outlier performance. Future work in this area, similar to that of Chapter 5,

will look to expand the functionality of the regression model to benefit from transfer learning, which would allow the regression model to adapt to future datasets. This is increasingly becoming a crucial aspect for a regression model to have in a biopharmaceutical setting, as a result of the fast turnaround required by the ever-expanding range of organic molecules used throughout protein biologics research, and in the manufacturing of new therapeutic drugs.

Chapter 7:

Outlook and Future Work

The content presented in Chapters 3 and 4, in collaboration with members of the Baumberg research group at the University of Cambridge, demonstrate the benefits of the robust data analysis technique to nanotechnology research. By treating the SERS data captured for this research as images, a combination of machine learning architectures and image processing techniques were used to analyse picocavity spectra produced by single molecules interacting with metal nanostructures. From extracting and isolating transient spectral features, to clustering Configurations representing picocavity events, the technique developed in Chapter 3 offers an end-to-end pipeline for the processing of SERS data. This has lead to a powerful tool for studying the formation dynamics of picocavity events, through a comparison to vibrational modes calculated from DFT simulations adapted to account for the effects of a picocavity field gradient.

By producing a near-field map to reveal the most likely adatom position responsible for creating a local field gradient, and statistically analysing the formation rates and mean formation times of several NPoM varieties, Chapter 3 quantitatively showed that picocavity formation was suppressed on metal surfaces functionalised with palladium. This has provided a method to verify the desirable effects of catalyst surface or near-surface modifications. Chapter 4 introduced an extension to the data analysis pipeline by incorporating the full suite of spatiotemporal information present in SERS data - treated as images - to classify the polarity of correlated picocavity peaks. Using these predicted outputs, the resulting correlation matrix built during the course of this analysis and described in Chapter 4 has enabled a means to analyse how specific vibrational modes act across single molecules, and evaluate the effects of any proposed modifications to catalyst surfaces. An example of which would be surface doping with palladium to reduce the occurrence of undesirable picocavity event types, and subsequently reduce undesirable by-products from catalytic processes.

Thus, the foundational research in Chapter 4 extends the work done in Chapter 3, with the goal of advancing the field of heterogeneous catalysis by enabling a method of evaluating tailored catalyst designs, aided by the spatiotemporal analysis technique developed within the chapter. Future work might expand the functionality of the Siamese-CNN model to provide a fine-grained evaluation of correlated modes, and thence a better understanding of such interactions. This research has many practical applications in improving the selectivity and efficiency of catalytic processes, such as the Haber-Bosch and CO_2 reduction processes described in Chapter 4, showing clear global benefits to advancements in this field.

In Chapters 5 and 6, a competitive machine learning regression tool was developed in collaboration

with IS-Instruments Ltd. that outperformed industry standard tools PCR and PLS regression used as comparative benchmarks within this study. By separately training the AE-Ridge regression model on datasets from both nuclear and biopharmaceutical industries, it was proven that the model could effectively adapt across a wide variety of data. This encompassed data with varying magnitudes in concentration, different Raman spectroscopy measurement methods (spontaneous and RRS), and analyte molecule sizes. The final scenario, due to sample attenuation, resulted in non-linear variations of Raman responses with changes in sample concentration for both biopharmaceutical datasets.

An important theme within these chapters is the consideration of practical challenges in the real world. Due to the common occurrence of low data volumes in industrial applications, both the neural network architecture and data augmentation process were designed to overcome issues with limited sample variance and consequent model underfitting. By utilising the data augmentation process developed in Chapter 5, and adapting the sampling strategy in Chapter 6 to account for non-uniform data measurements, the AE-Ridge regression model was shown to predict unknown mixture concentrations with substantially greater precision in comparison to conventional regression tools across all datasets.

In addition to the novel data analysis pipeline introduced in Chapter 3 and extended upon in Chapter 4, as well as the competitive industrial regression model developed and applied in Chapters 5 and 6, there are aspects of these neural network processes that would benefit from improvements. Namely in improvements to neural network model performance to increase the accuracy, precision, and efficiency of each model where applicable. This would likely constitute refinements to the machine learning algorithms, optimising model hyperparameters, or testing alternative architectures. For example, the CAE developed in Chapter 3 extracts picocavity signals from individual spectra, however as suggested in the conclusion of Chapter 3, transformer architectures have been shown to perform state-of-the-art classification on hyper-spectral Raman datasets [211, 212], which may have the potential to jointly process the spatiotemporal features of SERS scans. This could provide a more reliable extraction of weaker picocavity peaks that often appear at lower wavenumbers. Another beneficial change would be the throughput of greater data volumes, as the current clustering process is limited by the silhouette score evaluation metric in creating data clusters that can represent the complex and diverse range of picocavity event types. Hierarchical [213] or density-based [214] clustering techniques may overcome this challenge.

As mentioned in the conclusions to Chapter 4, adapting both the dataset assembly and labelling process, and the Siamese-CNN model architecture (potentially only in the output layer), would allow for the magnitude of correlated picocavity peaks to be incorporated, thus improve the accuracy of predicted adatom locations, and hence increase the wealth of information gathered by this method to inform future catalyst modifications.

Following the theme of future improvements to model capabilities, there is an industrial incentive to extend the functionality of the AE-Ridge model utilised in Chapters 5 and 6 to include additional

analytes in the mixture sample databases. This extension, from the perspective of nuclear decommissioning, would provide a regression model encompassing the full range of expected contaminants. From a biopharmaceutical standpoint, the use of transfer learning is desirable in order to create a robust regression model able to quickly adapt to new datasets produced within the growing market.

An active area of machine learning worth exploring within the context of this thesis is that of explainable artificial intelligence (XAI). As the vast majority of machine learning algorithms are 'black boxes', there are numerous techniques that exist or are in development that seek to provide human-interpretable reasoning behind model outputs. This has clear benefits to scientific applications by providing transparency in AI model decisions, allowing for scientists to better understand the results of a model. The large-scale adoption of these techniques in various applications, including the control of therapeutic drug production within the biopharmaceutical sector, holds major significance. This is due to the substantial financial repercussions arising from incorrect decisions made by a neural network. Hence, there is a growing necessity to justify these decisions. One such example of XAI is gradient-weighted class activation mapping (Grad-CAM) [253], which uses the final convolutional layer of a CNN to provide a high resolution visual mapping, highlighting regions of an image that strongly contribute to a particular model decision. Grad-CAM could therefore be applied to the data used in Chapter 4 by the Siamese-CNN to explain model predictions, and hence increase the trustworthiness of the approach.

At this stage, the robust data analysis pipeline developed in Chapter 3 provides a novel method to study single molecule SERS data. Extending this research in Chapter 4 to incorporate the temporal information in the SERS data, the data analysis pipeline has provided a unique insight into the coordination geometries of adatoms in metal surfaces, as well as interactions between individual vibrational modes on single molecules. It is believed at the present stage that this technique, enabled through the use of machine learning and image processing tools, can translate to many other analyte molecules as long as the required stable state exists from which the CAE can be trained. However, with modifications suggested in Chapters 3 and 4, this requirement may be overcome, expanding the list of molecules that can be processed. Thus this research contributes a powerful verification tool to the field of nanotechnology, as well as for more general spectral analysis across physics and chemistry, which can be used to assist in the rational design of heterogeneous catalysts.

References

[1] Alex Poppe, Jack Griffiths, Shu Hu, Jeremy J. Baumberg, Margarita Osadchy, Stuart Gibson, and Bart de Nijs. Mapping Atomic-Scale Metal-Molecule Interactions: Salient Feature Extraction through Autoencoding of Vibrational Spectroscopy Data. *The Journal of Physical Chemistry Letters*, 14(34):7603–7610, August 2023. Publisher: American Chemical Society.

[2] L. Claron Hoskins. Pure rotational raman spectroscopy of diatomic molecules. *Journal of Chemical Education*, 52(9):568, 1975.

[3] Kārlis Bērziņš, Sara J. Fraser-Miller, and Keith C. Gordon. Recent advances in low-frequency raman spectroscopy for pharmaceutical applications. *International Journal of Pharmaceutics*, 592:120034, 2021.

[4] H. S. A. Molecular Diffraction of Light. *Nature*, 110(2763):505–506, October 1922. Number: 2763 Publisher: Nature Publishing Group.

[5] C. V. Raman. A new radiation. *Proceedings of the Indian Academy of Sciences - Section A*, 37(3):333–341, March 1953.

[6] Paolo Ranzieri, Alberto Girlando, Silvia Tavazzi, Marcello Campione, Luisa Raimondo, Ivano Bilotti, Aldo Brillante, Raffaele G. Della Valle, and Elisabetta Venuti. Polymorphism and phonon dynamics of alpha-quaterthiophene. *Chemphyschem: A European Journal of Chemical Physics and Physical Chemistry*, 10(4):657–663, March 2009.

[7] Trang Thi Thu Nguyen, Yejin Kim, Soungmin Bae, Maryam Bari, Hye Ri Jung, William Jo, Yong-Hoon Kim, Zuo-Guang Ye, and Seokhyun Yoon. Raman Scattering Studies of the Structural Phase Transitions in Single-Crystalline CH3NH3PbCl3. *The Journal of Physical Chemistry Letters*, 11(10):3773–3781, May 2020. Publisher: American Chemical Society.

[8] Yao-Hui Wang, Shisheng Zheng, Wei-Min Yang, Ru-Yu Zhou, Quan-Feng He, Petar Radjenovic, Jin-Chao Dong, Shunning Li, Jiaxin Zheng, Zhi-Lin Yang, Gary Attard, Feng Pan, Zhong-Qun Tian, and Jian-Feng Li. In situ Raman spectroscopy reveals the structure and dissociation of interfacial water. *Nature*, 600(7887):81–85, December 2021. Number: 7887 Publisher: Nature Publishing Group.

[9] Yilin Deng, Albertus D. Handoko, Yonghua Du, Shibo Xi, and Boon Siang Yeo. In Situ Raman Spectroscopy of Copper and Copper Oxide Surfaces during Electrochemical Oxygen Evolution Reaction: Identification of CuIII Oxides as Catalytically Active Species. *ACS Catalysis*, 6(4):2473–2481, April 2016. Publisher: American Chemical Society.

[10] Alaa Y. Faid, Alejandro Oyarce Barnett, Frode Seland, and Svein Sunde. Ni/nio nanosheets for alkaline hydrogen evolution reaction: In situ electrochemical-raman study. *Electrochimica Acta*, 361:137040, 2020.

[11] Juhyung Choi, Daekyu Kim, Weiran Zheng, Bingyi Yan, Yong Li, Lawrence Yoon Suk Lee, and Yuanzhe Piao. Interface engineered nife2o4-x/nimoo4 nanowire arrays for electrochemical oxygen evolution. *Applied Catalysis B: Environmental*, 286:119857, 2021.

[12] Weiran Zheng, Mengjie Liu, and Lawrence Yoon Suk Lee. Electrochemical Instability of Metal–Organic Frameworks: In Situ Spectroelectrochemical Investigation of the Real Active Sites. *ACS Catalysis*, 10(1):81–92, January 2020. Publisher: American Chemical Society.

[13] G.A. Hope, R. Woods, and C.G. Munce. Raman microprobe mineral identification. *Minerals Engineering*, 14(12):1565–1577, 2001. Applied Mineralogy. Papers presented at Applied Mineralogy '01, Brisbane, Australia.

[14] Saikat Roy, Brianna Chamberlin, and Adam J. Matzger. Polymorph Discrimination using Low Wavenumber Raman Spectroscopy. *Organic process research & development*, 17(7):976–980, July 2013.

[15] Thi Huyen Nguyen, Thi Minh Hien Nguyen, Boyoun Kang, Beongki Cho, Mancheon Han, Hyoung Joon Choi, Mihye Kong, Yongjae Lee, and In-Sang Yang. Raman spectroscopic evidence of impurity-induced structural distortion in smb6. *Journal of Raman Spectroscopy*, 50(11):1661–1671, 2019.

[16] P. J. Caspers, G. W. Lucassen, and G. J. Puppels. Combined In Vivo Confocal Raman Spectroscopy and Confocal Microscopy of Human Skin. *Biophysical Journal*, 85(1):572–580, July 2003.

[17] Dominique Lunter, Victoria Klang, Dorottya Kocsis, Zsófia Varga-Medveczky, Szilvia Berkó, and Franciska Erdő. Novel aspects of Raman spectroscopy in skin research. *Experimental Dermatology*, 31(9):1311–1329, September 2022.

[18] J. Wohlrab, A. Vollmann, S. Wartewig, W. C. Marsch, and R. Neubert. Noninvasive characterization of human stratum corneum of undiseased skin of patients with atopic dermatitis and psoriasis as studied by Fourier transform Raman spectroscopy. *Biopolymers*, 62(3):141–146, 2001.

[19] K. U. Schallreuter, J. Moore, J. M. Wood, W. D. Beazley, D. C. Gaze, D. J. Tobin, H. S. Marshall, A. Panske, E. Panzig, and N. A. Hibberts. In vivo and in vitro evidence for hydrogen peroxide (H2O2) accumulation in the epidermis of patients with vitiligo and its successful removal by a UVB-activated pseudocatalase. *The Journal of Investigative Dermatology. Symposium Proceedings*, 4(1):91–96, September 1999.

[20] Annieke Nijssen, Tom C. Bakker Schut, Freerk Heule, Peter J. Caspers, Donal P. Hayes, Martino H. A. Neumann, and Gerwin J. Puppels. Discriminating basal cell carcinoma from its surrounding tissue by Raman spectroscopy. *The Journal of Investigative Dermatology*, 119(1):64–69, July 2002.

[21] Fay Nicolson, Moritz F. Kircher, Nick Stone, and Pavel Matousek. Spatially offset Raman spectroscopy for biomedical applications. *Chemical Society Reviews*, 50(1):556–568, January 2021. Publisher: The Royal Society of Chemistry.

[22] P. Matousek, I. P. Clark, E. R. C. Draper, M. D. Morris, A. E. Goodship, N. Everall, M. Towrie, W. F. Finney, and A. W. Parker. Subsurface Probing in Diffusely Scattering Media Using Spatially Offset Raman Spectroscopy. *Applied Spectroscopy*, 59(4):393–400, April 2005. Publisher: SAGE Publications Ltd STM.

[23] P. Matousek, M. D. Morris, N. Everall, I. P. Clark, M. Towrie, E. Draper, A. Goodship, and A. W. Parker. Numerical Simulations of Subsurface Probing in Diffusely Scattering Media Using Spatially Offset Raman Spectroscopy. *Applied Spectroscopy*, 59(12):1485–1492, December 2005. Publisher: Society for Applied Spectroscopy.

[24] Charlotte Eliasson and Pavel Matousek. Noninvasive Authentication of Pharmaceutical Products through Packaging Using Spatially Offset Raman Spectroscopy. *Analytical Chemistry*, 79(4):1696–1701, February 2007. Publisher: American Chemical Society.

[25] Matthew Bloomfield, Paul W. Loeffen, and Pavel Matousek. Detection of concealed substances in sealed opaque plastic and coloured glass containers using SORS. In *Optics and Photonics for Counterterrorism and Crime Fighting VI and Optical Materials in Defence Systems Technology VII*, volume 7838, pages 51–65. SPIE, October 2010.

[26] Paul W. Loeffen, Guy Maskall, Stuart Bonthron, Matthew Bloomfield, Craig Tombling, and Pavel Matousek. Spatially offset Raman spectroscopy (SORS) for liquid screening. In *Optics and Photonics for Counterterrorism and Crime Fighting VII; Optical Materials in Defence Systems Technology VIII; and Quantum-Physics-based Information Security*, volume 8189, pages 78–87. SPIE, October 2011.

[27] William J. Olds, Esa Jaatinen, Peter Fredericks, Biju Cletus, Helen Panayiotou, and Emad L. Izake. Spatially offset raman spectroscopy (sors) for the analysis and detection of packaged pharmaceuticals and concealed drugs. *Forensic Science International*, 212(1):69–77, 2011.

[28] Paul W. Loeffen, Guy Maskall, Stuart Bonthron, Matthew Bloomfield, Craig Tombling, and Pavel Matousek. Chemical and explosives point detection through opaque containers using spatially offset Raman spectroscopy (SORS). In *Chemical, Biological, Radiological, Nuclear, and Explosives (CBRNE) Sensing XII*, volume 8018, pages 387–395. SPIE, June 2011.

[29] C. Eliasson, N. A. Macleod, and P. Matousek. Noninvasive Detection of Concealed Liquid Explosives Using Raman Spectroscopy. *Analytical Chemistry*, 79(21):8185–8189, November 2007. Publisher: American Chemical Society.

[30] Infrared and Raman Spectroscopy in Forensic Science | Wiley.

[31] Matthew V. Schulmerich, Kathryn Dooley, Michael D. Morris, Thomas M. Vanasse, and Steven A. Goldstein. Transcutaneous fiber optic Raman spectroscopy of bone using annular illumination and a circular array of collection fibers. *Journal of Biomedical Optics*, 11(6):060502, November 2006. Publisher: SPIE.

[32] M. Z. Vardaki, C. G. Atkins, H. G. Schulze, D. V. Devine, K. Serrano, M. W. Blades, and R. F. B. Turner. Raman spectroscopy of stored red blood cell concentrate within sealed transfusion blood bags. *Analyst*, 143(24):6006–6013, December 2018. Publisher: The Royal Society of Chemistry.

[33] Nicholas Stone, Rebecca Baker, Keith Rogers, Anthony William Parker, and Pavel Matousek. Subsurface probing of calcifications with spatially offset Raman spectroscopy (SORS): future possibilities for the diagnosis of breast cancer. *Analyst*, 132(9):899–905, August 2007. Publisher: The Royal Society of Chemistry.

[34] Claudia Conti, Chiara Colombo, Marco Realini, Giuseppe Zerbi, and Pavel Matousek. Subsurface Raman Analysis of Thin Painted Layers. *Applied Spectroscopy*, 68(6):686–691, June 2014. Publisher: SAGE Publications Ltd STM.

[35] Mark E. Brezinski. 13 - other technologies. In Mark E. Brezinski, editor, *Optical Coherence Tomography*, pages 353–368. Academic Press, Burlington, 2006.

[36] D.C. Harris and M.D. Bertolucci. *Symmetry and Spectroscopy: An Introduction to Vibrational and Electronic Spectroscopy*. Dover Books on Chemistry Series. Dover Publications, 1989.

[37] Dennis P. Strommen and Kazuo Nakamoto. Resonance raman spectroscopy. *Journal of Chemical Education*, 54(8):474, August 1977. Publisher: American Chemical Society.

[38] R. S. Chao, R. K. Khanna, and E. R. Lippincott. Theoretical and experimental resonance raman intensities for the manganate ion. *Journal of Raman Spectroscopy*, 3(2-3):121–131, 1975.

[39] Académie des sciences (France) Auteur du texte. Comptes rendus hebdomadaires des séances de l'Académie des sciences, July 1946.

[40] Evtim V. Efremov, Freek Ariese, and Cees Gooijer. Achievements in resonance Raman spectroscopy review of a technique with a distinct analytical chemistry potential. *Analytica Chimica Acta*, 606(2):119–134, January 2008.

[41] Michael D. Morris and David J. Wallan. Resonance Raman spectroscopy. Current applications and prospects. *Analytical Chemistry*, 51(2):182A–192A, February 1979. Publisher: American Chemical Society.

[42] T G Spiro and P Stein. Resonance effects in vibrational scattering from complex molecules. *Annual Review of Physical Chemistry*, 28(1):501–521, 1977.

[43] Robin J. H. Clark and Trevor J. Dines. Resonance raman spectroscopy, and its application to inorganic chemistry. new analytical methods (27). *Angewandte Chemie International Edition in English*, 25(2):131–158, 1986.

[44] Roman S. Czernuszewicz and Marzena B. Zaczek. *Resonance Raman Spectroscopy*. John Wiley & Sons, Ltd, 2008.

[45] Thomas G. Spiro and Roman S. Czernuszewicz. [18] resonance raman spectroscopy of metalloproteins. In *Biochemical Spectroscopy*, volume 246 of *Methods in Enzymology*, pages 416–460. Academic Press, 1995.

[46] Rosalind Wolstenholme. *Raman Spectroscopy*, chapter 8, pages 161–183. John Wiley & Sons, Ltd, 2021.

[47] Danilo Bersani, Claudia Conti, Pavel Matousek, Federica Pozzi, and Peter Vandenabeele. Methodological evolutions of Raman spectroscopy in art and archaeology. *Analytical Methods*, 8(48):8395–8409, 2016. Publisher: Royal Society of Chemistry.

[48] EJ Woodbury and WK Ng. Ruby laser operation in the near ir. *proc. IRE*, 50(11):2347–2348, 1962.

[49] Richard C. Prince, Renee R. Frontiera, and Eric O. Potma. Stimulated Raman Scattering: From Bulk to Nano. *Chemical Reviews*, 117(7):5070–5094, April 2017. Publisher: American Chemical Society.

[50] Luigi Sirleto and Maria Antonietta Ferrara. Fiber amplifiers and fiber lasers based on stimulated raman scattering: A review. *Micromachines*, 11(3), 2020.

[51] Gisela Eckhardt, D. P. Bortfeld, and M. Geller. STIMULATED EMISSION OF STOKES AND ANTI-STOKES RAMAN LINES FROM DIAMOND, CALCITE, AND α-SULFUR SINGLE CRYSTALS. *Applied Physics Letters*, 3(8):137–138, November 2004.

[52] Nitzan Mayorkas, Shay Izbitski, Amir Bernat, and Ilana Bar. Simultaneous Ionization-Detected Stimulated Raman and Visible–Visible–Ultraviolet Hole-Burning Spectra of Two Tryptamine Conformers. *The Journal of Physical Chemistry Letters*, 3(5):603–607, March 2012. Publisher: American Chemical Society.

[53] Nitzan Mayorkas, Amir Bernat, Shay Izbitski, and Ilana Bar. Vibrational and vibronic spectra of tryptamine conformers. *The Journal of Chemical Physics*, 138(12):124312, March 2013.

[54] Christian W. Freudiger, Wei Min, Brian G. Saar, Sijia Lu, Gary R. Holtom, Chengwei He, Jason C. Tsai, Jing X. Kang, and X. Sunney Xie. Label-Free Biomedical Imaging with High Sensitivity by Stimulated Raman Scattering Microscopy. *Science*, 322(5909):1857–1861, December 2008. Publisher: American Association for the Advancement of Science.

[55] P. D. Maker and R. W. Terhune. Study of optical effects due to an induced polarization third order in the electric field strength. *Phys. Rev.*, 137:A801–A818, Feb 1965.

[56] R. F. Begley, A. B. Harvey, and R. L. Byer. Coherent anti-Stokes Raman spectroscopy. *Applied Physics Letters*, 25(7):387–390, 10 2003.

[57] Shaowei Li, Yanping Li, Rongxing Yi, Liwei Liu, and Junle Qu. Coherent Anti-Stokes Raman Scattering Microscopy and Its Applications. *Frontiers in Physics*, 8, 2020.

[58] W. M. Tolles, J. W. Nibler, J. R. McDonald, and A. B. Harvey. A review of the theory and application of coherent anti-stokes raman spectroscopy (cars). *Applied Spectroscopy*, 31(4):253–271, 1977.

[59] A. M. Zheltikov. Coherent anti-stokes raman scattering: from proof-of-the-principle experiments to femtosecond cars and higher order wave-mixing generalizations. *Journal of Raman Spectroscopy*, 31(8-9):653–667, 2000.

[60] V. D. Kobtsev, D. N. Kozlov, S. A. Kostritsa, V. V. Smirnov, and O. M. Stel'makh. Temperature fluctuations in turbulent flame measured using coherent anti-Stokes Raman scattering. *Technical Physics Letters*, 41(8):756–758, August 2015.

[61] Huijie Zhao, Ziyang Tian, Yan Li, and Haoyun Wei. Hybrid fs/ps vibrational coherent anti-stokes raman scattering for simultaneous gas-phase n2/o2/co2 measurements. *Opt. Lett.*, 46(7):1688–1691, Apr 2021.

[62] Yvette Zuzeek, Inchul Choi, Mruthunjaya Uddi, Igor V. Adamovich, and Walter R. Lempert. Pure rotational CARS thermometry studies of low-temperature oxidation kinetics in air and ethene–air nanosecond pulse discharge plasmas. *Journal of Physics D: Applied Physics*, 43(12):124001, March 2010.

[63] Igor V. Adamovich, Ting Li, and Walter R. Lempert. Kinetic mechanism of molecular energy transfer and chemical reactions in low-temperature air-fuel plasmas. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 373(2048):20140336, August 2015. Publisher: Royal Society.

[64] Shaowei Li, Yanping Li, Rongxing Yi, Liwei Liu, and Junle Qu. Coherent anti-stokes raman scattering microscopy and its applications. *Frontiers in Physics*, 8, 2020.

[65] Xiaolin Nan, Ji-Xin Cheng, and X. Sunney Xie. Vibrational imaging of lipid droplets in live fibroblast cells with coherent anti-Stokes Raman scattering microscopy. *Journal of Lipid Research*, 44(11):2202–2208, November 2003. Publisher: Elsevier.

[66] T.B. HUFF and J.-X. CHENG. In vivo coherent anti-stokes raman scattering imaging of sciatic nerve tissue. *Journal of Microscopy*, 225(2):175–182, 2007.

[67] Evan J. Blackie, Eric C. Le Ru, and Pablo G. Etchegoin. Single-molecule surface-enhanced raman spectroscopy of nonresonant molecules. *Journal of the American Chemical Society*, 131(40):14466–14472, 2009.

[68] E. C. Le Ru, E. Blackie, M. Meyer, and P. G. Etchegoin. Surface enhanced raman scattering enhancement factors: A comprehensive study. *The Journal of Physical Chemistry C*, 111(37):13794–13803, 2007.

[69] Katrin Kneipp, Yang Wang, Harald Kneipp, Lev T. Perelman, Irving Itzkan, Ramachandra R. Dasari, and Michael S. Feld. Single molecule detection using surface-enhanced raman scattering (sers). *Phys. Rev. Lett.*, 78:1667–1670, Mar 1997.

[70] Wei Xie, Bernd Walkenfort, and Sebastian Schlücker. Label-free sers monitoring of chemical reactions catalyzed by small gold nanoparticles using 3d plasmonic superstructures. *Journal of the American Chemical Society*, 135(5):1657–1660, 2013.

[71] Shu Tian, Oara Neumann, Michael J. McClain, Xiao Yang, Linan Zhou, Chao Zhang, Peter Nordlander, and Naomi J. Halas. Aluminum Nanocrystals: A Sustainable Substrate for Quantitative SERS-Based DNA Detection. *Nano Letters*, 17(8):5071–5077, August 2017. Publisher: American Chemical Society.

[72] Klaus Bo Mogensen, Marina Gühlke, Janina Kneipp, Shima Kadkhodazadeh, Jakob B. Wagner, Marta Espina Palanco, Harald Kneipp, and Katrin Kneipp. Surface-enhanced raman scattering on aluminum using near infrared and visible excitation. *Chem. Commun.*, 50:3744–3746, 2014.

[73] Jack Griffiths, Tamás Földes, Bart de Nijs, Rohit Chikkaraddy, Demelza Wright, William Deacon, Dénes Berta, Charlie Readman, David-Benjamin Grys, Edina Rosta, and Jeremy Baumberg. Resolving sub-angstrom ambient motion through reconstruction from vibrational spectra. *Nature Communications*, 12, 11 2021.

[74] Jack Griffiths, Bart de Nijs, Rohit Chikkaraddy, and Jeremy J. Baumberg. Locating single-atom optical picocavities using wavelength-multiplexed raman scattering. *ACS Photonics*, 8(10):2868–2875, 2021.

[75] Felix Benz, Mikolaj K. Schmidt, Alexander Dreismann, Rohit Chikkaraddy, Yao Zhang, Angela Demetriadou, Cloudy Carnegie, Hamid Ohadi, Bart de Nijs, Ruben Esteban, Javier Aizpurua, and Jeremy J. Baumberg. Single-molecule optomechanics in "picocavities". *Science*, 354(6313):726–729, November 2016.

[76] M. Fleischmann, P.J. Hendra, and A.J. McQuillan. Raman spectra of pyridine adsorbed at a silver electrode. *Chemical Physics Letters*, 26(2):163–166, 1974.

[77] B. Barbiellini. Enhancement of raman scattering from molecules placed near metal nanoparticles. *Low Temperature Physics*, 43(1):159–161, 2017.

[78] David L. Jeanmaire and Richard P. Van Duyne. Surface raman spectroelectrochemistry: Part i. heterocyclic, aromatic, and aliphatic amines adsorbed on the anodized silver electrode. *Journal of Electroanalytical Chemistry and Interfacial Electrochemistry*, 84(1):1–20, 1977.

[79] E.C. Le Ru and P.G. Etchegoin. Rigorous justification of the |e|4 enhancement factor in surface enhanced raman spectroscopy. *Chemical Physics Letters*, 423(1):63–66, 2006.

[80] Judith Langer, Dorleta Jimenez de Aberasturi, Javier Aizpurua, Ramon A. Alvarez-Puebla, Baptiste Auguié, Jeremy J. Baumberg, Guillermo C. Bazan, Steven E. J. Bell, Anja Boisen, Alexandre G. Brolo, Jaebum Choo, Dana Cialla-May, Volker Deckert, Laura Fabris, Karen Faulds, F. Javier García de Abajo, Royston Goodacre, Duncan Graham, Amanda J. Haes, Christy L. Haynes, Christian Huck, Tamitake Itoh, Mikael Käll, Janina Kneipp, Nicholas A. Kotov, Hua Kuang, Eric C. Le Ru, Hiang Kwee Lee, Jian-Feng Li, Xing Yi Ling, Stefan A. Maier, Thomas Mayerhöfer, Martin Moskovits, Kei Murakoshi, Jwa-Min Nam, Shuming Nie, Yukihiro Ozaki, Isabel Pastoriza-Santos, Jorge Perez-Juste, Juergen Popp, Annemarie Pucci, Stephanie Reich, Bin Ren, George C. Schatz, Timur Shegai, Sebastian Schlücker, Li-Lin Tay, K. George Thomas, Zhong-Qun Tian, Richard P. Van Duyne, Tuan Vo-Dinh, Yue Wang, Katherine A. Willets, Chuanlai Xu, Hongxing Xu, Yikai Xu, Yuko S. Yamamoto, Bing Zhao, and Luis M. Liz-Marzán. Present and future of surface-enhanced raman scattering. *ACS Nano*, 14(1):28–117, 2020.

[81] M. Grant Albrecht and J. Alan Creighton. Anomalously intense Raman spectra of pyridine at a silver electrode. *Journal of the American Chemical Society*, 99(15):5215–5217, June 1977. Publisher: American Chemical Society.

[82] Jayeong Kim, Yujin Jang, Nam-Jung Kim, Heehun Kim, Gyu-Chul Yi, Yukyung Shin, Myung Hwa Kim, and Seokhyun Yoon. Study of chemical enhancement mechanism in non-plasmonic surface enhanced raman spectroscopy (sers). *Frontiers in Chemistry*, 7, 2019.

[83] Judith Langer, Dorleta Jimenez de Aberasturi, Javier Aizpurua, Ramon A. Alvarez-Puebla, Baptiste Auguié, Jeremy J. Baumberg, Guillermo C. Bazan, Steven E. J. Bell, Anja Boisen, Alexandre G. Brolo, Jaebum Choo, Dana Cialla-May, Volker Deckert, Laura Fabris, Karen Faulds,

F. Javier García de Abajo, Royston Goodacre, Duncan Graham, Amanda J. Haes, Christy L. Haynes, Christian Huck, Tamitake Itoh, Mikael Käll, Janina Kneipp, Nicholas A. Kotov, Hua Kuang, Eric C. Le Ru, Hiang Kwee Lee, Jian-Feng Li, Xing Yi Ling, Stefan A. Maier, Thomas Mayerhöfer, Martin Moskovits, Kei Murakoshi, Jwa-Min Nam, Shuming Nie, Yukihiro Ozaki, Isabel Pastoriza-Santos, Jorge Perez-Juste, Juergen Popp, Annemarie Pucci, Stephanie Reich, Bin Ren, George C. Schatz, Timur Shegai, Sebastian Schlücker, Li-Lin Tay, K. George Thomas, Zhong-Qun Tian, Richard P. Van Duyne, Tuan Vo-Dinh, Yue Wang, Katherine A. Willets, Chuanlai Xu, Hongxing Xu, Yikai Xu, Yuko S. Yamamoto, Bing Zhao, and Luis M. Liz-Marzán. Present and Future of Surface-Enhanced Raman Scattering. *ACS Nano*, 14(1):28–117, January 2020. Publisher: American Chemical Society.

[84] Nariman Banaei, Anne Foley, Jean Marie Houghton, Yubing Sun, and Byung Kim. Multiplex detection of pancreatic cancer biomarkers using a SERS-based immunoassay. *Nanotechnology*, 28(45):455101, October 2017. Publisher: IOP Publishing.

[85] Nariman Banaei, Javad Moshfegh, Arman Mohseni-Kabir, Jean Marie Houghton, Yubing Sun, and Byung Kim. Machine learning algorithms enhance the specificity of cancer biomarker detection using SERS-based immunoassays in microfluidic chips. *RSC Advances*, 9(4):1859–1868, January 2019. Publisher: The Royal Society of Chemistry.

[86] Duo Lin, Shangyuan Feng, Hao Huang, Weisheng Chen, Hong Shi, Nenrong Liu, Long Chen, Weiwei Chen, Yun Yu, and Rong Chen. Label-Free Detection of Blood Plasma Using Silver Nanoparticle Based Surface-Enhanced Raman Spectroscopy for Esophageal Cancer Screening. *Journal of Biomedical Nanotechnology*, 10(3):478–484, March 2014.

[87] Damien Thompson, Jianhui Liao, Michael Nolan, Aidan Quinn, Christian Nijhuis, Colm O'Dwyer, Peter Nirmalraj, Christian Schoenenberger, and Michel Calame. Formation mechanism of metal-molecule-metal junctions: Molecule-assisted migration on metal defects. *The Journal of Physical Chemistry C*, 119, 08 2015.

[88] Federico Raffone, Francesca Risplendi, and Giancarlo Cicero. A New Theoretical Insight Into ZnO NWs Memristive Behavior. *Nano Letters*, 16(4):2543–2547, April 2016.

[89] Siyuan Lyu, Yuan Zhang, Yao Zhang, Kainan Chang, Guangchao Zheng, and Luxia Wang. Picocavity-Controlled Subnanometer-Resolved Single-Molecule Fluorescence Imaging and Mollow Triplets. *The Journal of Physical Chemistry C*, 126(27):11129–11137, July 2022.

[90] Ioana Fechete, Ye Wang, and Jacques C. Védrine. The past, present and future of heterogeneous catalysis. *Catalysis Today*, 189(1):2–27, 2012. Catalytic Materials for Energy: Past, Present and Future.

[91] Martin Schmal. *Heterogeneous Catalysis and its Industrial Applications*. Springer International Publishing, Cham, 2016.

[92] Xiao-Feng Yang, Aiqin Wang, Botao Qiao, Jun Li, Jingyue Liu, and Tao Zhang. Single-atom catalysts: A new frontier in heterogeneous catalysis. *Accounts of Chemical Research*, 46(8):1740–1748, 2013.

[93] Tatyana Tabakova. Recent advances in design of gold-based catalysts for h2 clean-up reactions. *Frontiers in Chemistry*, 7, 2019.

[94] Seoin Back, Min Sun Yeom, and Yousung Jung. Active sites of au and ag nanoparticle catalysts for co2 electroreduction to co. *ACS Catalysis*, 5(9):5089–5096, 2015.

[95] Juha Kostamovaara, Jussi Tenhunen, Martin Kögler, Ilkka Nissinen, Jan Nissinen, and Pekka Keränen. Fluorescence suppression in raman spectroscopy using a time-gated cmos spad. *Opt. Express*, 21(25):31632–31645, Dec 2013.

[96] Thomas Bocklitz, Angela Walter, Katharina Hartmann, Petra Rösch, and Jürgen Popp. How to pre-process Raman spectra for reliable and stable models? *Analytica Chimica Acta*, 704(1):47–56, October 2011.

[97] Kyriakos Kachrimanis, Doris E. Braun, and Ulrich J. Griesser. Quantitative analysis of paracetamol polymorphs in powder mixtures by FT-Raman spectroscopy and PLS regression. *Journal of Pharmaceutical and Biomedical Analysis*, 43(2):407–412, January 2007.

[98] Seng Khoon Teh, Wei Zheng, Khek Yu Ho, Ming Teh, Khay Guan Yeoh, and Zhiwei Huang. Diagnosis of gastric cancer using near-infrared Raman spectroscopy and classification and regression tree techniques. *Journal of Biomedical Optics*, 13(3):034013, May 2008. Publisher: SPIE.

[99] Lei Zhang, Qingqing Li, Wei Tao, Bohao Yu, and Yiping Du. Quantitative analysis of thymine with surface-enhanced Raman spectroscopy and partial least squares (PLS) regression. *Analytical and Bioanalytical Chemistry*, 398(4):1827–1832, October 2010.

[100] A. Ya. Chervonenkis V. N. Vapnik. A class of algorithms for pattern recognition learning, Avtomat. i Telemekh., 1964.

[101] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, page 144–152, New York, NY, USA, 1992. Association for Computing Machinery.

[102] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, September 1995.

[103] Petra Rösch, Michaela Harz, Michael Schmitt, Klaus-Dieter Peschke, Olaf Ronneberger, Hans Burkhardt, Hans-Walter Motzkus, Markus Lankers, Stefan Hofer, Hans Thiele, and Jürgen Popp. Chemotaxonomic identification of single bacteria by micro-raman spectroscopy: Application to clean-room-relevant biological contaminations. *Applied and Environmental Microbiology*, 71(3):1626–1637, 2005.

[104] Shaoxin Li, Yanjiao Zhang, Junfa Xu, Linfang Li, Qiuyao Zeng, Lin Lin, Zhouyi Guo, Zhiming Liu, Honglian Xiong, and Songhao Liu. Noninvasive prostate cancer screening based on serum surface-enhanced Raman spectroscopy and support vector machine. *Applied Physics Letters*, 105(9):091104, 09 2014.

[105] Saranjam Khan, Rahat Ullah, Asifullah Khan, Noorul Wahab, Muhammad Bilal, and Mushtaq Ahmed. Analysis of dengue infection based on raman spectroscopy and support vector machine (svm). *Biomed. Opt. Express*, 7(6):2249–2256, Jun 2016.

[106] Yun Yu, Yating Lin, Chaoxian Xu, Kecan Lin, Qing Ye, Xiaoyan Wang, Shusen Xie, Rong Chen, and Juqiang Lin. Label-free detection of nasopharyngeal and liver cancer using surface-enhanced raman spectroscopy and partial lease squares combined with support vector machine. *Biomed. Opt. Express*, 9(12):6053–6066, Dec 2018.

[107] Roman M. Balabin and Ekaterina I. Lomakina. Support vector machine regression (svr/ls-svm)—an alternative to neural networks (ann) for analytical chemistry? comparison of nonlinear methods on near infrared (nir) spectroscopy data. *Analyst*, 136:1703–1712, 2011.

[108] Tin Kam Ho. Random decision forests. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, volume 1, pages 278–282 vol.1, 1995.

[109] William A. Belson. Matching and Prediction on the Principle of Biological Classification. *Journal of the Royal Statistical Society Series C*, 8(2):65–75, June 1959.

[110] Leo Breiman. *Classification and Regression Trees*. Routledge, New York, October 2017.

[111] Leo Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, August 1996.

[112] Leo Breiman. Random Forests. *Machine Learning*, 45(1):5–32, October 2001.

[113] Wei Hu, Sheng Ye, Yujin Zhang, Tianduo Li, Guozhen Zhang, Yi Luo, Shaul Mukamel, and Jun Jiang. Machine Learning Protocol for Surface-Enhanced Raman Spectroscopy. *The Journal of Physical Chemistry Letters*, 10(20):6026–6031, October 2019. Publisher: American Chemical Society.

[114] Arslan Amjad, Rahat Ullah, Saranjam Khan, Muhammad Bilal, and Asifullah Khan. Raman spectroscopy based analysis of milk using random forest classification. *Vibrational Spectroscopy*, 99:124–129, November 2018.

[115] Carlos Diego L. Albuquerque, Rodrigo B. Nogueira, and Ronei J. Poppi. Determination of 17β-estradiol and noradrenaline in dog serum using surface-enhanced Raman spectroscopy and random Forest. *Microchemical Journal*, 128:95–101, September 2016.

[116] Marta Anghelone, Dubravka Jembrih-Simbürger, and Manfred Schreiner. Identification of copper phthalocyanine blue polymorphs in unaged and aged paint systems by means of micro-Raman spectroscopy and Random Forest. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 149:419–425, October 2015.

[117] Judea Pearl. *Bayesian Networks: A Model of Self-activated Memory for Evidential Reasoning*. UCLA Computer Science Department, 1985. Google-Books-ID: 1sfMOgAACAAJ.

[118] Stephen M. Stigler. *The History of Statistics: The Measurement of Uncertainty before 1900*. Belknap Press, Cambridge, MA, January 1990.

[119] Elon Correa and Royston Goodacre. A genetic algorithm-Bayesian network approach for the analysis of metabolomics and spectroscopic data: application to the rapid identification of Bacillus spores and classification of Bacillus species. *BMC Bioinformatics*, 12(1):33, January 2011.

[120] Shaolong Feng, Tyson P. Eucker, Mayumi K. Holly, Michael E. Konkel, Xiaonan Lu, and Shuo Wang. Investigating the Responses of Cronobacter sakazakii to Garlic-Drived Organosulfur Compounds: a Systematic Study of Pathogenic-Bacterium Injury by Use of High-Throughput Whole-Transcriptome Sequencing and Confocal Micro-Raman Spectroscopy. *Applied and Environmental Microbiology*, 80(3):959–971, February 2014. Publisher: American Society for Microbiology.

[121] Matthew Moores, Kirsten Gracie, Jake Carson, Karen Faulds, Duncan Graham, and Mark Girolami. Bayesian modelling and quantification of raman spectroscopy, 2018.

[122] Mingjun Zhong, Mark Girolami, Karen Faulds, and Duncan Graham. Bayesian Methods to Detect Dye-Labelled DNA Oligonucleotides in Multiplexed Raman Spectra. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 60(2):187–206, 01 2011.

[123] Hua Zheng, Wei Xie, Ilya O. Ryzhov, and Dongming Xie. Policy Optimization in Dynamic Bayesian Network Hybrid Models of Biomanufacturing Processes. *INFORMS Journal on Computing*, 35(1):66–82, 2023. Publisher: INFORMS.

[124] Andy S. Anker, Keith T. Butler, Raghavendra Selvan, and Kirsten M. Ø. Jensen. Machine learning for analysis of experimental scattering and spectroscopy data in materials chemistry. *Chemical Science*, 14(48):14003–14019, 2023.

[125] Juergen Schmidhuber. Annotated history of modern ai and deep learning, 2022.

[126] S.-I. Amari. Learning patterns and pattern sequences by self-organizing nets of threshold elements. *IEEE Transactions on Computers*, C-21(11):1197–1206, 1972.

[127] Seppo Linnainmaa. Taylor expansion of the accumulated rounding error. *BIT Numerical Mathematics*, 16(2):146–160, June 1976.

[128] Paul J. Werbos. Applications of advances in nonlinear sensitivity analysis. In R. F. Drenick and F. Kozin, editors, *System Modeling and Optimization*, pages 762–770, Berlin, Heidelberg, 1982. Springer Berlin Heidelberg.

[129] Félix Lussier, Vincent Thibault, Benjamin Charron, Gregory Q. Wallace, and Jean-Francois Masson. Deep learning and artificial intelligence methods for raman and surface-enhanced raman scattering. *TrAC Trends in Analytical Chemistry*, 124:115796, 2020.

[130] Ying Liu, Belle R. Upadhyaya, and Masoud Naghedolfeizi. Chemometric data analysis using artificial neural networks. *Applied Spectroscopy*, 47(1):12–23, 1993.

[131] I.R. Lewis, N.W. Daniel, N.C. Chaffin, and P.R. Griffiths. Raman spectrometry and neural networks for the classification of wood types—1. *Spectrochimica Acta Part A: Molecular Spectroscopy*, 50(11):1943–1958, 1994.

[132] Monika Gniadecka, Hans Christian Wulf, N N Mortensen, Ole Faurskov Nielsen, and Daniel Højgaard Christensen. Diagnosis of basal cell carcinoma by raman spectroscopy. *Journal of Raman Spectroscopy*, 28:125–129, 1997.

[133] Jinchao Liu, Margarita Osadchy, Lorna Ashton, Michael Foster, Christopher J. Solomon, and Stuart J. Gibson. Deep convolutional neural networks for Raman spectrum recognition: a unified solution. *Analyst*, 142(21):4067–4074, October 2017.

[134] Andrea Angulo, Lankun Yang, Eray S. Aydil, and Miguel A. Modestino. Machine learning enhanced spectroscopic analysis: towards autonomous chemical mixture characterization for rapid process optimization. *Digital Discovery*, 1(1):35–44, 2022.

[135] Jinchao Liu, Stuart J. Gibson, James Mills, and Margarita Osadchy. Dynamic spectrum matching with one-shot learning. *Chemometrics and Intelligent Laboratory Systems*, 184:175–181, January 2019.

[136] Aoune Barhoumi, Dongmao Zhang, Felicia Tam, and Naomi J. Halas. Surface-Enhanced Raman Spectroscopy of DNA. *Journal of the American Chemical Society*, 130(16):5523–5529, April 2008.

[137] Jian-An Huang, Mansoureh Z. Mousavi, Yingqi Zhao, Aliaksandr Hubarevich, Fatima Omeis, Giorgia Giovannini, Moritz Schütte, Denis Garoli, and Francesco De Angelis. SERS discrimination of single DNA bases in single oligonucleotides by electro-plasmonic trapping. *Nature Communications*, 10(1):5321, November 2019.

[138] Edinburgh Instruments Ltd. How to choose your lasers for raman spectroscopy, 2022.

[139] Muddasir Naeem, Noor-ul-ain Fatima, Mukhtar Hussain, Tayyab Imran, and Arshad Saleem Bhatti. Design Simulation of Czerny–Turner Configuration-Based Raman Spectrometer Using Physical Optics Propagation Algorithm. *Optics*, 3(1):1–7, March 2022. Number: 1 Publisher: Multidisciplinary Digital Publishing Institute.

[140] J. Harlander, R. J. Reynolds, and F. L. Roesler. Spatial Heterodyne Spectroscopy for the Exploration of Diffuse Interstellar Emission Lines at Far-Ultraviolet Wavelengths. *The Astrophysical Journal*, 396, September 1992.

[141] Stavros G. Demos, Rajesh N. Raman, Steven T. Yang, Raluca A. Negres, Kathleen I. Schaffers, and Mark A. Henesian. Measurement of the raman scattering cross section of the breathing mode in kdp and dkdp crystals. *Opt. Express*, 19(21):21050–21059, Oct 2011.

[142] Shuming Nie and Steven R. Emory. Probing single molecules and single nanoparticles by surface-enhanced raman scattering. *Science*, 275(5303):1102–1106, 1997.

[143] Yuanhui Zheng, Lorenzo Rosa, Thibaut Thai, Soon Hock Ng, Saulius Juodkazis, and Udo Bach. Phase controlled sers enhancement. *Scientific Reports*, 9, 01 2019.

[144] Felix Benz, Rohit Chikkaraddy, Andrew Salmon, Hamid Ohadi, Bart de Nijs, Jan Mertens, Cloudy Carnegie, Richard W. Bowman, and Jeremy J. Baumberg. Sers of individual nanoparticles on a mirror: Size does matter, but so does shape. *The Journal of Physical Chemistry Letters*, 7(12):2264–2269, 2016. PMID: 27223478.

[145] Cloudy Carnegie, Jack Griffiths, Bart de Nijs, Charlie Readman, Rohit Chikkaraddy, William M. Deacon, Yao Zhang, István Szabó, Edina Rosta, Javier Aizpurua, and Jeremy J. Baumberg. Room-temperature optical picocavities below 1 nm3 accessing single-atom geometries. *The Journal of Physical Chemistry Letters*, 9(24):7146–7151, 2018.

[146] Mattin Urbieta, Marc Barbry, Yao Zhang, Peter Koval, Daniel Sánchez-Portal, Nerea Zabala, and Javier Aizpurua. Atomic-Scale Lightning Rod Effect in Plasmonic Picocavities: A Classical View to a Quantum Effect. *ACS Nano*, 12(1):585–595, January 2018.

[147] Marie Richard-Lacroix and Volker Deckert. Direct molecular-level near-field plasmon and temperature assessment in a single plasmonic hotspot. *Light: Science & Applications*, 9(1):35, March 2020.

[148] Hyun-Hang Shin, Gyu Yeon, Han-Kyu Choi, Sang-Min Park, Kang Lee, and Zee Kim. Frequency-domain proof of the existence of atomic-scale sers hot-spots. *Nano Letters*, 18, 12 2017.

[149] Qianqi Lin, Shu Hu, Tamás Földes, Junyang Huang, Demelza Wright, Jack Griffiths, Eoin Elliott, Bart de Nijs, Edina Rosta, and Jeremy J. Baumberg. Optical suppression of energy barriers in single molecule-metal binding. *Science Advances*, 8(25):eabp9285, June 2022.

[150] OpenAI, Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Dębiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, Rafal Józefowicz, Scott Gray, Catherine Olsson, Jakub Pachocki, Michael Petrov, Henrique P. d O. Pinto, Jonathan Raiman, Tim Salimans, Jeremy Schlatter, Jonas Schneider, Szymon Sidor, Ilya Sutskever, Jie Tang, Filip Wolski, and Susan Zhang. Dota 2 with Large Scale Deep Reinforcement Learning, dec 2019. arXiv:1912.06680 [cs, stat].

[151] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In Yee Whye Teh and Mike Titterington, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR.

[152] Fei-Fei Li, Jiajun Wu, and Ruohan Gao. Cs231n convolutional neural networks for visual recognition, 2022.

[153] Coenraad Mouton, Johannes C. Myburgh, and Marelie H. Davel. Stride and translation invariance in cnns. In Aurona Gerber, editor, *Artificial Intelligence Research*, pages 267–281, Cham, 2020. Springer International Publishing.

[154] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014.

[155] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

[156] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In Sanjoy Dasgupta and David McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 1139–1147, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR.

[157] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural networks for machine learning, 2012.

[158] Alex Graves. Generating sequences with recurrent neural networks, 2013.

[159] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

[160] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In Geoffrey Gordon, David Dunson, and Miroslav Dudík, editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 315–323, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR.

[161] Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *in ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, 2013.

[162] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. Empirical evaluation of rectified activations in convolutional network. *arXiv*, 05 2015.

[163] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 448–456. JMLR.org, 2015.

[164] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[165] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization, 2016.

[166] Ping Luo, Xinjiang Wang, Wenqi Shao, and Zhanglin Peng. Towards understanding regularization in batch normalization. In *International Conference on Learning Representations*, 2019.

[167] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[168] Yuxin Wu and Kaiming He. Group normalization, 2018.

[169] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization, 2016.

[170] Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. *CoRR*, abs/1701.02096, 2017.

[171] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. A neural algorithm of artistic style, 2015.

[172] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

[173] Simone Scardapane, Danilo Comminiello, Amir Hussain, and Aurelio Uncini. Group sparse regularization for deep neural networks. *Neurocomputing*, 241:81–89, 2017.

[174] Yuqiang Fang, Ruili Wang, Bin Dai, and Xindong Wu. Graph-based learning via auto-grouped sparse regularization and kernelized extension. *IEEE Transactions on Knowledge and Data Engineering*, 27(1):142–154, 2015.

[175] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[176] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, 2015.

[177] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks, 2012.

[178] Shanshan Qin, Johannes Will, Hyesung Kim, Nikita Denisov, Simon Carl, Erdmann Spiecker, and Patrik Schmuki. Single atoms in photocatalysis: Low loading is good enough! *ACS Energy Letters*, 8(2):1209–1214, 2023.

[179] Xuning Li, Xiaofeng Yang, Junming Zhang, Yanqiang Huang, and Bin Liu. In situ/operando techniques for characterization of single-atom catalysts. *ACS Catalysis*, 9(3):2521–2531, 2019.

[180] Hongyu An, Longfei Wu, Laurens D. B. Mandemaker, Shuang Yang, Jim de Ruiter, Jochem H. J. Wijten, Joris C. L. Janssens, Thomas Hartman, Ward van der Stam, and Bert M. Weckhuysen. Sub-second time-resolved surface-enhanced raman spectroscopy reveals dynamic co intermediates during electrochemical co2 reduction on copper. *Angewandte Chemie International Edition*, 60(30):16576–16584, 2021.

[181] Resonance Structures, nov 6 2021. [Online; accessed 2022-02-23].

[182] Catherine E Housecroft and Catherine E Housecroft. *Solutions manual [for] Inorganic chemistry, 4th ed.* Pearson, 2012.

[183] E. Smith and G. Dent. *Surface-Enhanced Raman Scattering and Surface-Enhanced Resonance Raman Scattering*, chapter 5, pages 113–133. John Wiley & Sons, Ltd, 2004.

[184] Marina Gühlke, Zsuzsanna Heiner, and Janina Kneipp. Combined near-infrared excited sehrs and sers spectra of ph sensors using silver nanostructures. *Phys. Chem. Chem. Phys.*, 17:26093–26100, 2015.

[185] Cloudy Carnegie, Mattin Urbieta, Rohit Chikkaraddy, Bart de Nijs, Jack Griffiths, William Deacon, Marlous Kamp, N. Zabala, Javier Aizpurua, and Jeremy Baumberg. Flickering nanometre-scale disorder in a crystal lattice tracked by plasmonic flare light emission. *Nature Communications*, 11:682, 02 2020.

[186] Jeremy Baumberg, Javier Aizpurua, Maiken Mikkelsen, and D. Smith. Extreme nanophotonics from ultrathin metallic gaps. *Nature Materials*, 18, 04 2019.

[187] Yao Zhang, Zhen-Chao Dong, and Javier Aizpurua. Theoretical treatment of single-molecule scanning raman picoscopy in strongly inhomogeneous near fields. *Journal of Raman Spectroscopy*, 52(2):296–309, 2021.

[188] Lukas A. Jakob, William M. Deacon, Yuan Zhang, Bart de Nijs, Elena Pavlenko, Shu Hu, Cloudy Carnegie, Tomas Neuman, Ruben Esteban, Javier Aizpurua, and Jeremy J. Baumberg. Softening molecular bonds through the giant optomechanical spring effect in plasmonic nanocavities, 2022.

[189] Graham J. Hutchings. Heterogeneous gold catalysis. *ACS Central Science*, 4(9):1095–1101, 2018. PMID: 30276242.

[190] Masatake Haruta. Size- and support-dependency in the catalysis of gold. *Catalysis Today*, 36(1):153–166, 1997. Copper, Silver and Gold in Catalysis.

[191] Jonas H. K. Pfisterer, Yunchang Liang, Oliver Schneider, and Aliaksandr S. Bandarenka. Direct instrumental identification of catalytically active surface sites. *Nature*, 549(7670):74–77, September 2017.

[192] Junyang Huang, David-Benjamin Grys, Jack Griffiths, Bart de Nijs, Marlous Kamp, Qianqi Lin, and Jeremy J. Baumberg. Tracking interfacial single-molecule ph and binding dynamics via vibrational spectroscopy. *Science Advances*, 7(23):eabg1790, 2021.

[193] Tong Wu, Wei Yan, and Philippe Lalanne. Bright plasmons with cubic nanometer mode volumes through mode hybridization. *ACS Photonics*, 8(1):307–314, 2021.

[194] Won-Hwa Park and Zee Hwan Kim. Charge transfer enhancement in the sers of a single molecule. *Nano letters*, 10(10):4040—4048, October 2010.

[195] Arthur G. Anderson and Bernard M. Steckler. Azulene. viii. a study of the visible absorption spectra and dipole moments of some 1- and 1,3-substituted azulenes1,2. *Journal of the American Chemical Society*, 81(18):4941–4946, 1959.

[196] D. Wright, Q. Lin, D. Berta, et al. Mechanistic study of an immobilized molecular electrocatalyst by in situ gap-plasmon-assisted spectro-electrochemistry. *Nat Catal*, 4:157–164, 2021.

[197] R. C. Maher, L. F. Cohen, J. C. Gallop, E. C. Le Ru, and P. G. Etchegoin. Temperature-dependent anti-stokes/stokes ratios under surface-enhanced raman scattering conditions. *The Journal of Physical Chemistry B*, 110(13):6797–6803, 2006. PMID: 16570987.

[198] Ronen Basri, Meirav Galun, Amnon Geifman, David Jacobs, Yoni Kasten, and Shira Kritchman. Frequency bias in neural networks for input of non-uniform density, 2020.

[199] Guozhong An. The effects of adding noise during backpropagation training on a generalization performance. *Neural Computation*, 8(3):643–674, 1996.

[200] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[201] Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[202] Stefan Grimme, Stephan Ehrlich, and Lars Goerigk. Effect of the damping function in dispersion corrected density functional theory. *Journal of Computational Chemistry*, 32(7):1456–1465, 2011.

[203] Yao Zhang, Zhen-Chao Dong, and Javier Aizpurua. Theoretical treatment of single-molecule scanning raman picoscopy in strongly inhomogeneous near fields. *Journal of Raman Spectroscopy*, 52(2):296–309, 2021.

[204] Andrey Turchanin, Daniel Käfer, Mohamed El-Desawy, Christof Wöll, Gregor Witte, and Armin Gölzhäuser. Molecular mechanisms of electron-induced cross-linking in aromatic sams. *Langmuir*, 25(13):7342–7352, 2009. PMID: 19485375.

[205] Jeremy J. Baumberg. Picocavities: a primer. *Nano Letters*, 22(14):5859–5865, 2022. PMID: 35793541.

[206] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[207] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 2169–2178, 2006.

[208] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. *Spatial Pyramid Matching*, page 401–415. Cambridge University Press, 2009.

[209] S. Hu, Q. Lin, E. S. A. Goerlitzer, B. de Nijs, V. M. Silkin, and J. J. Baumberg. Alchemically-glazed plasmonics using atomic layer metals: controllably synergizing catalysis and plasmonics. [manuscript in preparation].

[210] Lang Xu, Konstantinos G Papanikolaou, Barbara A J Lechner, Lisa Je, Gabor A Somorjai, Miquel Salmeron, and Manos Mavrikakis. Formation of active sites on transition metals through reaction-driven migration of surface atoms. *Science*, 380(6640):70–76, 2023.

[211] Benjamin Lundquist Thomsen, Jesper B. Christensen, Olga Rodenko, Iskander Usenov, Rasmus Birkholm Grønnemose, Thomas Emil Andersen, and Mikael Lassen. Accurate and fast identification of minimally prepared bacteria phenotypes using Raman spectroscopy assisted by machine learning. *Scientific Reports*, 12(1):16436, September 2022. Number: 1 Publisher: Nature Publishing Group.

[212] Zenghui Wang, Jun Zhang, Xiaochu Zhang, Peng Chen, and Bing Wang. Transformer Model for Functional Near-Infrared Spectroscopy Classification. *IEEE Journal of Biomedical and Health Informatics*, 26(6):2559–2569, June 2022. Conference Name: IEEE Journal of Biomedical and Health Informatics.

[213] Frank Nielsen. *Introduction to HPC with MPI for Data Science*. Undergraduate Topics in Computer Science. Springer International Publishing, Cham, 2016.

[214] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 226–231. AAAI Press, 1996.

[215] Federico Calle-Vallejo, Marc Koper, and Aliaksandr Bandarenka. ChemInform Abstract: Tailoring the Catalytic Activity of Electrodes with Monolayer Amounts of Foreign Metals. *Chemical Society reviews*, 42, April 2013.

[216] Marc T. M. Koper. Introductory Lecture. *Faraday Discussions*, 140(0):11–24, October 2008. Publisher: The Royal Society of Chemistry.

[217] Sebastian Schnur and Axel Groß. Challenges in the first-principles description of reactions in electrocatalysis. *Catalysis Today*, 165(1):129–137, May 2011.

[218] Gerhard Ertl. Reactions at Surfaces: From Atoms to Complexity (Nobel Lecture). *Angewandte Chemie International Edition*, 47(19):3524–3535, 2008. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/anie.200800480.

[219] Max Appl. Ammonia. In *Ullmann's Encyclopedia of Industrial Chemistry*. John Wiley & Sons, Ltd, 2006.

[220] Michikazu Hara, Masaaki Kitano, and Hideo Hosono. Ru-Loaded C12A7:e– Electride as a Catalyst for Ammonia Synthesis. *ACS Catalysis*, 7(4):2313–2324, April 2017. Publisher: American Chemical Society.

[221] Masaaki Kitano, Jun Kujirai, Kiya Ogasawara, Satoru Matsuishi, Tomofumi Tada, Hitoshi Abe, Yasuhiro Niwa, and Hideo Hosono. Low-Temperature Synthesis of Perovskite Oxynitride-Hydrides as Ammonia Synthesis Catalysts. *Journal of the American Chemical Society*, 141(51):20344–20353, dec 2019. Publisher: American Chemical Society.

[222] Yang Song, Daniel Johnson, Rui Peng, Dale K. Hensley, Peter V. Bonnesen, Liangbo Liang, Jingsong Huang, Fengchang Yang, Fei Zhang, Rui Qiao, Arthur P. Baddorf, Timothy J. Tschaplinski, Nancy L. Engle, Marta C. Hatzell, Zili Wu, David A. Cullen, Harry M. Meyer, Bobby G. Sumpter, and Adam J. Rondinone. A physical catalyst for the electrolysis of nitrogen to ammonia. *Science Advances*, 4(4):e1700336, 2018.

[223] Yang Song, Rui Peng, Dale K. Hensley, Peter V. Bonnesen, Liangbo Liang, Zili Wu, Harry M. Meyer III, Miaofang Chi, Cheng Ma, Bobby G. Sumpter, and Adam J. Rondinone. High-Selectivity Electrochemical Conversion of CO2 to Ethanol using a Copper Nanoparticle/N-Doped Graphene Electrode. *ChemistrySelect*, 1(19):6055–6061, 2016. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/slct.201601169.

[224] Hefei Li, Tianfu Liu, Pengfei Wei, Long Lin, Dunfeng Gao, Guoxiong Wang, and Xinhe Bao. High-Rate CO2 Electroreduction to C2+ Products over a Copper-Copper Iodide Catalyst. *Angewandte Chemie International Edition*, 60(26):14329–14333, 2021. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/anie.202102657.

[225] Ryo Iwama, Koji Takizawa, Kenichi Shinmei, Eisuke Baba, Noritoshi Yagihashi, and Hiromasa Kaneko. Design and Analysis of Metal Oxides for CO2 Reduction Using Machine Learning, Transfer Learning, and Bayesian Optimization. *ACS Omega*, 7(12):10709–10717, March 2022. Publisher: American Chemical Society.

[226] Chen Jia, Kamran Dastafkan, Wenhao Ren, Wanfeng Yang, and Chuan Zhao. Carbon-based catalysts for electrochemical CO2 reduction. *Sustainable Energy & Fuels*, 3(11):2890–2906, October 2019. Publisher: The Royal Society of Chemistry.

[227] Bart de Nijs, Felix Benz, Steven J. Barrow, Daniel O. Sigle, Rohit Chikkaraddy, Aniello Palma, Cloudy Carnegie, Marlous Kamp, Ravishankar Sundararaman, Prineha Narang, Oren A. Scherman, and Jeremy J. Baumberg. Plasmonic tunnel junctions for single-molecule redox chemistry.

Nature Communications, 8(1):994, October 2017. Number: 1 Publisher: Nature Publishing Group.

[228] C. A. Glasbey and K. V. Mardia. A review of image-warping methods. *Journal of Applied Statistics*, 25(2):155–171, 1998.

[229] D. T. Lee and B. J. Schachter. Two algorithms for constructing a Delaunay triangulation. *International Journal of Computer & Information Sciences*, 9(3):219–242, June 1980.

[230] Charles L. Lawson. Transforming triangulations. *Discrete Mathematics*, 3(4):365–372, January 1972.

[231] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, March 2010. ISSN: 1938-7228.

[232] Lei Huang, Lei Zhao, Yi Zhou, Fan Zhu, Li Liu, and Ling Shao. An investigation into the stochasticity of batch whitening, 2020.

[233] J. K. Nørskov, T. Bligaard, J. Rossmeisl, and C. H. Christensen. Towards the computational design of solid catalysts. *Nature Chemistry*, 1(1):37–46, April 2009.

[234] Compound summary: Tributyl phosphate. https://pubchem.ncbi.nlm.nih.gov/compound/Tributyl-phosphate. Accessed: 11-09-2023.

[235] M. J. Foster, J. Storey, and M. A. Zentile. Spatial-heterodyne spectrometer for transmission-Raman observations. *Optics Express*, 25(2):1598, jan 2017.

[236] David J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Copyright Cambridge University Press, 2003.

[237] Liangrui Pan, Pronthep Pipitsunthonsan, Chalongrat Daengngam, and Mitchai Chongcheawchamnan. Identification of complex mixtures for raman spectroscopy using a novel scheme based on a new multi-label deep neural network, 2020.

[238] William John Thrift and Regina Ragan. Quantification of Analyte Concentration in the Single Molecule Regime Using Convolutional Neural Networks. *Analytical Chemistry*, 91(21):13337–13342, November 2019. Publisher: American Chemical Society.

[239] Bulent Ayhan and Chiman Kwan. New Results on Radioactive Mixture Identification and Relative Count Contribution Estimation. *Sensors*, 21(12):4155, January 2021. Number: 12 Publisher: Multidisciplinary Digital Publishing Institute.

[240] Desimoni E Brunetti B. About Estimating the Limit of Detection by the Signal to Noise Approach. *Pharmaceutica Analytica Acta*, 06(04), 2015.

[241] Ian T. Jolliffe. A note on the use of principal components in regression. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 31(3):300–303, 1982.

[242] Thomas J. Seymour, Patricia Crawford, and Dawn M. Ecker. MAb Products: Market Trends and Projections, October 2020.

[243] Arpit Arunkumar Bana, Nithin Sajeev, Sabyasachi Halder, Haidar Abbas Masi, Shikha Patel, and Priti Mehta. Comparative stability study and aggregate analysis of Bevacizumab marketed formulations using advanced analytical techniques. *Heliyon*, 9(9):e19478, September 2023.

[244] Karolina L. Zapadka, Frederik J. Becher, A. L. Gomes dos Santos, and Sophie E. Jackson. Factors affecting the physical stability (aggregation) of peptide therapeutics. *Interface Focus*, 7(6):20170030, October 2017. Publisher: Royal Society.

[245] Arpit Arun K. Bana, Priti Mehta, and Khushboo Ashok Kumar Ramnani. Physical Instabilities of Therapeutic Monoclonal Antibodies: A Critical Review. *Current Drug Discovery Technologies*, 19(6):1–11, 2022.

[246] Michael Foster, William Brooks, Philipp Jahn, Jesper Hedberg, Andreas Andersson, and And Lorna Ashton. Demonstration of a compact deep UV Raman spatial heterodyne spectrometer for biologics analysis. *Journal of Biophotonics*, 15(7):e202200021, July 2022.

[247] Lorna Ashton and Royston Goodacre. APPLICATION OF DEEP UV RESONANCE RAMAN SPECTROSCOPY TO BIOPROCESSING. *European Pharmaceutical Review*, 16(3), 2011.

[248] Ewen Smith and Geoffrey Dent. *Modern Raman Spectroscopy: A Practical Approach*. John Wiley & Sons, Ltd, August 2005. Journal Abbreviation: Modern Raman Spectroscopy - A Practical Approach Publication Title: Modern Raman Spectroscopy - A Practical Approach.

[249] Jr Charles A Janeway, Paul Travers, Mark Walport, and Mark J. Shlomchik. The structure of a typical antibody molecule. In *Immunobiology: The Immune System in Health and Disease. 5th edition*. Garland Science, 2001.

[250] Compound summary: Tryptophan. https://pubchem.ncbi.nlm.nih.gov/compound/Tryptophan. Accessed: 11-09-2023.

[251] Michael Foster, Michael Wharton, William Brooks, Matthew Goundry, Charles Warren, and Jonathan Storey. Remote sensing of chemical agents within nuclear facilities using raman spectroscopy. *Journal of Raman Spectroscopy*, 51(12):2543–2551, 2020.

[252] S. P. Burton, M. A. Vaughan, R. A. Ferrare, and C. A. Hostetler. Separating mixtures of aerosol types in airborne high spectral resolution lidar data. *Atmospheric Measurement Techniques*, 7(2):419–436, 2014.

[253] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, October 2017. ISSN: 2380-7504.