Digital Discovery

Accepted Manuscript

This article can be cited before page numbers have been issued, to do this please use: A. Sineesh and A. R. Kamsali, *Digital Discovery*, 2026, DOI: 10.1039/D6DD00044D.

ROYAL SOCIETY OF CHEMISTRY

rsc.li/digitaldiscovery

View Article Online
DOI: 10.1039/D6DD00044D

Journal Name

ARTICLE TYPE

Cite this: DOI: 00.0000/xxxxxxxxxx

Benchmarking Deep Learning Models for Raman Spectroscopy Across Open-Source Datasets

Adithya Sineesh*[a] and Akshita Ramya Kamsali [a]

Received Date
Accepted Date

DOI: 00.0000/xxxxxxxxxx

Deep learning classifiers for Raman spectroscopy are increasingly reported to outperform classical chemometric approaches. However, their evaluations are often conducted in isolation or compared against traditional machine learning methods or trivially adapted vision-based architectures that were not originally proposed for Raman spectroscopy. As a result, direct comparisons between existing deep learning models developed specifically for Raman spectral analysis on shared open-source datasets remain scarce. In this work, we focus on supervised Raman spectra classification where each spectrum is assigned to a predefined material, bacterial/yeast isolate, drug treatment or pharmaceutical compound. To the best of our knowledge, this study presents one of the first benchmarks comparing three or more published Raman-specific deep learning classifiers across multiple opensource Raman datasets. We evaluate five representative Deep Learning (DL) architectures along with two conventional Machine Learning (ML) methods under a unified training and hyperparameter tuning protocol across three open-source Raman datasets selected to support standard evaluation, fine-tuning, and explicit distribution-shift testing. In this comparative study, we primarily focus on classification because the selected open-source datasets provide classification annotations, while annotations for complete structure elucidation are not available. We report classification accuracies and macro-averaged F1 scores to provide a fair and reproducible comparison of the supervised ML and DL models for Raman spectra based classification.

1 Introduction

Raman spectroscopy is a non-destructive characterization technique, where the spectra encode vibrational signatures of molecular bonds. This non-destructive capability enables applications spanning biomedical diagnostics[1], pharmaceuticals[2], materials identification[3] and food quality assessment[4]. Despite this breadth, Raman spectra are a weak scattering phenomenon, which leads to captured spectra often containing artifacts such as distortion by noise, background fluorescence, cosmic ray impacts and other environmental factors. These artifacts can dominate the class distinguishing features in the signal which complicates statistical learning for material identification[5]. Another challenge for automated identification of materials is the spectral overlap of the characteristic peaks of different components in the substance. Beyond additive noise, Raman spectra vary across systems and their configurations leading to significant distributional shifts that undermine the performance guarantees established on the test datasets when the models are deployed in the real world[6]. At the same time, constructing large, labeled Raman spectroscopy datasets is costly as hardware is expensive and careful sample preparation, calibration and repeated measurements are labor intensive. Furthermore, annotation and assigning peaks to specific bonds or chemical groups often requires domain expertise which is often not available at scale. Machine Learning has therefore been increasingly adopted for material identification. Early works showcased the effectiveness of traditional chemometric approaches like Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Support Vector Machines (SVMs) in material classification using Raman spectra[7 8]. More recent approaches leverage deep neural models, particularly onedimensional convolutional networks[3 9 10] and attention-based architectures[11 12 13] and often report substantial gains over classical baselines. However, most of the existing works do not compare their proposed methods against other published deep learning approaches for Raman spectra classification. Such comparisons are hindered by challenges in data reproducibility, dataset accessibility, and limited data-sharing frameworks[14].

Benchmarking under a standard evaluation framework allows for fair and meaningful comparisons across different architectures and algorithms. Such evaluations help researchers identify which strategies are effective across different datasets and problem settings. These established reference points can serve as an excellent starting point for advancing state-of-the-art performance. In this work, we benchmark five supervised deep learning models proposed for Raman

[a]Elmore Family School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN-47907, USA
*E-mail: asineesh@purdue.edu

View Article Online
DOI: 10.1039/D6DD00044D

spectra classification, along with two conventional machine learning models, under a controlled and reproducible training and evaluation protocol across three open-source Raman datasets and report their accuracy and macro-averaged F1 score. We focus on classification because it is a widely used and well defined Raman spectroscopy task for material identification. The selected open-source datasets provide labels that directly support this task. Other Raman analysis tasks such as mixture analysis and complete structure elucidation require different annotations, which are not available for all the datasets at the time of this study and are therefore beyond the scope of the present benchmark. In subsequent sections, we outline the motivation behind the selection along with the details of the deep learning models and the datasets.

2 Literature Survey

Machine learning techniques have been widely used for multiclass classification, where each spectrum is assigned to exactly one material. It has also been applied to the broader problem of multi-label classification, where the constraint of a sample containing only one target is relaxed. This task involves associating each Raman spectrum with multiple materials simultaneously. Multi-label classification with material concentration estimation further extends this task by estimating the mixing ratios in addition to predicting the components present in the sample. In the following subsections, we review the prior work in all these different types of Raman spectra classification. We organize the literature from foundational statistical approaches to modern deep learning based models.

2.1 Classic Chemometrics

Early Raman-based classification pipelines used linear dimensionality reduction to compress the spectrum and then trained shallow discriminative models with limited number of samples. For example, Rebrošová et al. categorized staphylococci into 16 strains using Principal Component Analysis (PCA) for feature compression followed by 1-NN classification on 277 spectra with 5-fold cross-validation[15]. Similarly, Monavar et al. leveraged PCA-derived representations with shallow Artificial Neural Networks (ANNs) and Linear Discriminant Analysis (LDA) for caviar type identification despite having only 93 spectra[7]. In pharmaceuticals, Roggo et al. used Support Vector Machine (SVM) based hierarchies to first identify tablet product families and then refine predictions to formulation categories via correlation-based methods and additional SVMs, using 25 product families comprising of 44 formulations (15 spectra per formulation)[8] in total. Generally, these classic chemometrics studies used compressed feature spaces coupled with lightweight classifiers. Recent years have seen a gradual shift from such traditional pipelines toward modern deep learning approaches that aim to learn representations directly from raw (or minimally processed) spectra and better tolerate perturbations expected in Raman spectra.

2.2 Deep Learning Models Evaluated on Small Experimental Raman Datasets

Deep Learning approaches for Raman spectroscopy mainly involve developing 1-D Convolutional Neural Networks (CNNs) or transformers and comparing them against classical baselines. An early example is the Deep CNN mineral classifier by Liu et al.[3], which was evaluated against KNN/SVM/Random Forest on RRUFF-derived datasets[16]. While such studies established the potential of CNN-based models for Raman analysis, their evaluations were often conducted on test sets comprising of a limited number of experimentally acquired spectra. For example, Dong et al. designed a CNN with constrained kernels that emulate standard denoising and baseline-correction operations. The model achieved superior binary classification of human vs. animal blood (109 test spectra spanning humans, dogs, and rabbits using a 67:33 train-test split), outperforming SVM and PLS-DA[9]. Similarly, Kirchberger-Tolstik et al. developed a CNN model to predict the severity of Ulcerative Colitis using the Raman spectra of the colon biopsy[10]. The model was assessed using patient level cross-validation on 227 Raman spectra acquired from 42 patients and not compared with any other approaches.

A Locally Connected Neural Network (LCNN) designed by Houston et al.[17] was used to detect chlorinated solvents using the Raman spectra of the samples. Despite outperforming SVM, KNN, Decision Tree, Gaussian Naive Bayes and Fully Connected Neural Network (FCNN), the evaluation was performed on a small real dataset of 58 test spectra. In another example, the classification of three marine pathogen strains was achieved with the help of Generative Adversarial Networks[18] by Yu et al. The independent Generator-Discriminator for each pathogen strain were trained on the corresponding 50 experimental spectra. Testing was conducted on 60 real spectra, achieving a reported 100% classification accuracy, though the method was not compared against competing approaches.

RaMixNet I and II[19] were developed for multi-label material identification and multi-label classification with concentration estimation respectively. These models were trained primarily on synthetic spectra generated by linear combinations of four pure compounds with additional baseline drift and spectral augmentations. Evaluation on real data was restricted to only six experimentally measured mixtures where they outperformed correlation-based methods and Partial Least Squares Regression.

While these works showed the promising performance of deep learning methods, their heavy reliance on small-scaled experimental datasets raises questions about their efficacy during realworld deployment.

2.3 Synthetic Data Driven Evaluation of Raman Classification Models

Subsequent works tried to alleviate the limitation of small-scaled experimental datasets by making extensive use of synthetic data. For example, Qi et al. developed a CNN model to classify 10 different 2-D materials[20] like Graphene using limited number of real Raman spectra along with large quantities of synthetic data generated by Denoising Diffusion Probabilistic Models (DDPMs)[21].

View Article Online
DOI: 10.1039/D6DD00044D

The dataset consisted of 10,000 synthetic spectra and 594 experimental spectra. The authors compared their model to SVM, ANN, KNN methods using a 10 fold cross-validation scheme. Similarly, Hamed Mozaffari and Tay trained a model consisting of a single 1-D convolutional layer with two linear layers[22] on 40,000 synthetic spectra generated by augmenting 5 real Raman spectra of 5 different materials. This model outperformed several other published Raman ML models on this 5-category classification problem. However, the test dataset comprised of 5000 synthetic spectra generated by augmenting the same 5 gold-standard spectra as the train dataset.

DeepCID[23] further exemplifies this evaluation protocol. The framework is comprised of a suite of CNN models that each predict if a specific material is present in the sample or not. The models are trained on synthetically generated spectra that were generated by applying augmentations to 167 real spectra of pure common pharmaceutical raw materials. It outperforms traditional chemometric methods and a FCNN on synthetic data and limited real Raman spectra of mixtures. Overall, synthetic data is often used as a remedy for the scarcity of experimentally acquired Raman spectra. But evaluating models on synthetic test datasets leads to the same limitations of using small real test datasets i.e. uncertainty about their performance on unseen samples during real-world deployment.

2.4 Evaluation Limited to Classical or Generic Deep Learning Baselines

In addition to the limitations due to the use of small experimental or synthetic test datasets, several deep learning models for Raman spectroscopy have been proposed without systematic comparison to existing Raman-specific deep learning approaches. Evaluations are often restricted to classical machine-learning baselines or generic image-based architectures which makes it difficult to assess the merits of the proposed methods within the broader Raman spectroscopy literature. For example, Maruthamuthu et al. used a ResNet-18[24] inspired CNN model for the detection of microbial contamination[25]. The dataset contained 6000 real Raman spectra each of Chinese Hamster Ovary (CHO) cells, which are widely used in the pharmaceutical industry, along with 12 microbes that are the common contaminants of CHO and 3 mixtures of CHO and contaminant microbes. The model was trained using a 5-fold cross-validation scheme but not compared against any other approaches, and the exhaustive dataset is not public.

Similarly, Primrose et al. adapted the VGG13[26] architecture for Raman spectra classification by replacing all the 2-D convolutional layers with 1-D convolutional layers[27]. This model was then trained on a synthetically mixed dataset and was evaluated on limited data collected by the First Defender Raman spectrometer and showed >90% detection rates for certain explosive materials and their precursors, outperforming the algorithm used by the spectrometer. This work was extended[28] to include additional non-explosive materials and the evaluation was performed over 10,000 real Raman spectra but the dataset was not made public and the model was not compared against other published Ramanspecific approaches.

A similar evaluation pattern is observed for Raman Spectral Translation (RST)[12], which was developed for multi-label classification by combining the ideas from CNNs and transformers. However, the comparisons were limited to a generic CNN and DenseNet[29], rather than against established Raman-domain architectures, on a real test dataset. Likewise, a transformer model for classification of deep-sea cold seep bacteria[11] and a CNN model for classification of plastics[30] were compared using experimental test spectra against 1-D variants of AlexNet[31], ResNet and SVM, LDA and Decision Trees respectively.

This trend extends to biomedical and agricultural applications as well. The RFBC[4] model used a hybrid CNN-LSTM architecture incorporating Fourier-domain features and outperformed PCASVM, PCA-KNN and PCA-XGBoost in detecting different brands of rice on a test dataset containing experimental Raman spectra. Similarly, Y. Lin et al. developed a ResNet-18 based model for cancer detection[1] using 2-D transforms of 1-D serum Raman spectra (CWT/heatmaps). The authors evaluated the model only against generic image classifiers like AlexNet, VGG16, DenseNet. Kok et al. designed another ResNet-based model, with multi-channel inputs spanning raw spectra plus multiple pre-processed views,[32] for osteoarthritis cartilage classification and compared the model against baseline CNNs. Similarly, Ullah et al. developed a MultiLayer Perceptron (MLP) to detect tuberculosis from blood serum samples[33] but did not compare the model with any other approaches.

Du et al. developed another shallow CNN to classify Bacillus spores[34] but only compared the model to traditional ML approaches like SVM. RaT and RaST were transformer and SwinTransformer[35] based architectures proposed for classifying lactic acid bacteria into 14 different strains[36]. These models were evaluated on an experimental test dataset against an adapted ResNet model, SVM, LDA, KNN and XGBoost.

Likewise, the evaluation is restricted to MLP, Least squares, modified VGG11 and ResNet-50 for the RamanFormer[13], which was proposed to quantify the presence of Methanol, Isopropyl Alcohol and Ethanolamine in the Raman spectra of the mixture. Another example is ConInceDeep[37], a multi-label classification model that combines Continuous Wavelet Transform (CWT) representations with convolution-based Inception Modules[38]. It was evaluated against ablated variants of its own proposed architecture.

Collectively, these works demonstrate a huge variety of deep learning approaches developed for Raman spectroscopy across different domains. However, the lack of benchmarking against established Raman-specific models hampers the ability to draw meaningful conclusions across multiple papers.

2.5 Limited Benchmarking

Only a few works in Raman deep learning literature attempt explicit benchmarking against previously published models. However, the scope of the comparison is often limited in these cases.

RamanNet was proposed as a general Raman spectra classifier[39] and was evaluated on open-source datasets like the COVID-19 Raman dataset[40], Melanoma dataset[41], RRUFF Mineral database[16] and the

View Article Online
DOI: 10.1039/D6DD00044D

Bacteria-ID dataset[42]. While RamanNet outperformed the baseline for each dataset, the benchmarking was limited as the comparison was just against one model per dataset. A Scale-Adaptive deep neural network (SANet) was designed for identifying the isolate and the empiric treatment for the sample based on its Raman spectrum[43]. The model was trained and evaluated on the Bacteria-ID dataset and compared against the model presented in that work[42] and traditional ML methods like SVM and Linear Regression rather than a broader set of Raman-specific deep learning models.

The Wavelet Packet transform and Gramian Angular field (WPGA) algorithm was developed by Liu et al. to generate 2D spectrograms from 1-D Raman spectral data[44]. The authors then showed that a ResNet based model using these 2-D spectrograms outperformed several published Raman classifier models[42 39 45] on the open-source Bacteria-ID dataset[42]. However, the official implementation of the model is not publicly available, and the paper and its supplementary material do not provide sufficient architectural details to allow for an exact reproduction of the proposed network.

Lange et al. published an open-source dataset containing 6960 Raman spectra of mixtures containing 8 different metabolites in varying concentrations[46]. The authors compared 11 different models on it but only one of those models was published for Raman spectra classification.

Overall, the prior work in Raman spectra classification shows tremendous architectural innovation but suffers from limitations due to: (i) small experimental datasets or synthetically generated test sets, (ii) comparisons restricted to classical machine-learning baselines or trivially adapted vision models. This complicates meaningful cross-paper performance assessment. In this work, we present reproducible benchmarking under consistent experimental settings for five supervised Raman-specific DL models and two conventional ML models across three open-source Raman spectral datasets.

3 Benchmark Models

In the previous section, we highlighted the numerous approaches employed over the years for material classification and material analysis of samples based on their Raman spectrum. In this work, the benchmarking is limited to just supervised deep learning models for multi-class classification. We chose five models, detailed in subsequent sections, which represent a variety of architectural designs, complexities and sizes as shown in Table 1. All these models were trained using Cross-Entropy Loss, unless mentioned otherwise. We also employ Random Forest and Support Vector Classifier (SVC) to investigate any performance differences between these conventional ML models and the five Raman-specific DL models.

3.1 CNN-based models

The Deep CNN model[3] was one of the first Deep Learning models introduced for Raman spectra classification and is based on the famous LeNet-5 Architecture[47]. It consists of three 1-D convolutional layers with kernel sizes of 21, 11 and 5 interleaved with pooling layers that reduce the spatial dimension of the spectra. The features generated by these layers are then passed through a dense linear layer followed by a classification head, which contains the same number of output nodes as the number of classes in the dataset.

Table 1 The five chosen models in terms of parameter count and multiplyaccumulate (MAC) operations for a Raman spectrum of length 1024 with
15 output classes

Model	Number of parameters (M)	MACs (M)
Deep CNN[3]	15.91	21.77
SANet[43]	2.23	102.48
RamanNet[39]	0.72	0.72
Transformer[11]	85.17	769.88
RamanFormer[13]	4.31	24.33

Convolution has the property of shift equivariance, which is desirable for images but detrimental for Raman spectra where the position of the peak plays a vital role in identifying the material. The RamanNet[39] model was developed to be free from this translational equivariance. The spectrum is split into overlapping sliding windows of a fixed size and each window is passed through its own multi-layer perceptron. This operation is similar to convolution but with different kernels applied to different spatial locations. The outputs for all the windows are then concatenated and passed through another multi-layer perceptron and a linear embedding layer to obtain the feature representation of the spectra. These features are then fed into a classification head to obtain the predicted label. RamanNet is trained using a linear combination of triplet loss on the feature representation of the spectra and Cross-Entropy loss on the predicted class labels.

At each convolutional layer in a CNN, the size of the Receptive Field (RF) is constant. This property was said to be undesirable for capturing the peaks of a Raman spectrum, which are of different widths, and thus the Scale Adaptive Network (SANet)[43] was developed. It consists of five Multi-Scale Blocks for feature extraction with each Block consisting of six 1-D convolutional layers having increasing kernel sizes from 3 to 13. This allows for each Block to capture the features at different scales of RF. These features are stacked along the channel dimension followed by channel-attention and point-wise convolution to extract only the relevant features while reducing the channel dimension. The features generated by the sequence of MultiScale Blocks are then flattened and passed through the classification head.

3.2 Transformer-based models

Recent years have seen the proliferation of attention-based mechanisms for vision and language tasks. The transformer[48] architecture, which is solely based on attention, has also been adopted for the classification of Raman spectra. This development was only natural as the attention mechanism of the transformer was originally designed to model sequential data. The earliest such attempt involved simply adapting the Vision Transformer (ViT)[49] for the 1-D spectral

View Article Online
DOI: 10.1039/D6DD00044D

domain[11]. The spectrum is split into patches and then passed through a linear layer to map them to tokens of dimension 768. A learnable class token of the same dimension is appended to this sequence of tokens followed by the addition of Position encoding to track the sequence of tokens. The model consists of 12 transformer blocks, each comprising of a 12-head selfattention layer and a multi-layer perceptron. The former captures the relationships between the different tokens in the sequence and the latter helps to generate higher-level features from each token. At the output of the transformer blocks, the embedding of the class token serves as the aggregate representation of the input sequence. Therefore, this embedding is passed through the classification head. This model shall be referred to as the Transformer in the subsequent sections.

The RamanFormer[13] further modifies the transformer architecture for processing Raman spectra. The generation of tokens from the spectrum remains the same, only without the class token and with a reduced embedding dimension of 256. This sequence of 8 tokens are passed through just three transformer blocks, each containing a 8-head self-attention layer and a MLP. The output token sequence is fed through two strided convolutional layers to capture the spatial hierarchy. The features are then pooled along the temporal dimension followed by passing them through a dense layer and classification head. Although RamanFormer was originally proposed for Mixture analysis of Raman spectra, we use it for classification tasks. The underlying architecture remains the same with Cross-Entropy loss being used instead of the original L1 loss.

4 Datasets

We surveyed publicly available Raman spectroscopy datasets and selected three that span distinct application domains and exhibit varying degrees of distribution shift between the training and test sets. Specifically, we choose the MLROD (minerals)[50], Bacteria-ID[42] (biomedical), and API[51] (pharmaceutical) datasets as shown in Figure 1. The MLROD test set was generated under different conditions than the training set, enabling evaluation under distribution shift. The Bacteria-ID dataset consists of a reference set suitable for pretraining and a smaller fine-tuning set that contains the same degradation in optical system efficiency as the test dataset. Meanwhile the test split of the API dataset is from the same distribution as the train and validation split. However, unlike the previously mentioned datasets, the API dataset does not provide predefined train and test splits and these splits must be constructed by the user. We now discuss the three datasets and their corresponding acquisition method and the criteria we test for.

Machine Learning Raman Open Dataset (MLROD) MLROD is a high-volume Raman dataset created for material detection on Mars. It contains 89,121 labeled spectra spanning 12 pure mineral classes and 3 binary 1:1 powder mixtures. Mineral classes are: Quartz, Albite, Anorthite, Microcline, Hornblende, Biotite, Muscovite, Forsterite, Augite, Enstatite, Calcite, Gypsum; mixtures are Quartz+Albite, Forsterite+Augite, Forsterite+Albite. It also consists of a separate set of test spectra (rocks, with and without dust): 39,720 spectra from Gabbro and Granite slabs, measured under clean (0% dust) and dusty (50% Basaltic dust coverage) conditions (e.g., Gabbro: 8,952 clean / 9,740 dusty; Granite: 11,028 and 10,000 across the two conditions as listed) *. The "dusty" regime was created using basaltic dust to mimic the obscuration in Martian conditions. Figure 2 shows the difference in the Raman spectra collected in the "train", "clean" test and "dusty" test regimes. The dataset was collected on Horiba LabRAM HR Evolution single stage spectrometer with 532 nm excitation with no pre-processing steps and only standardizing the axis via interpolation and trimming.

Bacteria-ID Dataset It is a bacterial (mostly monolayer and single-cell) Raman spectroscopy dataset intended for pathogen identification and downstream grouping by 8 antibiotic treatments. The dataset provides a reference dataset across 30 bacterial plus yeast isolates with 2000 spectra per isolate for training. Additionally, it contains fine-tuning and test sets each with 100 spectra per isolate. The dataset was collected on Horiba LabRAM HR Evolution Raman microscope at 633nm excitation with polynomial background correction and per-spectrum minmax normalization to [0,1] range.

Active Pharmaceutical Ingredients (API) Dataset The API dataset is an open-source dataset with 3510 spectra spanning 32 pure compounds measured on Raman Rxn2 analyzer at 785nm excitation. The dataset provided comes with automatic instrument pre-treatment which includes dark noise subtraction, cosmic ray filtering and intensity correction with no other preprocessing steps.

5 Methodology

In this section, we outline the training methodology and evaluation approach used to benchmark the five Raman-specific deep learning models, along with two conventional machine learning methods, on the three open-source Raman spectroscopy datasets. For consistency, we use the same optimization procedure (Adam), model-selection criterion and stopping rules across all deep learning models, while tuning hyperparameters independently for each model-dataset task pair using an identical search procedure. We also include Random Forest and a Support Vector Classifier (SVC) as conventional machine learning methods to evaluate how their performance compares with that of the five Raman-specific deep learning models.

Table 2 List of hyperparameters evaluated for the grid search, applied independently to each of the five DL models on all three datasets, with separate tuning for every model–dataset pair to ensure fair comparisons

Hyperparameters	Values
Batch Size	32, 128, 512
Learning Rate	1e-3, 1e-4, 1e-5

*When we downloaded the data from the official website, the training dataset had 90,341 spectra. Several of the spectra in the Granite 50% dust test dataset have a label that does not correspond to any of the labels in the training dataset. We have ignored these samples for the purposes of evaluation and therefore the size of the Granite 50% dust test dataset is 5,183 and the size of the overall test dataset is 34,903.

View Article Online
DOI: 10.1039/D6DD00044D

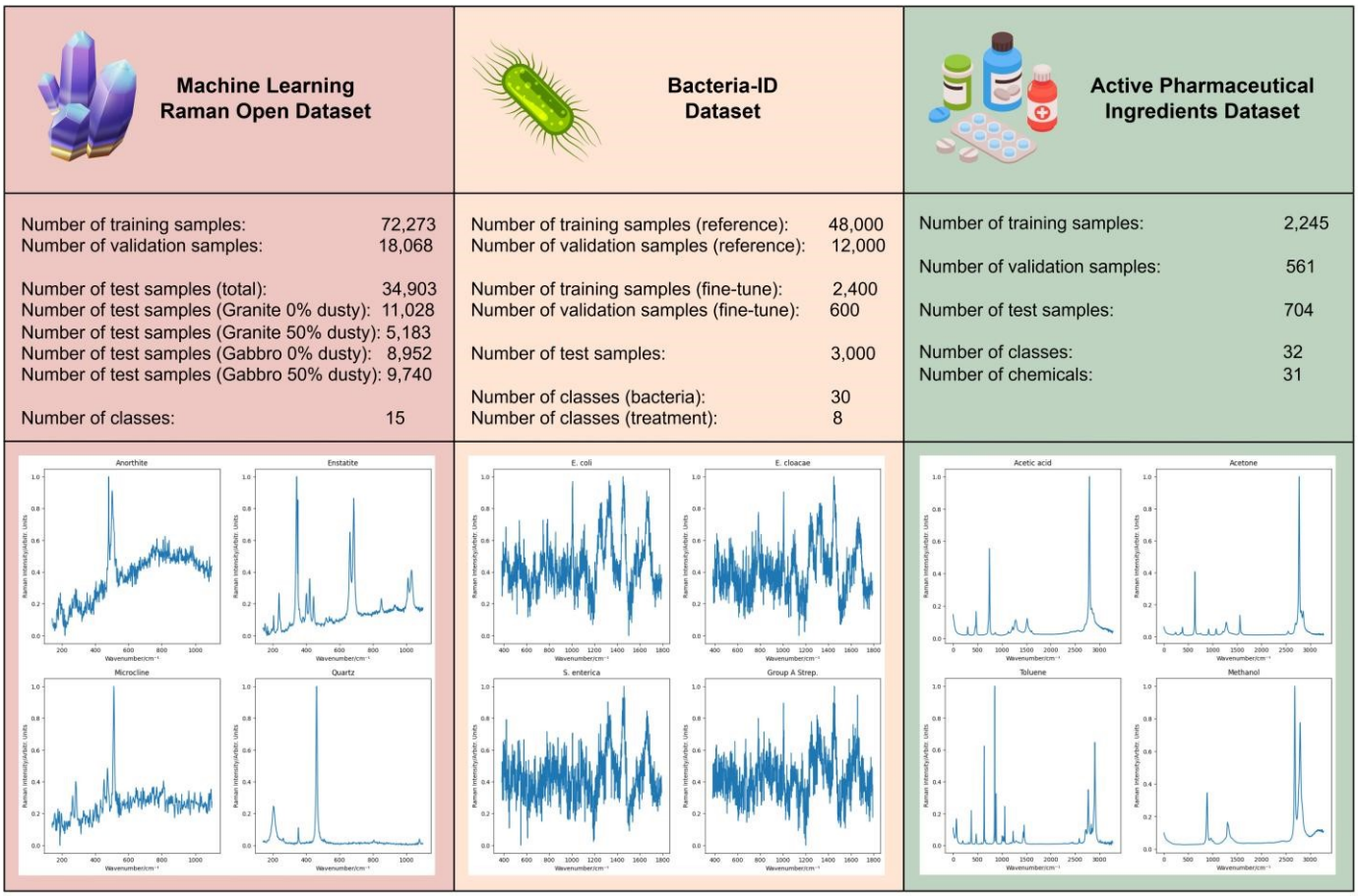

Fig. 1 Dataset overview and qualitative comparison across domains. Top row: the three Raman spectroscopy datasets used in this study: 1. MLROD (mineral spectra), 2. Bacteria-ID (spectra of bacterial species/strains), and 3. API dataset (spectra of active pharmaceutical ingredients). Middle row: dataset-level statistics summarizing scale and label structure including total number of classes. Bottom row: representative spectra randomly sampled from each dataset (intensity vs. Raman shift (cm^{-1})).

Preprocessing For all datasets, we applied only intensity scaling to the raw spectra to improve the numerical stability of the conventional ML and DL models. We intentionally skip any further pre-processing steps such as baseline correction, fluorescence removal, scatter correction, or denoising. Although several procedures exist for each of the processing steps, it is difficult to engineer an effective pre-processing pipeline that can be applied to all the datasets. This is because it is challenging to identify which combination of methods is optimal, as in most cases they lead to worse model performance[52]. Deep learning based approaches for artifact removal have also been developed recently[53 54] but they add to the time complexity of the pre-processing pipeline and can obscure whether performance gains arise from the classifier architecture or from learnable pre-processing. Our goal is to benchmark model behavior under a minimal, reproducible preprocessing regime.

Hyperparameter tuning and Final evaluation runs For each deep learning model-dataset task pair, we performed hyperparameter tuning via 3×3 grid search over the ranges listed in Table 2. For the Random Forest and the SVC, we performed hyperparameter tuning via grid search over the ranges listed in Supplementary Table 1 and Supplementary Table 2. We select the optimal hyperparameters based on the model with the best validation performance. This tuning was performed independently per dataset task to avoid transferring dataset taskspecific choices across domains. After hyperparameter tuning, each model-dataset task pair was retrained for five independent final evaluation runs using its corresponding optimal hyperparameters.

We used the same maximum epoch budget and early-stopping patience across all DL models to maintain a unified benchmark protocol for hyperparameter tuning and final evaluation runs. This choice is intended to compare the published architectures under identical constraints, rather than to independently optimize the training configuration for each DL model. We acknowledge that this shared protocol may not be optimal for all architectures, particularly the transformer-based models, whose performance may depend on learning-rate warmup, weight decay, dropout tuning, and longer training schedules. **MLROD** We divided the MLROD training dataset into the train and validation datasets using a random 80:20 train-val split. For the DL models, the training was for up to 40 epochs, with early stopping applied if the validation accuracy didn't improve for 10 consecutive epochs. We then evaluated the DL model with the

View Article Online
DOI: 10.1039/D6DD00044D

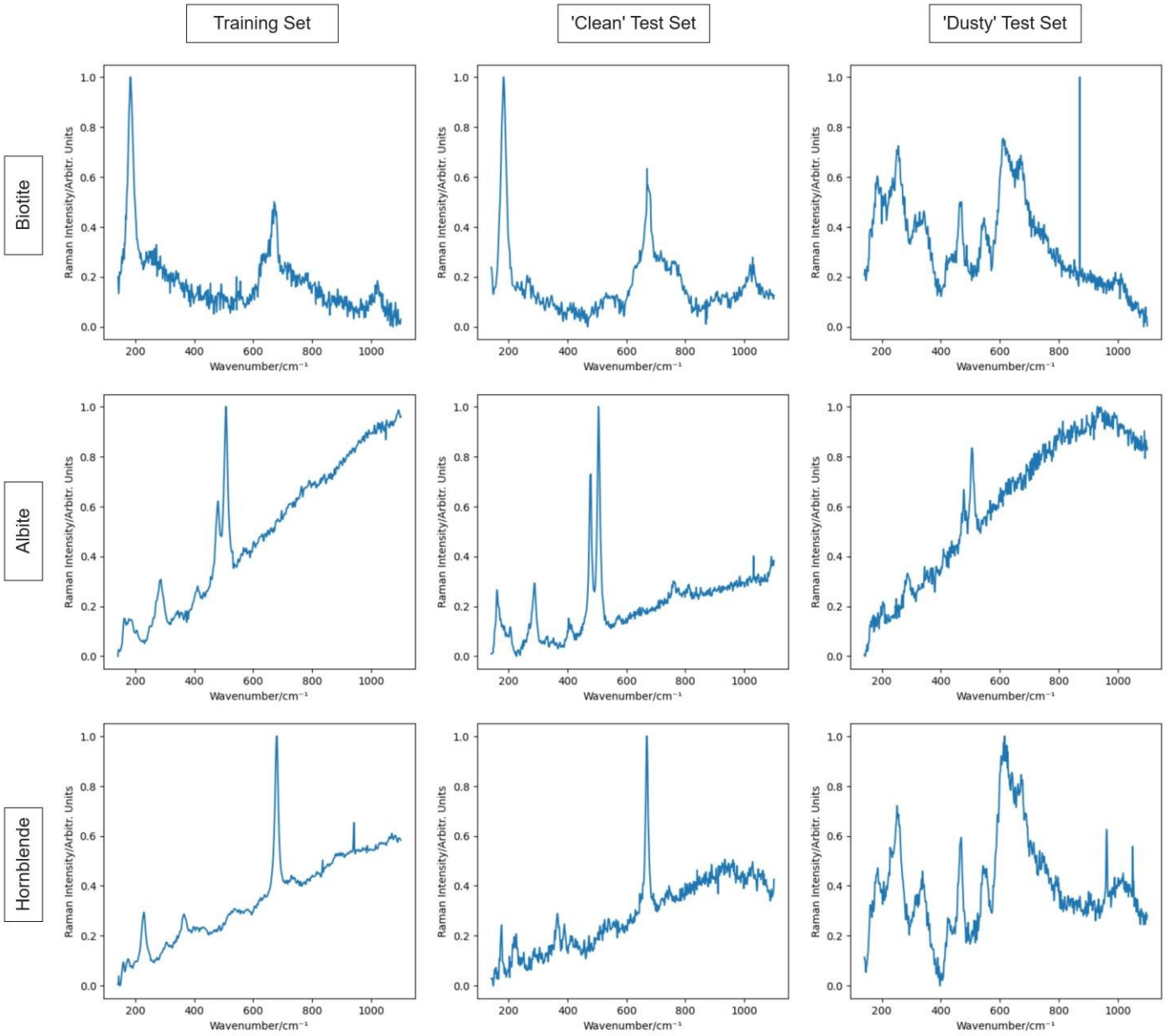

Fig. 2 MLROD spectral shift across evaluation conditions ("clean" vs. "dusty"). Representative Raman spectra for three minerals (Biotite, Albite, Hornblende) shown across the training set (left), 'clean' test set (middle), and 'dusty test' set (right). Each row corresponds to the same material label across splits; each trace is a single example spectrum plotted as Raman intensity (arb. units) versus wavenumber (cm^{-1}). While the training and 'clean' test spectra largely preserve characteristic peak locations and relative band structure, the 'dusty' split exhibits pronounced contamination artifacts such as elevated and drifting baselines, broadened features, and spurious high-intensity components. This illustrates a significant distribution shift that challenges model generalization.

best validation accuracy on the hold-out test dataset. The Random Forest and SVC were fit to the train split and evaluated on the hold-out test set. We also reported the accuracy separately for the clean Gabbro, clean Granite, dusty Gabbro and dusty Granite subsets of the test dataset.

Bacteria-ID The 3 relevant subsets used from the Bacteria-ID dataset to benchmark the selected models were the reference, fine-tune and test sets. We divided the reference and fine-tune sets into their corresponding train and validation datasets using a random 80:20 train-val split.

For the DL models, the pretraining was for up to 40 epochs on the train split of the reference set, with early stopping applied if the accuracy did not improve on the validation split of the reference set for 10 consecutive epochs. The DL model with the best reference validation accuracy was then trained on the fine-tune train split with a tenth of the learning rate used in pretraining. The fine-tuning was for up to 40 epochs with early stopping applied if the accuracy did not improve on the validation split of the fine-tune set for 10 consecutive epochs. We then reported the accuracy on the test set using the model with the best fine-tune validation accuracy.

View Article Online
DOI: 10.1039/D6DD00044D

Since Random Forest and SVC do not have a fine-tuning stage, we introduced an additional hyperparameter indicating whether the training set consisted of only the reference set, only the finetune set or both. For final evaluation, the conventional ML models were fit on the optimal train set and evaluated on the test set.

All the models were trained separately on the two tasks of classifying the isolate and classifying the empiric treatment. **API Dataset** For the API dataset, we generated the test split by randomly sampling 22 disjoint samples each for all the 32 classes and assigned the remaining samples to the train/val set. This approach approximately corresponds to a 80:20 train/val-test split. The train and validation sets were generated by using a subsequent random 80:20 split.

For the DL models, the training was capped at 40 epochs with early stopping if the validation accuracy of the DL model did not improve in the past 10 epochs. We then evaluated the DL model with the best validation accuracy on the test split. The conventional ML models were fit on the train split and evaluated on the test split.

Although the dataset contains 32 labeled compounds, two of them, 4-methyl-2-pentanone and methyl isobutyl ketone, are chemically identical [†]. This is also highlighted in the dataset, where the samples of both these compounds have highly similar Raman spectra, as shown in Figure 3. Therefore, we report the results under two evaluation protocols. First, a 32 class setting that follows the dataset labels. Second, a 31 class setting where these two labels are merged post-hoc during evaluation. All the models are trained under the 32 class problem.

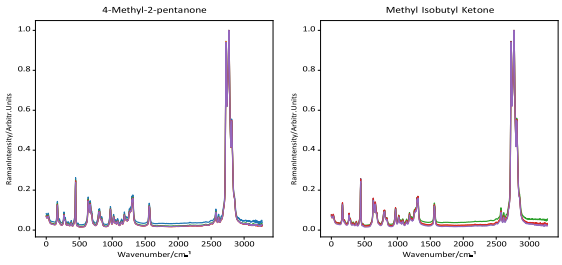

Fig. 3 API dataset label synonymy. Overlaid Raman spectra (five randomly selected measurements per panel) for 4-methyl-2-pentanone (left) and methyl isobutyl ketone (right), plotted as Raman intensity (arb. units) versus wavenumber (cm^{-1}). The near-identical signatures across the two panels reflect that methyl isobutyl ketone and 4-methyl2-pentanone are chemically identical, illustrating potential label aliasing in the dataset and motivating label harmonization during our evaluation

Implementation Details All experiments were run on a server with an Intel Xeon CPU (3.0 GHz), 64 GB RAM, and an NVIDIA RTX A5000 GPU. SANet uses the authors' official PyTorch implementation, while the other DL models were reimplemented based on the details in the original papers or accompanying TensorFlow code. The conventional machine learning models were implemented using the scikit-learn Python library.

6 Results

For all the five Raman-specific DL models and two conventional ML models, we report their performance across the three datasets using accuracy and macro-averaged F1 score in the following subsections. Accuracy provides a simple and intuitive measure of overall correctness across all the samples and it is defined as follows:

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{N} 1(\hat{y}_i = y_i) \quad (1)$$

where N is the total number of samples, $1()$ denotes the indicator function, y_i denotes the ground-truth class label and $\hat{y}_i$ denotes the predicted class label for the i^{th} sample.

However, accuracy can be misleading in the presence of class imbalance. For example, strong performance on more frequent classes can mask poor performance on less frequent classes. To provide a more comprehensive and class-balanced assessment of model performance, we additionally report the macro-averaged F1 score.

The macro-averaged F1 score is the unweighted average of the F1 scores per class and is defined as follows:

$$\text{macro F1} = \frac{1}{C}\sum_{c=1}^{C} \text{F1}_c \quad (2)$$

The F1 score per class is defined as:

$$\text{F1}_c = \frac{2*\text{TP}_c}{2*\text{TP}_c + \text{FP}_c + \text{FN}_c} \quad (3)$$

where TP_c is the number of true positives, FN_c is the number of false negatives, and FP_c is the number of false positives for class c.

By reporting both accuracy and macro-averaged F1 score, we capture the overall classification performance while ensuring that performance on minority classes is not obscured by the majority classes. This benchmark study evaluates single-label multiclass classification performance where the key question is whether the model assigns the correct class label. Accuracy and macro-F1 directly capture this, while AUC-ROC primarily evaluates the ranking of class scores across decision thresholds rather than the final predicted class assignments. Therefore we do not report AUCROC.

To provide simple baseline comparisons, we also report the macro-averaged F1 score and accuracy obtained on the test sets using a random classifier and a majority class classifier. The random classifier assigns labels to test samples by sampling uniformly from the class labels present in the training set. The majority class classifier assigns every test sample to the class that occurs most frequently in the training set. The random classifier was also evaluated over five

[†]https://pubchem.ncbi.nlm.nih.gov/compound/Methyl-Isobutyl-Ketone

View Article Online
DOI: 10.1039/D6DD00044D

independent random draws, while the deterministic majority class classifier has zero variance across runs.

Table 3 Test accuracy of the chosen models on different subsets of the MLROD test dataset. Values are reported as mean ± standard deviation across five independent final evaluation runs. ↑ indicates that higher values correspond to better performance for the reported metrics. Boldface highlights the best-performing model in each column. Underlined values indicate models whose mean plus one standard deviation equals or exceeds the highest mean in each column

Model	Granite 0% dusty ↑	Granite 50% dusty ↑	Gabbro 0% dusty ↑			Gabbro 50% dusty ↑	Overall ↑
Random Classifier	6.8% (± 0.36)	7.04% (± 0.34)	6.7% (± 0.26)			6.53% (± 0.31)	6.66% (± 0.19)
Majority Class Classifier	53.96% (± 0.0)	38.01% (± 0.0)	0.0% (± 0.0)			0.0% (± 0.0)	22.69% (± 0.0)
Random Forest	92.16% (± 0.19)	66.84% (± 0.62)	72.21% (± 2.81)			41.82% (± 0.42)	69.23% (± 0.78)
SVC	91.25% (± 0.21)	61.5% (± 0.27)	84.92% (± 3.34)			39.76% (± 0.24)	70.84% (± 0.93)
Deep CNN[3]	**97.66% (± 2.18)**		**97.77% (± 2.93)**	42.26% (± 1.71)	79.72% (± 1.15)		
			97.21% (± 1.79)	**48.87% (± 2.4)**	**80.34% (± 0.88)**		
			96.37% (± 2.64)	42.12% (± 1.21)	78.3% (± 1.32)		
			81.82% (± 6.41)	42.61% (± 3.73)	72.72% (± 2.41)		
			92.27% (± 1.68)	44.39% (± 3.38)	78.0% (± 1.02)		
SANet[43]	92.3% (± 1.19)						
RamanNet[39]							
Transformer[11]							
RamanFormer[13]							

Table 4 F1 score of the chosen models on different subsets of the MLROD test dataset. Values are reported as mean ± standard deviation across five independent final evaluation runs. The F1 score here refers to the macro-averaged F1 score. ↑ indicates that higher values correspond to better performance for the reported metrics. Boldface highlights the best-performing model in each column. Underlined values indicate models whose mean plus one standard deviation equals or exceeds the highest mean in each column

Model	Granite 0% dusty ↑	Granite 50% dusty ↑	Gabbro 0% dusty ↑	Gabbro 50% dusty ↑	Overall ↑
Random Classifier	0.0307 (± 0.0019)	0.031 (± 0.0012)	0.0279 (± 0.0015)	0.0246 (± 0.0012)	0.0438 (± 0.0009)
Majority Class Classifier	0.1001 (± 0.0)	0.0612 (± 0.0)	0.0 (± 0.0)	0.0 (± 0.0)	0.0247 (± 0.0)
Random Forest	0.4795 (± 0.0193)	0.2785 (± 0.0102)	0.4114 (± 0.0319)	0.1547 (± 0.0015)	0.5537 (± 0.0122)
SVC	0.396 (± 0.0127)	0.2519 (± 0.0086)	0.3866 (± 0.004)	0.1551 (± 0.0006)	0.581 (± 0.008)
Deep CNN[3]	**0.7284 (± 0.0248)**	**0.3779 (± 0.0145)**	0.6661 (± 0.0301)	**0.1812 (± 0.0095)**	**0.7132 (± 0.0202)**
SANet[43]	0.5543 (± 0.0621)	0.3177 (± 0.0211)	**0.715 (± 0.1056)**	0.1494 (± 0.0072)	0.6383 (± 0.0231)
RamanNet[39]	0.5273 (± 0.034)	0.3239 (± 0.0187)	0.5508 (± 0.0352)	0.1629 (± 0.0098)	0.6808 (± 0.0193)
Transformer[11]	0.5041 (± 0.061)	0.2953 (± 0.024)	0.3763 (± 0.0288)	0.1506 (± 0.0115)	0.5794 (± 0.0394)
RamanFormer[13]	0.5693 (± 0.063)	0.3328 (± 0.0284)	0.4076 (± 0.0486)	0.1455 (± 0.0137)	0.6121 (± 0.0234)

6.1 MLROD

Table 3 and Table 4 show the classification results for the five Raman-specific DL models and two conventional ML models trained and tested on the MLROD, along with the two baselines. All the models outperform the random classifier and majority class classifier baselines. In addition, the DL models excluding the Transformer outperform the two conventional ML models.

Another important note is that all the models perform substantially worse on the dusty samples, showcasing the sensitivity of these classifiers to interference. Since dust contaminated spectra are not part of the training dataset, this degradation in performance is due to distribution shift and not overfitting on the training dataset. Therefore, the reported overall test results across all the samples are a reflection of model robustness under domain shift rather than in-distribution classification performance.

The SANet model showcases the best overall accuracy performance, over 7 percentage points higher than the worst DL model (Transformer). The macro-averaged F1 score being significantly lower than the test accuracy across all the subsets of the MLROD test dataset means that one or more classes are performing considerably worse than the dominant classes.

6.2 Bacteria-ID

We report the classification results of the models on two tasks: (1) classifying the isolate (2) classifying the empiric treatment in Table 5. For both the tasks, the models outperform the random classifier and majority class classifier baselines and the Ramanspecific DL models outperform the conventional ML models.

Among the five DL models for the first task, the Transformer performs the worst and classification accuracy of all the other trained DL models are within 2 percentage points of each other. This spread is narrower for the second task, with the test accuracies of the best and worst performing DL models being within 1 percentage point of each other. The comparable macro-averaged F1 score and test accuracy for all the models suggest balanced performance across the classes.

View Article Online
DOI: 10.1039/D6DD00044D

6.3 API Dataset

We show the results of all the models and baselines on the 32 category classification task in the Dataset Label Space and the 31 category classification task in the Chemical Identity Space of the API dataset in Table 6. The five DL models and the two conventional ML models outperform the random classifier and the majority class classifier. However, unlike on the other datasets, the ML models perform comparably to the Raman-specific DL models across both the tasks.

For the former task, the Transformer is the worst performing model and the accuracy of the Deep CNN, RamanNet, RamanFormer and SVC are all within a percentage point of each other followed by the Random Forest and SANet being 1 and 2 percentage points clear from this pack respectively. The macro-averaged F1 score being similar to the test accuracy is indicative of relatively uniform classification performance across the different categories for all the 5 models.

Table 5 Test accuracy and F1 score of the chosen models on the test dataset of the Bacteria-ID dataset. Values are reported as mean ± standard deviation across five independent final evaluation runs. The F1 score here refers to the macro-averaged F1 score. ↑ indicates that higher values correspond to better performance for the reported metrics. Boldface highlights the best-performing model in each column. Underlined values indicate models whose mean plus one standard deviation equals or exceeds the highest mean in each column

	30 isolates				8 treatments	
Model	**Accuracy ↑**	**F1 score ↑**			**Accuracy ↑**	**F1 score ↑**
Random Classifier	3.45% (± 0.4)	0.0344 (± 0.004)			12.26% (± 0.73)	0.1062 (± 0.0055)
Majority Class Classifier	3.33% (± 0.0)	0.0022 (± 0.0)			26.67% (± 0.0)	0.0526 (± 0.0)
Random Forest	58.76% (± 0.78)	0.5735 (± 0.0078)			72.33% (± 0.51)	0.5891 (± 0.0073)
SVC	78.11% (± 0.38)	0.7704 (± 0.0054)			94.13% (± 0.23)	0.9399 (± 0.0026)
Deep CNN[3]		**0.8565 (± 0.0009)**	**97.12% (± 0.15)**	**0.9748 (± 0.0014)**		
SANet[43]	83.27% (± 0.62)	<u>0.8547 (± 0.0063)</u>	96.7% (± 0.2)	0.9718 (± 0.0012)	96.43% (± 0.13)	0.9667 (± 0.0013)
RamanNet[39]		0.8392 (± 0.0082)	96.47% (± 0.21)	0.9672 (± 0.0028)		
Transformer[11]						
RamanFormer[13]	84.63% (± 0.58)	0.8424 (± 0.0063)			<u>97.05% (± 0.14)</u>	<u>0.9725 (± 0.0025)</u>

We observe a consistent increase in performance for the 31 category evaluation task, with all the models achieving a test accuracy greater than 99%. This improvement across the models suggests that most of the errors in the 32-class setting arise from confusion between synonymous labels rather than a failure to learn chemically meaningful representations. This observation is further strengthened by the confusion matrix in Figure 4, which shows most of the misclassifications occurring between the Raman spectra of the chemically identical compounds 4-Methyl-2pentanone and Methyl Isobutyl Ketone.

7 Discussions

Benchmarking plays an important role in identifying the strengths and weaknesses of different approaches for a given problem. In this work, we compare the performance of five supervised deep learning Raman models and two conventional ML models across three open-source Raman spectroscopy datasets under consistent training configurations. With this setup, SANet and Deep CNN demonstrate the strongest performance across the datasets. In contrast, the transformer-based models underperform relative to the strongest CNN-based models under the unified training protocol used in this benchmark. This result should be interpreted within the limits of the present experimental design as transformer performance may depend on training choices that were not exhaustively explored here such as learning-rate warmup, weight decay, dropout tuning and longer training schedules. The stronger performance of RamanFormer (4M parameters) compared with the larger ViT-style Transformer (85M parameters) suggests that model scale and Raman-specific architectural choices can also play an important role.

Another note is that the conventional ML models underperform the DL models on the larger and more diverse MLROD and Bacteria-ID datasets. This suggests that the Raman-specific DL architectures are better able to capture discriminative spectral patterns in these more challenging settings. However, on the smaller API dataset, where the training and test sets are drawn from the same source distribution, the conventional ML models perform comparably to the DL models. This indicates that simpler ML models can remain competitive when the dataset is smaller and the train-test distribution shift is limited.

As shown in Table 3 and Table 4, variations in the spectroscopic hardware and acquisition set-up lead to covariate shifts in the Raman spectra captured in the train and test datasets. Calibration Transfer and Maintenance (CTM) methods in Chemometrics[55] can achieve adaptation between the source(train) and target(test) domains. Many of these CTM methods are not viable for the classification task in MLROD as they require labels for the samples in the target(test) domain[56] [57]. Recent works have proposed unsupervised domain adaptation frameworks[58] [59] but their effectiveness beyond the originally reported datasets are yet to be independently evaluated.

Another approach to solving this issue is by using selfsupervised frameworks on large amounts of unlabeled data to learn more robust representations of Raman spectra. SMAE[60] employs a transformer-based encoder and decoder architecture where the self-supervised

View Article Online
DOI: 10.1039/D6DD00044D

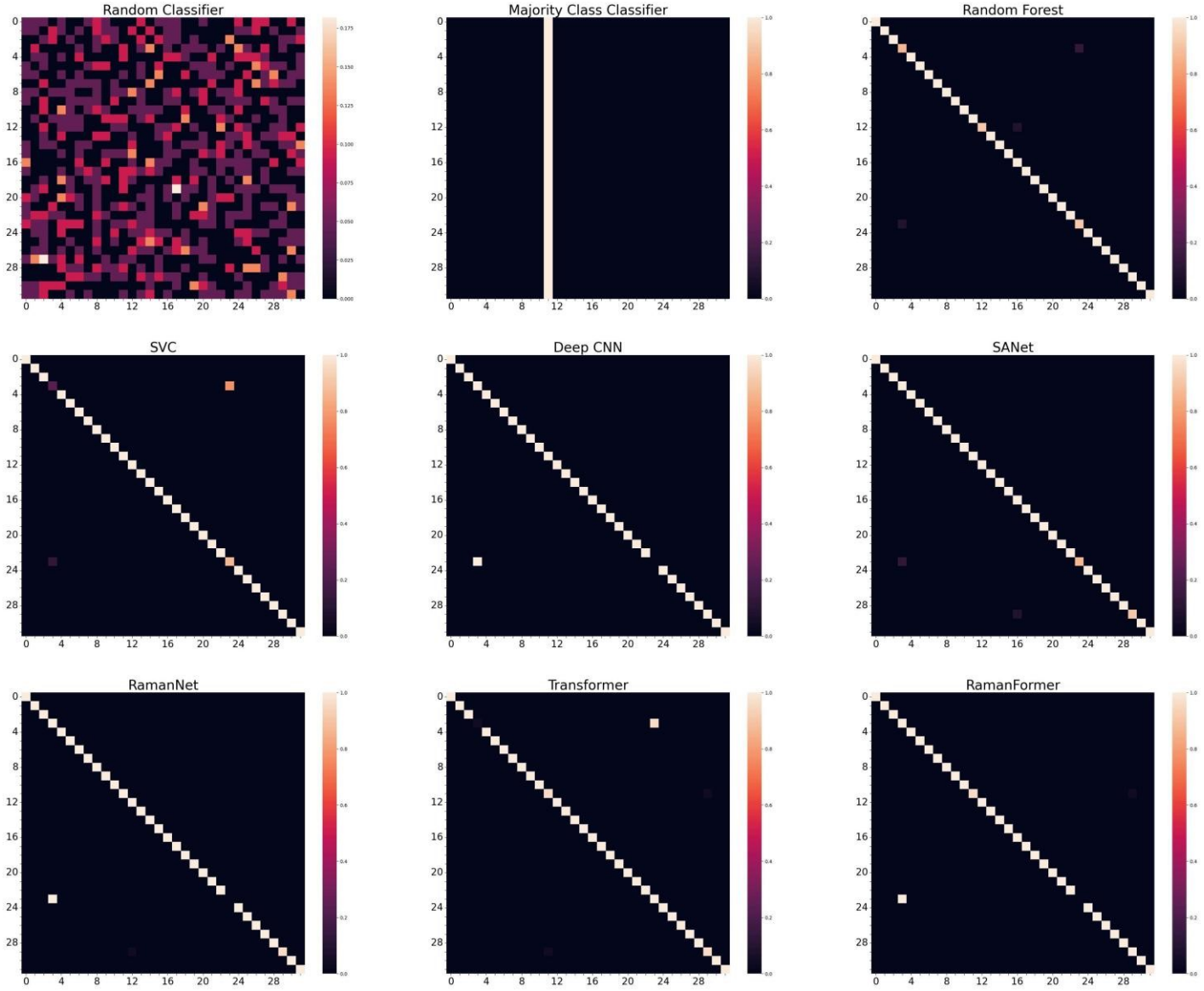

Fig. 4 Class-normalized confusion matrices for the 32 category classification task using the API test set on Dataset Label Space. Rows indicate true class labels and columns indicate predicted class labels. The matrices compare the random classifier, majority class classifier and the benchmarked machine learning and deep learning models. Brighter diagonal entries correspond to a higher classification accuracy. Supplementary Table 7 lists the materials associated with each class label. Most of the misclassifications for all the models are between class labels 3 and 23 corresponding to 4-Methyl-2-pentanone and Methyl Isobutyl Ketone respectively, which are chemically identical.

pretraining task involves recovering the original spectrum from a randomly masked spectrum. The effectiveness of the learned representations was showcased by achieving an unsupervised clustering accuracy of 80.56% for the 30 class pathogenic species identification problem, where Kmeans clustering was applied to the extracted features of the Raman spectra in the test set of the Bacteria-ID dataset. SemiRaman[61] is another framework that combines self-supervised contrastive learning with semi-supervised learning. It involves pretraining an encoder with several contrastive loss functions on augmented views of the same spectrum. The pretrained encoder is then fine-tuned in a multi-stage manner using limited labeled spectra and pseudo-labels. The representations learned in this semi-supervised framework show strong classification performance while using just 5% of the labeled data of a subset of the Bacteria-ID dataset.

However, these approaches primarily address the challenge of limited labeled Raman spectral data and do not explicitly take into account the variations arising from differences in instrumentation and experimental setups. Such variability can lead to significant distribution shifts, which can cause degradation of model performance during testing. Foundation models are designed to be quite robust to input distribution shifts by learning representations from large and diverse datasets. The Deep-spectral Component Filtering (DSCF)[62] is a foundation model that was developed through spectral component resolvable learning on over a million simulated and experimental spectra spanning several

View Article Online
DOI: 10.1039/D6DD00044D

experiments and spectroscopic modalities (UV, IR and Raman). The authors showcased strong performance of DSCF in multi-label classification with concentration estimation using simulated and experimental SERS spectra. However, its performance has not yet been evaluated on standard open-source Raman spectroscopy datasets.

The development of DSCF highlights the need for large-scale and diverse training data. Existing open-source Raman datasets are often restricted in size, chemical diversity or experimental variability. Creating large, curated experimental Raman spectral datasets that span multiple instruments, materials and measurement settings is key to developing a Raman-specific foundation model. These models could serve as a reusable backbone for a wide range of downstream tasks through lightweight fine-tuning using relatively small labeled datasets. This would significantly reduce annotation and training costs as the models would no longer need to be trained from scratch.

Conclusions

In this work, we presented a benchmark of five supervised deep learning models for Raman spectra classification and two conventional ML models evaluated across three public datasets. These datasets capture different real-world challenges like domain shift due to acquisition variability (MLROD), multi-task clinical label-ing (Bacteria-ID) and high-accuracy multi-category classification (API). All the models were evaluated using a unified experimental protocol to ensure fair comparison.

In MLROD, we observed a significant degradation in performance for all the models on the 50% dusty test samples compared to 0% dusty test samples, implying brittleness to background interference and acquisition shift. In Bacteria-ID, isolate classification accuracy sits between 83-86%, while antibiotic treatment prediction is 96-98% for the five DL models with the ML models performing a step below them. Finally for the API dataset, all the models achieved near-ceiling accuracies between 99-100%. Overall, SANet and Deep CNN demonstrate the best overall performance across the datasets, with the other Raman-specific deep learning models not too far behind, followed by the two conventional machine learning models under this unified experimental protocol.

A key trend is that simpler machine learning methods can remain competitive when the classification task has limited traintest distribution shift, while Raman-specific deep learning architectures offer advantages in larger datasets with train and test sets not drawn from the same source.

The results of this benchmarking experiment have also shown that classifying test samples that are in-distribution to the training dataset is significantly easier than test samples suffering from distribution shift due to changes in instruments and acquisition conditions, and additional contaminants. While this study benchmarks only five Raman-specific supervised DL models and two conventional ML models while relying on minimal spectral preprocessing, it establishes a transparent and reproducible baseline for evaluating supervised Raman spectra classifiers. We hope this benchmark will facilitate more rigorous comparisons, thereby enabling researchers to identify effective design choices and develop improved models in the future.

Author contributions

AS: investigation, software, formal analysis, methodology, writing – original draft (lead), visualization. ARK: writing – original draft, writing – review and editing.

Conflicts of interest

There are no conflicts to declare.

Table 6 Test accuracy and F1 score of the chosen models on the test dataset of the API dataset. Values are reported as mean ± standard deviation across five independent final evaluation runs. The F1 score here refers to the macro-averaged F1 score. ↑ indicates that higher values correspond to better performance for the reported metrics. Boldface highlights the best-performing model in each column. Underlined values indicate models whose mean plus one standard deviation equals or exceeds the highest mean in each column

	Dataset Label Space		Chemical Identity Space	
Model	**Accuracy ↑**	**F1 score ↑**	**Accuracy ↑**	**F1 score ↑**
Random Classifier	3.32% (± 0.73)	0.0332 (± 0.0073)	3.32% (± 0.79)	0.0326 (± 0.0078)
Majority Class Classifier	3.12% (± 0.0)	0.0019 (± 0.0)	6.25% (± 0.0)	0.0038 (± 0.0)
Random Forest	98.66% (± 0.17)	0.9867 (± 0.0017)	99.35% (± 0.21)	0.9936 (± 0.002)
SVC	97.16% (± 0.16)	0.965 (± 0.0057)	**100.0% (± 0.0)**	**1.0 (± 0.0)**
Deep CNN[3]		**0.9963 (± 0.0021)**	**100.0% (± 0.0)**	**1.0 (± 0.0)**
SANet[43]		0.9593 (± 0.0049)	99.86% (± 0.13)	0.9985 (± 0.0013)
RamanNet[39]		0.9566 (± 0.0029)	99.83% (± 0.17)	0.9983 (± 0.0016)
Transformer[11]		0.9628 (± 0.0048)	99.29% (± 0.24)	0.9927 (± 0.0024)
RamanFormer[13]			99.91% (± 0.07)	0.9991 (± 0.0007)

View Article Online
DOI: 10.1039/D6DD00044D

Data availability

The Raman spectral datasets used in this work are available from their respective open-source repositories:

1) MLROD - https://odr.io/MLROD#/search/display/1348/eyJkdF9pZCI6IjYwMCJ9

2) Bacteria-ID - https://github.com/csho33/bacteria-ID/blob/master/README.md

3) API - https://springernature.figshare.com/articles/dataset/Opensource_Raman_spectra_of_chemical_compounds_for_active_pharmaceutical_ingredie nt_development/27931131

The implementation of the model architectures, training scripts and evaluation code can be found in the GitHub repo - https://github.com/asineesh/Benchmark_Raman_DeepLearning.

A citable archived version of the repository is available at https://doi.org/10.5281/zenodo.20621093 . The trained models and log files are available at https://doi.org/10.5281/zenodo.19701494.

Acknowledgements

The authors would like to thank Dr. Avinash Kak for valuable discussions, guidance and access to computational resources that supported this work. The authors would also like to thank Dr. Rahul Deshmukh for his feedback on the manuscript.

References

1 Y. Lin, Q. Zhang, H. Chen, S. Liu, K. Peng, X. Wang, L. Zhang, J. Huang, X. Yan, X. Lin *et al.*, *BMC medicine*, 2025, **23**, 97.

2 Y. Roggo, K. Degardin and P. Margot, *Talanta*, 2010, **81**, 988– 995.

3 J. Liu, M. Osadchy, L. Ashton, M. Foster, C. J. Solomon and S. J. Gibson, *Analyst*, 2017, **142**, 4067–4074.

4 M. Chai, W. Hasi, X. Ming, S. Han, G. Fang and Y. Bu, *Journal of Food Composition and Analysis*, 2024, **136**, 106793.

5 R. t. Vulchi, V. Morgunov, R. Junjuri and T. Bocklitz, *Molecules*, 2024, **29**, 4748.

6 S. Guo, C. Beleites, U. Neugebauer, S. Abalde-Cela, N. K. Afseth, F. Alsamad, S. Anand, C. Araujo-Andrade, S. Aškrabi´c, E. Avci, M. Baia, M. Baranska *et al.*, *Analytical Chemistry*, 2020, **92**, 15745–15756.

7 H. M. Monavar, N. Afseth, J. Lozano, R. Alimardani, M. Omid and J. Wold, *Talanta*, 2013, **111**, 98–104.

8 Y. Roggo, K. Degardin and P. Margot, *Talanta*, 2010, **81**, 988– 995.

9 J. Dong, M. Hong, Y. Xu and X. Zheng, *Journal of Chemometrics*, 2019, **33**, e3184.

10 T. Kirchberger-Tolstik, P. Pradhan, M. Vieth, P. Grunert, J. Popp, T. W. Bocklitz and A. Stallmach, *Analytical Chemistry*, 2020, **92**, 13776–13784.

11 B. Liu, K. Liu, X. Qi, W. Zhang and B. Li, *Scientific Reports*, 2023, **13**, 3240.

12 Z. Wang, Y. Li, J. Zhai, S. Yang, B. Sun and P. Liang, *Talanta*, 2024, **275**, 126138.

13 O. C. Koyun, R. K. Keser, S. O. Sahin, D. Bulut, M. Yorulmaz, V. Yucesoy and B. U. Toreyin, *ACS omega*, 2024, **9**, 23241– 23251.

14 N. Coca-Lopez, V. Alcolea-Rodriguez, M. A. Bañares, S. Brockhauser, J. Gorenflot, A. Henderson, R. Hildebrandt, N. Jeliazkova, N. Kochev, E. Lozano Diz *et al.*, *ACS nano*, 2025, **19**, 38189–38218.

15 K. Rebrošová, M. Šiler, O. Samek, F. Ruži˚ ˇcka, S. Bernatová, V. Holá, J. Ježek, P. Zemánek, J. Sokolová and P. Petráš, *Scientific reports*, 2017, **7**, 14846.

16 B. Lafuente, R. T. Downs, H. Yang and N. Stone, *Highlights in mineralogical crystallography*, De Gruyter (O), 2015, pp. 1–30.

17 J. Houston, F. G. Glavin and M. G. Madden, *Journal of Chemical Information and Modeling*, 2020, **60**, 1936–1954.

18 S. Yu, H. Li, X. Li, Y. V. Fu and F. Liu, *Science of The Total Environment*, 2020, **726**, 138477.

19 M. H. Mozaffari and L.-L. Tay, 2021 5th SLAAI International Conference on Artificial Intelligence (SLAAI-ICAI), 2021, pp. 1–6.

20 Y. Qi, D. Hu, M. Zheng, Y. Jiang and Y. P. Chen, *Applied Materials Today*, 2024, **41**, 102499.

21 J. Ho, A. Jain and P. Abbeel, *Advances in neural information processing systems*, 2020, **33**, 6840–6851.

22 M. H. Mozaffari and L.-L. Tay, *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 2022, **272**, 120961.

23 X. Fan, W. Ming, H. Zeng, Z. Zhang and H. Lu, *Analyst*, 2019, **144**, 1789–1798.

24 K. He, X. Zhang, S. Ren and J. Sun, Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

25 M. K. Maruthamuthu, A. H. Raffiee, D. M. De Oliveira, A. M. Ardekani and M. S. Verma, *MicrobiologyOpen*, 2020, **9**, e1122.

26 K. Simonyan and A. Zisserman, *Very Deep Convolutional Networks for Large-Scale Image Recognition*, 2015, https:// arxiv.org/abs/1409.1556.

27 M. S. Primrose, J. Giblin, C. Smith, M. R. Anguita and G. H. Weedon, Algorithms, Technologies, and Applications for Multispectral and Hyperspectral Imaging XXVIII, 2022, pp. 98–108.

28 M. S. Primrose, G. H. Weedon and J. Giblin, Chemical, Biological, Radiological, Nuclear, and Explosives (CBRNE) Sensing XXIV, 2023, pp. 151–160.

29 G. Huang, Z. Liu, L. Van Der Maaten and K. Q. Weinberger, Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4700–4708.

30 Y. Qin, J. Qiu, N. Tang, Y. He and L. Fan, *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 2024, **309**, 123854.

31 A. Krizhevsky, I. Sutskever and G. E. Hinton, *Advances in neural information processing systems*, 2012, **25**, year.

32 Y. E. Kok, A. Crisford, A. Parkes, S. Venkateswaran, R. Oreffo, S. Mahajan and M. Pound, *Scientific Reports*, 2024, **14**, 15902.

33 R. Ullah, S. Khan, Z. Ali, H. Ali, A. Ahmad and I. Ahmed, *Photodiagnosis and Photodynamic Therapy*, 2022, **39**, 102924.

View Article Online
DOI: 10.1039/D6DD00044D

34 F. Du, L. He, X. Lu, Y.-q. Li and Y. Yuan, *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 2023, **289**, 122216.

35 Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin and B. Guo, Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 10012–10022. 36 Y. Wang, L. Xu, L. Shang, H. Peng, K. Liu, X. Bao, X. Tang, P. Liang, Y. Wang, M. Zheng *et al.*, *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 2026, **344**, 126662.

37 Z. Zhao, Z. Liu, M. Ji, X. Zhao, Q. Zhu and M. Huang, *Chemometrics and Intelligent Laboratory Systems*, 2023, **234**, 104757.

38 C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2818–2826.

39 N. Ibtehaz, M. E. Chowdhury, A. Khandakar, S. Kiranyaz, M. S. Rahman and S. M. Zughaier, *Neural Computing and Applications*, 2023, **35**, 18719–18735.

40 G. Yin, L. Li, S. Lu, Y. Yin, Y. Su, Y. Zeng, M. Luo, M. Ma, H. Zhou, L. Orlandini *et al.*, *Journal of Raman Spectroscopy*, 2021, **52**, 949–958.

41 M. Erzina, A. Trelin, O. Guselnikova, B. Dvorankova, K. Strnadova, A. Perminova, P. Ulbrich, D. Mares, V. Jerabek, R. Elashnikov *et al.*, *Sensors and Actuators B: Chemical*, 2020, **308**, 127660.

42 C.-S. Ho, N. Jean, C. A. Hogan, L. Blackmon, S. S. Jeffrey, M. Holodniy, N. Banaei, A. A. Saleh, S. Ermon and J. Dionne, *Nature communications*, 2019, **10**, 4927.

43 L. Deng, Y. Zhong, M. Wang, X. Zheng and J. Zhang, *IEEE Journal of Biomedical and Health Informatics*, 2021, **26**, 369– 378.

44 Y. Liu, Y. Gao, R. Niu, Z. Zhang, G.-W. Lu, H. Hu, T. Liu and Z. Cheng, *Analytica Chimica Acta*, 2024, **1332**, 343376.

45 B. Zhou, Y.-K. Tong, R. Zhang and A. Ye, *RSC advances*, 2022, **12**, 26463–26469.

46 C. Lange, M. Altmann, D. Stors, S. Seidel, K. Moynahan, L. Cai, S. Born, P. Neubauer and M. N. C. Bournazou, *Measurement*, 2025, 118884.

47 Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, *Proceedings of the IEEE*, 2002, **86**, 2278–2324.

48 A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser and I. Polosukhin, *Advances in neural information processing systems*, 2017, **30**, year.

49 A. Dosovitskiy, *arXiv preprint arXiv:2010.11929*, 2020.

50 G. Berlanga, Q. Williams and N. Temiquel, *Earth and Space Science*, 2022, **9**, e2021EA002125.

51 A. R. Flanagan and F. G. Glavin, *Scientific Data*, 2025, **12**, 498.

52 J. Engel, J. Gerretzen, E. Szymanska, J. J. Jansen, G. Downey,´ L. Blanchet and L. M. Buydens, *TrAC Trends in Analytical Chemistry*, 2013, **50**, 96–106.

53 M. T. Gebrekidan, C. Knipfer and A. S. Braeuer, *Journal of Raman Spectroscopy*, 2021, **52**, 723–736.

54 J. Sjöberg, N. Siminea, A. Paun, A. Lita, M. Larion and I. Petre,˘ *Advanced Optical Materials*, 2025, 2500736.

55 R. Nikzad-Langerodi and E. Andries, *Journal of Chemometrics*, 2021, **35**, e3373.

56 J. Lai, M. Li, S. Chen, J. Long, Y. Chen, H. Lu, C. Zou and Z. Zhang, *Analytical Chemistry*, 2025, **97**, 19009–19018.

57 T. Boucher, M. D. Dyar and S. Mahadevan, *Journal of Chemometrics*, 2017, **31**, e2877.

58 A. Umprecht, V. F. Diaz, B. Hüpfl, B. Kozma, A. Schwaighofer, M. Henson, R. Nikzad-Langerodi and O. Spadiut, *Measurement*, 2025, **255**, 117906.

59 Z. Zhang, Y. Liu, C. Chen, X. Lv and C. Chen, *Sensors*, 2025, **25**, 6186.

60 P. Ren, R.-g. Zhou and Y. Li, *Expert Systems with Applications*, 2025, 128576.

61 Z. Sun and Z. Wang, *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 2025, 127356.

62 B. Xue, X. Bi, Z. Dong, Y. Xu, M. Liang, X. Fang, Y. Yuan, R. Wang, S. Liu, R. Jiao *et al.*, *Nature Machine Intelligence*, 2025, **7**, 743–757.

View Article Online
DOI: 10.1039/D6DD00044D

The Raman spectral datasets used in this work are available from their respective open-source repositories:

1) MLROD - https://odr.io/MLROD#/search/display/1348/eyJkdF9pZCI6IjYwMCJ9
2) Bacteria-ID - https://github.com/csho33/bacteria-ID/blob/master/README.md
3) API - https://springernature.figshare.com/articles/dataset/Open-source_Raman_spectra_of_chemical_compounds_for_active_pharmaceutical_ingredient_development/27931131

The implementation of the model architectures, training scripts and evaluation code can be found in the GitHub repo - https://github.com/asineesh/Benchmark_Raman_DeepLearning.

A citable archived version of the repository is available at https://doi.org/10.5281/zenodo.20621093 . The trained models and log files are available at https://doi.org/10.5281/zenodo.19701494.