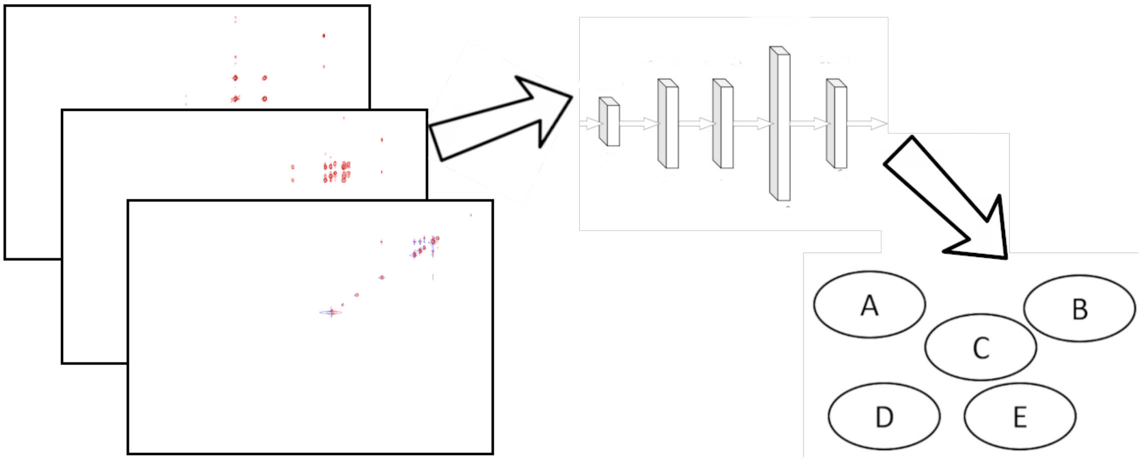
Graphical Abstract

Direct deduction of chemical class from NMR spectra

Stefan Kuhn, Carlos Cobas, Agustin Barba, Simon Colreavy-Donnelly, Fabio Caraffini, Ricardo Moreira Borges



Highlights

Direct deduction of chemical class from NMR spectra

Stefan Kuhn, Carlos Cobas, Agustin Barba, Simon Colreavy-Donnelly, Fabio Caraffini, Ricardo Moreira Borges

- Chemical class can be deduced directly from NMR spectra using a convolutional neural network.
- Other methods were found to be not suitable for this task.
- We also present a large set of uniformly processed NMR spectra as images.

Direct deduction of chemical class from NMR spectra

Stefan Kuhn[a,b,*], Carlos Cobas[c], Agustin Barba[c], Simon Colreavy-Donnelly[d], Fabio Caraffini[e,b], Ricardo Moreira Borges[f]

[a] *Institute of Computer Science, University of Tartu, Narva mnt. 18, Tartu, 51009, Tartumaa, Estonia*

[b] *School of Computer Science and Informatics, De Montfort University, The Gateway, Leicester, LE1 9BH, United Kingdom*

[c] *Mestrelab Research, S.L., Feliciano Barrera 9B Bajo, Santiago de Compostela, 15706, A Coruña, Spain*

[d] *School of Computer Science and Information Systems, University of Limerick, Castletroy, Limerick, V94 T9PX, Limerick, Ireland*

[e] *Department of Computer Science, Swansea University, Computational Foundry, Swansea, SA18EN, Wales, United Kingdom*

[f] *Instituto de Pesquisas de Produtos Naturais Walter Mors, Universidade Federal do Rio de Janeiro, 373 Avenida Carlos Chagas Filho, Rio de Janeiro, 21941-903, RJ, Brazil*

Abstract

This paper presents a proof-of-concept method for classifying chemical compounds directly from NMR data without performing structure elucidation. This can help to reduce the time in finding good structure candidates, as in most cases matching must be done by a human engineer, or at the very least a process for matching must be meaningfully interpreted by one. The method identified as suitable for classification is a convolutional neural network (CNN). Other methods, including clustering and image registration, have not been found to be suitable for the task in a comparative analysis. The result shows that deep learning can offer solutions to spectral interpretation problems.

Keywords: NMR, chemical classification, image processing, convolutional neural network, deep learning

[*] E-mail address: stefan.kuhn@ut.ee

1. Introduction

Nuclear Magnetic Resonance (NMR) spectroscopy is an established technique in analytical chemistry. As a result of its rich structural and dynamic information content, it is particularly suitable for compound identification. However, full elucidation may not always be possible or even necessary since some properties might be achieved directly from the spectra. If this is the case, a prioritization of substances to be closely investigated for compound assignment can be performed in the early stages of a study.

A previous example of this idea was demonstrated in [1], where the authors showed that the existence of certain substructures can be concluded from profiles in the spectra. Another potentially useful application is chemical classification. These rely strongly on annotated chemical entities to provide a computable chemical taxonomy based on substructures. In this paper, we infer chemical taxonomy from the spectra. Similar approaches have been applied with Mass Spectrometry (MS) data where chemical classifications can be achieved from fragmentation patterns on MS/MS data [2].

Together with the MolNetEnhancer method [3] and Molecular Networks [4], chemical classification has been shown to be a valuable tool for the annotation of compounds of unknown compounds and for the classification of samples for prioritization, for example. The authors have decided to work with spectral images in this current study, but working with raw data would be another option. Working with the images, we frame the problem very much as a digital image processing and machine vision methodology. Computer science has developed a wide range of techniques here, in particular over the last few years, with image processing gaining importance in areas like autonomous driving or security monitoring (see [5] for an overview). While these techniques are mostly used to process video or photographic images, they can also be used to process data from scientific instruments [6]. On this basis, in this paper, we want to examine the question of whether chemical classes can be deduced directly from NMR spectra, using digital image processing techniques. If so, the best method will be identified.

2. Results

2.1. Image similarities

Table 1 shows the results of the image similarity methods described in Section 4.2. First, we can see that, as expected, all of the methods find the spectra identical to themselves. Calculating the similarities within and between superclasses gives, on average, similar results. For example, using MobileNetV2 CS gives an average similarity of 0.94 within the superclasses and of 0.91 between superclasses. Although the similarity is slightly lower between superclasses, the difference is not significant. The same is true for the other methods, where SSICompare even gives the same average similarity. From this we conclude that these methods are not suitable for classifying the NMR spectra according to the chemical class of the compound.

2.2. Image registration

In order to test the image registration, we have trained the CNN model used in VoxelMorph for 20 epochs to ensure convergence with the training images of one class. We then register test images from that class and some other classes. We then sum up pixel by pixel the absolute value of the shifts found by the registration. Similar images should need less change and therefore have a smaller shift overall. An example run is available on the GitHub repository in *results/voxelmorph.txt*. The class used for training here is the Benzoids. The convergence in training is visible. We then sum up the shifts for three Benzoids, the resulting values are 43499, 38834, and 41487. Doing the same for three Organic Oxygen compounds gives 50596, 44232, and 44537. For other classes, the results were similar. In some cases, the other molecules needed even less change than those from the same class. From those values we can conclude that this method did not find useful information.

2.3. Clustering based on deep learning

The authors have executed the clustering as described in Section 4.4 on the 400 images in the training set. These are from nine superclasses, so we did the clustering with nine clusters. The nine clusters formed are shown in Table 2, together with the superclasses from which the compounds clustered in them came from. From this table, we can conclude that the clustering is not conclusive, clusters contain compounds from a variety of classes, and the classes are distributed over all clusters.

In order to test this, we make an assignment of classes to clusters that optimizes the number of correct guesses. The pairs of clusters/classes are Alkaloids and derivatives/9, Benzenoids/8, Lipids and lipid-like molecules/4,

		MobileNetV2 CS	MobileNetV2 E	ORBCompare	SSICompare
With itself	1-1	1	0	1	1
	5-5	1	0	1	1
	9-9	1	0	1	1
Within superclass	1-2	0.96	8.84	0.97	0.97
	1-3	0.96	8.19	0.99	0.98
	1-4	0.96	9.21	0.99	0.98
	5-6	0.92	12.83	0.95	0.96
	5-7	0.93	12.58	0.55	0.94
	5-8	0.94	11.30	0.75	0.96
	9-10	0.98	6.69	0.99	0.99
	9-11	0.91	13.87	1.00	0.99
	9-12	0.96	9.63	0.90	0.98
Average		0.94	10.34	0.88	0.97
Between superclasses	1-5	0.91	13.64	0.98	0.96
	1-6	0.92	13.27	0.93	0.98
	1-7	0.89	15.33	0.63	0.95
	1-8	0.93	11.96	0.75	0.97
	2-5	0.92	13.22	0.97	0.97
	2-6	0.91	13.94	0.95	0.98
	2-7	0.90	14.92	0.60	0.95
	2-8	0.94	11.65	0.82	0.97
	3-5	0.92	13.09	0.98	0.97
	3-6	0.94	10.89	0.92	0.98
	3-7	0.90	14.51	0.66	0.96
	3-8	0.94	11.24	0.76	0.97
	4-5	0.91	14.18	0.98	0.97
	4-6	0.90	14.51	0.94	0.98
	4-7	0.89	15.38	0.62	0.96
	4-8	0.93	12.26	0.70	0.97
	1-9	0.93	12.05	1.00	0.98
	1-10	0.93	12.00	0.98	0.98
	1-11	0.87	16.50	1.00	0.99
	...	...	...	...	...
Average		0.91	12.84	0.86	0.97

Table 1: Distance measures achieved using various clustering methods. The numbers in the column “Images” column indicate the images used as explained in Section 4.2. The full table is found in Supplemental Material S1.

Actual superclasses/Cluster number	1	2	3	4	5	6	7	8	9	Sum
Alkaloids and derivatives	0	5	0	0	2	1	1	0	9	18
Benzenoids	1	17	2	6	11	4	2	4	8	55
Lipids and lipid-like molecules	0	4	4	23	13	15	3	3	16	81
Nucleosides, nucleotides, and analogues	0	1	1	0	0	1	5	0	5	13
Organic acids and derivatives	6	2	17	13	2	16	6	1	3	66
Organic nitrogen compounds	1	2	0	0	0	8	1	0	0	12
Organic oxygen compounds	1	8	14	3	2	24	5	1	4	62
Organoheterocyclic compounds	0	11	10	4	7	6	6	0	5	49
Phenylpropanoids and polyketides	1	11	0	4	3	2	7	1	15	44
Sum	10	61	48	53	40	77	36	10	65	400

Table 2: Results of k-means clustering based on deep learning. The 9 clusters formed are shown from left to right. The actual classes of the members are shown from top to bottom. Sums of classes are given on the right, and sums of clusters are given on the bottom. For example, there were 18 samples from the class of "Alkaloids and derivates". Of these, 0 were clustered in cluster 1, 5 in cluster 2, 0 in cluster 3 and so forth. On the other hand, cluster 1 had 10 members altogether; it consisted of 0 compounds from "Alkaloids and derivates", 1 from "Benzoids" and so forth.

Nucleosides, nucleotides, and analogues/7, Organic acids and derivatives/6, Organic nitrogen compounds/6, Organic oxygen compounds/3, Organoheterocyclic compounds/5, Phenylpropanoids and polyketides/2. The sum of correctly classified samples in this combination is 87, which is 21.75%. This is clearly better than the value to expect from a random guess, but also much worse than the CNN results.

The results using the other clustering methods are similar. For agglomerative clustering, the correctly classified samples were 22.75%, see Table 3 for details. Affinity-propagation clustering produced 41 clusters with up to 28 entries, many having only one, using the default settings in Scikit-learn. Changing the parameter preference and random_state did not produce significantly better results. Spectral clustering produced similarly inconclusive results.

These results show that the overall clustering is inferior to the CNN approach for this problem.

Actual superclasses/Cluster number	1	2	3	4	5	6	7	8	9	Sum
Alkaloids and derivatives	0	6	0	1	9	0	0	1	1	18
Benzenoids	2	23	6	4	6	1	1	6	6	55
Lipids and lipid-like molecules	10	19	4	19	15	2	0	1	11	81
Nucleosides, nucleotides, and analogues	1	0	0	2	5	2	0	3	0	13
Organic acids and derivatives	15	5	1	19	7	8	6	3	2	66
Organic nitrogen compounds	0	1	0	8	0	0	1	2	0	12
Organic oxygen compounds	3	9	1	25	5	13	1	4	1	62
Organoheterocyclic compounds	8	14	1	7	6	5	0	2	6	49
Phenylpropanoids and polyketides	0	10	1	3	14	0	0	4	12	44
Sum	39	87	14	88	67	31	9	26	39	400

Table 3: Results of agglomerative clustering with ward linkage based on deep learning. The 9 clusters formed are shown from left to right. The actual classes of the members are shown from top to bottom. The sums of classes are given on the right, and the sums of clusters are given on the bottom. For example, there were 18 samples from the class of "Alkaloids and derivates". Of these, 0 were clustered in cluster 1, 6 in cluster 2, 0 in cluster 3 and so forth. On the other hand, cluster 1 had 39 members altogether; it consisted of 0 compounds from "Alkaloids and derivates", 2 from "Benzoids" and so on.

2.4. CNN

The accuracy achieved using the simple Convolutional Neural Network described in Section 4.5 is given in Table 4. The results of the tests are provided in the *results* directory of the GitHub repository. Since this is using a fixed test set, we have also run the HMBC case using cross-validation. The average results for the metrics in Table 4 are 53.52, 53.53, 51.76, 55.54, and 48.07%. This is less than for our test set, but still significantly above the random guess. It should be noted that the same training set used here is used for image registration and clustering, indicating that direct comparison should be made with the results in Table 4.

The numbers show as the main finding that a CNN is able to distinguish chemical classes by using visual NMR spectra. Since we have 9 classes, a random selection of classes should give about 11% accuracy. All the accuracies from this method are well above that. HMBC alone has better results than HSQC alone, and both together are similar to HSQC. For single networks, this is in line with the results reported in [1] for substructure classification. It should be noted that, as explained in Section 4.5, the results are likely to be optimized, and this represents a proof of concept. Unlike [1], the combined results are not similar to the HMBC results, but rather to the HSQC result.

		HMBC	HSQC	HMBC and HSQC
Accuracy	Average	62.47	43.42	48.76
	Min/Max	55.23/64.76	38.09/51.42	37.14/62.85
F1-score	Average	62.10	44.36	48.97
	Min/Max	47.99/69.84	40.72/52.36	35.86/63.74
Recall	Average	60.47	43.67	47.29
	Min/Max	41.14/69.09	40.36/5104.	34.89/61.54
Precision	Average	64.15	45.12	51.02
	Min/Max	58.00/70.60	41.02/53.79	36.93/66.19
Matthews Correlation Coefficient	Average	56.71	34.13	40.34
	Min/Max	47.98/59.57	27.43/43.	26.54/57.36

Table 4: Classification metric achieved by the network described in Section 4.5, over ten runs. All numbers in %.

3. Discussion

The results show that the convolutional neural network is the only method that is capable of directly determining a chemical class from NMR spectra. We can achieve an accuracy of more than 60% with an unoptimised standard CNN. Almost certainly, optimizing the network would increase the accuracy even more. On the contrary, other methods have not provided an accuracy significantly above the random result. This is perhaps unsurprisingly true for conventional image similarities. It is also true for the clustering and image registration methods we have tried. It should be noted that there is a wide range of such methods available, and others might give better results. In particular, clustering might give better results with the right combination of distance metrics, clustering method, and parameters. Considering that CNN is also the first attempt and a standard architecture, the result is significant.

In the case of CNN, the precision is less than that of the same network for the classification of substructures in [1]. This is reasonable, since the concept of chemical class as implemented in ClassyFire is more complex than that of a single substructure. As in [1], we get a better result from the HMBC spectra than from the HSQC spectra. In this paper, we obtain an accuracy for the combined spectra closer to the HSQC data than the HMBC data. This shows that the processing method is not suitable here, since the higher information content of the HMBC is effectively lost.

4. Methods

4.1. The data

The source of NMR spectra for this article is nmrshiftdb2 [7] and the Biological Magnetic Resonance Bank (BMRB) [8]. For BMRB, raw NMR data for several small molecules are available for download at `https://bmrb.io/ftp/pub/bmrb/metabolomics/entry_directories/`. We have downloaded, using a script, the HMBC and HSQC spectra for all compounds for which both spectra exist. The structures were also downloaded in the `mol` file format. Similarly, nmrshiftdb2 offers a raw data download for some compounds. We have downloaded HMBC and HSQC spectra together with the structures in mol file format here as well, where available.

Then all structures were submitted to the ClassyFire interface at `http://classyfire.wishartlab.com/queries/new`. ClassyFire [9] is a software that classifies chemical compounds according to a well-defined ontology by structural features. The ontology term used in this article is ClassyFire *superclass*. The use of superclasses was mandated because there is a reasonable number of examples of superclasses in the data, whereas more specific terms do not have enough examples for training. We have only used superclasses, which have a minimum number of examples of 15. We also performed a stratified random split into training and test data. The final classes and their examples are shown in Table 5.

All 2D spectra were processed automatically starting from the raw time domain data using a custom Mnova script. Mnova NMR [10] is a software package for the processing, analysis and prediction of NMR data. It includes a scripting engine (Javascript based) for tasks automation and custom developments. In all cases, a zero filling level was applied along the direct dimension (F2). For the indirect dimension (F1), zero filling was applied so that the number of points is the same as in F2. Linear prediction was not used. For non-phase sensitive experiments (e.g. COSY), the magnitude was calculated, and the apodisation functions applied were sine bell for F2 and sine square for F1. Phase-sensitive experiments (e.g. HSQC) were automatically phase-corrected, and the apodization functions were sine bell (90°) for both dimensions. As a result, the spectral images have the same scale, no additional rulers, grids, or other formatting in the image, and the same depth display of the z-dimension. This is different from the approach in [1], where images uploaded to BMRB were used. These images are not uniform with respect to the settings mentioned. The images used, in directories for the

Superclass	Training instances	Test instances	Total instances
Alkaloids and derivatives	13/5	3/2	16/7
Benzenoids	48/7	12/0	60/7
Lipids and lipid-like molecules	68/13	16/5	84/18
Nucleosides, nucleotides, and analogues	13/0	4/0	17/0
Organic acids and derivatives	64/2	19/0	83/2
Organic nitrogen compounds	12/0	3/0	15/0
Organic oxygen compounds	52/10	15/3	67/13
Organoheterocyclic compounds	45/4	11/2	56/6
Phenylpropanoids and polyketides	39/5	9/1	48/6
Sum	354/46	92/13	446/59

Table 5: Overview of the number of samples for which HMBC and HSQC spectra were available for this project. In the pairs of numbers, the first number refers to BMRB, and the second to nmrshiftdb2.

superclasses and the training / test set, are available in the *classesbothfinal* folder in the GitHub repository of the project. Such a large set of uniformly processed NMR data, to the authors' knowledge, is not available so far.

4.2. Image similarities

As a working hypothesis, the authors assume that the spectra of compounds of the same class are more "similar" than the spectra of compounds of different classes. On the basis of this, it should be possible to determine the distance of a spectrum from the spectra of compounds of the various ClassyFire superclasses. The most similar spectrum should then tell which class the new compound belongs to. To test this hypothesis, the authors have tested a number of commonly used image similarity metrics.

- MobileNetV2 pre-trained ML model by Google [11], combined with cosine similarity (MobileNetV2 CS in here), using the implementation from Keras.
- MobileNetV2 pre-trained ML model by Google, combined with Euclidean distance (MobileNetV2 E in here), using the implementation from Keras.

- ORB key features matching [12], as implemented in OpenCV (`https://docs.opencv.org/3.4/d1/d89/tutorial_py_orb.html`).
- Structural Similarity Index (SSIM)[13], as implemented in OpenCV (`https://docs.opencv.org/4.x/dd/d3d/tutorial_gpu_basics_similarity.html`).

These methods were applied using the mentioned implementations. The authors first calculated the similarities between two instances of the same images to check the methods, since we expect identity here. Then, we calculated the similarities between HMBC images from the same class and between images from different classes. The average of similarities in and between classes was calculated. If the methods can perform a proper classification, we would expect the similarities within the classes to be significantly higher than between different classes. The BMRB numbers of the images used and their numbers in Table 1 are as follows: Alkaloids and derivatives: bmse001010 (1), bmse001193 (2), bmse001248 (3), bmse001281 (4), Lipids and lipid-like molecules: bmse000317 (5), bmse000394 (6), bmse000478 (7), bmse000484 (8), Organic oxygen compounds: bmse000302 (9), bmse000303 (10), bmse000304 (11), bmse000306 (12). The code for the calculations is available in the *python* folder in the github repository of the project.

4.3. Image registration

Image registration in general means transforming different sets of data into one coordinate system. Possible data include images. A typical application would be to align MRI images of different slices of a human brain. For our purposes, we measure the amount of transformation needed to align two spectra. Similarly to the image similarity calculation, our hypothesis is that the spectra for compounds of the same class would require less change to align.

To test this, we have used the VoxelMorph image registration program [14]. VoxelMorph is based on deep learning and is considered an advanced image registration technique. We train it with the images of one class and then try to register images from the same class and other classes to see the necessary changes. The code for the calculations is available in the *python* folder in the GitHub repository of the project.

4.4. Clustering based on deep learning

Clustering techniques can be used to group instances by similarity. Similarly as before, if the spectra could be clustered by their similarity, then the ClassyFire superclass of a compound could be determined by the cluster into which its spectra fall. We use a pre-trained network for feature extraction and execute k-means clustering, following `https://towardsdatascience.com/how-to-cluster-images-based-on-visual-similarity-cd6e7209fe34`. k-means clustering seems particularly appropriate in this case since the number of clusters needed is known and must be the same as the number of ClassyFire superclasses.

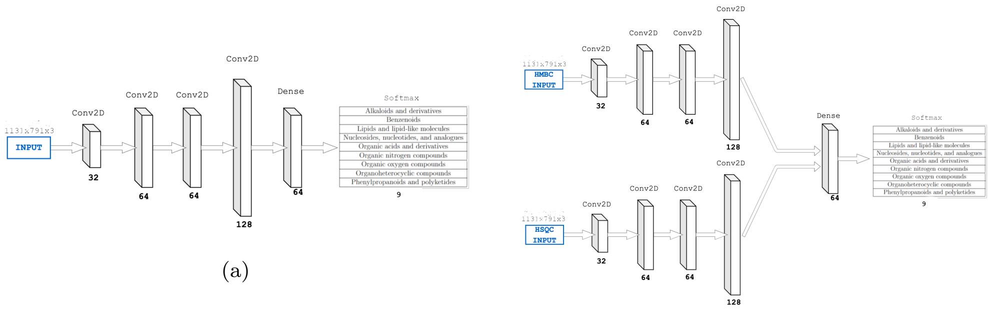
In order to make sure that the clustering algorithm is not at fault for this, we have tried to find more clustering methods, namely agglomerative clustering with ward linkage, affinity propagation, and spectral clustering. The algorithms were taken from Scikit-learn [15]. The code is available in the *python* folder in the GitHub repository of the project.

4.5. CNN

Convolutional neural networks (CNNs) have become a popular choice for many tasks, in particular, image processing. Typical tasks here are image classification (e.g., finding all images in a set that contain a car) or image segmentation (finding all cars in an image). The authors do not cover details of the work and many applications here; for details and literature references, see the recent review [16]. What is important in this context is the ability of CNNs to extract complex information. We have shown in [1] that chemical information is accessible for CNNs from the image representation of the NMR spectra. CNNs have also been shown to process raw data from NMR spectra [17]. This paper focuses purely on visual information.

The authors used the same networks as in [1]. For the prediction of HMBC and HSQC prediction only, this is a network consisting of 7 layers, 2 of which are input and output layers (Figure 1a). This network has been trained with HMBC and HSQC separately. The same structure is also used in the combined neural network (Figure 1b). This combined CNN architecture consists of 2 initial branches, forming independent CNNs, which are then connected to a dense layer and followed by an output layer. Networks are implemented with Python 3 using the Keras and Tensorflow libraries [18, 19].

Furthermore, this study assumes a closed-world hypothesis, where all compounds fall into exactly one category (i.e. are of exactly one superclass). The network will decide on one class due to the Softmax activation function


1131x791x3
INPUT
Conv2D
32
Conv2D
64
Conv2D
64
Conv2D
128
Dense
64
Softmax
Alkaloids and derivatives
Benzenoids
Lipids and lipid-like molecules
Nucleosides, nucleotides, and analogues
Organic acids and derivatives
Organic nitrogen compounds
Organic oxygen compounds
Organoheterocyclic compounds
Phenylpropanoids and polyketides
9
1131x791x3
HMBC INPUT
Conv2D
32
Conv2D
64
Conv2D
64
Conv2D
128
1131x791x3
HSQC INPUT
Conv2D
32
Conv2D
64
Conv2D
64
Conv2D
128
Dense
64
Softmax
Alkaloids and derivatives
Benzenoids
Lipids and lipid-like molecules
Nucleosides, nucleotides, and analogues
Organic acids and derivatives
Organic nitrogen compounds
Organic oxygen compounds
Organoheterocyclic compounds
Phenylpropanoids and polyketides
9

(a)

(b)

Figure 1: Architecture of (a) the single spectrum convolutional neural network and (b) the convolutional neural network for the combined HMBC and HSQC spectra (from [1]).

in the output layer. Working under an open-world assumption is an open problem in AI research and not part of this pilot study.

Note that this is a proof-of-concept study with a relatively low number of examples. Therefore, we have not separated a validation set from the main data set. The authors are using a fixed test set rather than cross-validation, which was performed for the first class without returning significantly different results. The test and training set are stratified, i. e. there each class is split into test and training set and both contain the classes in the same proportions. The authors also did not optimize the network by tuning its architecture and parameters. Furthermore, we did not try to tune the image export. The images were exported as PNG files with a size of 1133x791 pixels; no further image manipulations were applied. In this sense, the numbers reported in Section 2.4 are a baseline. They represent the average result of ten runs of the training process with the fixed test set. Ten runs were chosen to iron out minor differences between runs. We trained for 20 epochs for the single networks and for 30 for the combined network since the networks converge somewhere beyond 10 and, respectively, 20 epochs. The main metric used is the accuracy, since this is most appropriate for a classification problem. Additionally, we have calculated the F1 score, recall, precision, and Matthews correlation coefficient.

In addition to CNNs, we have also tried Capsule Neural Networks (CapsNet). These were pioneered by Geoff Hinton et al. in [20] and promise better image processing than CNNs. We have tried the implementations available at `https://colab.research.google.com/drive/1WiqyF7dCdnNBIANEY80Pxw_`

mVz4fyV-S?usp=sharing and https://towardsdatascience.com/implementing-capsule-netwo
Both deal with the MNIST dataset in the original implementation, which contains images of size 28 × 28 pixels. The authors found that dealing with images of our size is impossible for these implementations, even on a computer with large memory. Therefore, we have decided not to use CapsNet. This is also justified because a major advantage here is the ability of the NMR spectra to deal with the rotation of elements, which is not a common distortion in NMR spectra.

5. Conclusion

We have shown that CNNs can extract chemical class information from NMR spectra. This is in contrast to other image processing methods, which are unable to do so. The accuracies achieved are intended only to show a proof-of-concept, not as a final result. Improving the network is a task for future research. In particular, the combination of HMBC and HSQC spectra needs attention, since it should get at least as good a result as the best single input result. Another potential research route is the inclusion of more information, e. g. from mass spectrometry.

Acknowledgement

S.K acknowledges funding by De Montfort University for computational facilities (VC2020 new staff L SL 2020). C.C and A.B thank Xunta de Galicia for funding the Mestrelab Research Center (CIM), subsidized by the Galician Innovation Agency, through the business aid program for the creation and integration of new business research centers 001_IN853D_2022.

Data statement

Data and code are available at https://github.com/stefhk3/nmrchemclassify.

References

[1] S. Kuhn, E. Tumer, S. Colreavy-Donnelly, R. Moreira Borges, A pilot study for fragment identification using 2d nmr and deep learning, Magnetic Resonance in Chemistry 60 (11) (2022) 1052–1060. arXiv:https://analyticalsciencejournals.onlinelibrary.

wiley.com/doi/pdf/10.1002/mrc.5212, doi:https://doi.org/10.1002/mrc.5212.
URL https://analyticalsciencejournals.onlinelibrary.wiley.com/doi/abs/10.1002/mrc.5212

[2] J. J. J. van Der Hooft, J. Wandy, M. P. Barrett, K. E. Burgess, S. Rogers, Topic modeling for untargeted substructure exploration in metabolomics, Proceedings of the National Academy of Sciences 113 (48) (2016) 13738–13743.

[3] M. Ernst, K. B. Kang, A. M. Caraballo-Rodríguez, L.-F. Nothias, J. Wandy, C. Chen, M. Wang, S. Rogers, M. H. Medema, P. C. Dorrestein, J. J. van der Hooft, Molnetenhancer: Enhanced molecular networks by integrating metabolome mining and annotation tools, Metabolites 9 (7) (2019). doi:10.3390/metabo9070144.
URL https://www.mdpi.com/2218-1989/9/7/144

[4] L.-F. Nothias, D. Petras, R. Schmid, K. Dührkop, J. Rainer, A. Sarvepalli, I. Protsyuk, M. Ernst, H. Tsugawa, M. Fleischauer, F. Aicheler, A. A. Aksenov, O. Alka, P.-M. Allard, A. Barsch, X. Cachet, A. M. Caraballo-Rodriguez, R. R. Da Silva, T. Dang, N. Garg, J. M. Gauglitz, A. Gurevich, G. Isaac, A. K. Jarmusch, Z. Kameník, K. B. Kang, N. Kessler, I. Koester, A. Korf, A. Le Gouellec, M. Ludwig, C. Martin H., L.-I. McCall, J. McSayles, S. W. Meyer, H. Mohimani, M. Morsy, O. Moyne, S. Neumann, H. Neuweger, N. H. Nguyen, M. Nothias-Esposito, J. Paolini, V. V. Phelan, T. Pluskal, R. A. Quinn, S. Rogers, B. Shrestha, A. Tripathi, J. J. J. van der Hooft, F. Vargas, K. C. Weldon, M. Witting, H. Yang, Z. Zhang, F. Zubeil, O. Kohlbacher, S. Böcker, T. Alexandrov, N. Bandeira, M. Wang, P. C. Dorrestein, Feature-based molecular networking in the gnps analysis environment, Nature Methods 17 (9) (2020) 905–908. doi:10.1038/s41592-020-0933-6.
URL https://doi.org/10.1038/s41592-020-0933-6

[5] R. C. Gonzalez, R. E. Woods, Digital image processing, Pearson, 2018.

[6] P. M. Kozlowski, Y. Kim, B. M. Haines, H. F. Robey, T. J. Murphy, H. M. Johns, T. S. Perry, Use of computer vision for analysis of image datasets from high temperature plasma experiments, Rev Sci Instrum 92 (3) (2021) 033532.

[7] S. Kuhn, N. E. Schlörer, H. Kolshorn, R. Stoll, From chemical shift data through prediction to assignment and nmr lims - multiple functionalities of nmrshiftdb2, Journal of Cheminformatics 4 (1) (2012) P52. doi:10.1186/1758-2946-4-S1-P52.
URL http://dx.doi.org/10.1186/1758-2946-4-S1-P52

[8] E. L. Ulrich, H. Akutsu, J. F. Doreleijers, Y. Harano, Y. E. Ioannidis, J. Lin, M. Livny, S. Mading, D. Maziuk, Z. Miller, E. Nakatani, C. F. Schulte, D. E. Tolmie, R. Kent Wenger, H. Yao, J. L. Markley, BioMagResBank, Nucleic Acids Research 36 (suppl 1) (2007) D402–D408. arXiv:https://academic.oup.com/nar/article-pdf/36/suppl_1/D402/7635401/gkm957.pdf, doi:10.1093/nar/gkm957.
URL https://doi.org/10.1093/nar/gkm957

[9] Y. Djoumbou Feunang, R. Eisner, C. Knox, L. Chepelev, J. Hastings, G. Owen, E. Fahy, C. Steinbeck, S. Subramanian, E. Bolton, R. Greiner, D. S. Wishart, Classyfire: automated chemical classification with a comprehensive, computable taxonomy, Journal of Cheminformatics 8 (1) (2016) 61. doi:10.1186/s13321-016-0174-y.
URL https://doi.org/10.1186/s13321-016-0174-y

[10] S. Mestrelab Research S.L:, Santiago de Compostela, Mnova (mestrenova) (version 14.3.1).
URL https://www.mestrelab.com

[11] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, Mobilenetv2: Inverted residuals and linear bottlenecks (2018). doi:10.48550/ARXIV.1801.04381.
URL https://arxiv.org/abs/1801.04381

[12] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, Orb: An efficient alternative to sift or surf, in: 2011 International Conference on Computer Vision, 2011, pp. 2564–2571. doi:10.1109/ICCV.2011.6126544.

[13] Z. Wang, A. Bovik, H. Sheikh, E. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Transactions on Image Processing 13 (4) (2004) 600–612. doi:10.1109/TIP.2003.819861.

[14] G. Balakrishnan, A. Zhao, M. R. Sabuncu, A. V. Dalca, J. Guttag, An unsupervised learning model for deformable medical image registration, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern

Recognition, IEEE, 2018, pp. 9252–9260. doi:10.1109/cvpr.2018.00964.
URL https://doi.org/10.1109%2Fcvpr.2018.00964

[15] 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, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[16] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, L. Farhan, Review of deep learning: concepts, cnn architectures, challenges, applications, future directions, Journal of Big Data 8 (1) (2021) 53. doi:10.1186/s40537-021-00444-8.
URL https://doi.org/10.1186/s40537-021-00444-8

[17] C. Li, Y. Cong, W. Deng, Identifying molecular functional groups of organic compounds by deep learning of nmr data, Magnetic Resonance in Chemistry 60 (11) (2022) 1061–1069. arXiv:https://analyticalsciencejournals.onlinelibrary.wiley.com/doi/pdf/10.1002/mrc.5292, doi:https://doi.org/10.1002/mrc.5292.
URL https://analyticalsciencejournals.onlinelibrary.wiley.com/doi/abs/10.1002/mrc.5292

[18] F. Chollet, et al., Keras (2015).
URL https://github.com/fchollet/keras

[19] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: Large-scale machine learning on heterogeneous systems, software available from tensorflow.org (2015).
URL http://tensorflow.org/

[20] S. Sabour, N. Frosst, G. E. Hinton, Dynamic routing between capsules (2017). doi:10.48550/ARXIV.1710.09829.
URL https://arxiv.org/abs/1710.09829