Molecular Identification from AFM images using the IUPAC Nomenclature and Attribute Multimodal Recurrent Neural Networks

Jaime Carracedo-Cosme,[1,2] Carlos Romero-Muñiz,[3]
Pablo Pou,[2,4] Rubén Pérez,[2,4,*]

[1]Quasar Science Resources S.L., Camino de las Ceudas 2, E-28232 Las Rozas de Madrid, Spain
[2]Departamento de Física Teórica de la Materia Condensada,
Universidad Autónoma de Madrid, E-28049 Spain
[3]Departamento de Física Aplicada I, Universidad de Sevilla, E-41012, Seville, Spain
[4]Condensed Matter Physics Center (IFIMAC),
Universidad Autónoma de Madrid, E-28049 Madrid, Spain

[*] ruben.perez@uam.es

May 1, 2022

Despite being the main tool to visualize molecules at the atomic scale, Atomic Force Microscopy (AFM) with CO-functionalized metal tips is unable to chemically identify the observed molecules. Here we present a strategy to address this challenging task using deep learning techniques. Instead of identifying a finite number of molecules following a traditional classification approach, we define the molecular identification as an image captioning problem. We design an architecture, composed of two multimodal recurrent neural networks, capable of identifying the structure and composition of an unknown molecule using a 3D-AFM image stack as input. The neural network is trained to pro-

vide the name of each molecule according to the IUPAC nomenclature rules. To train and test this algorithm we use the novel QUAM-AFM dataset, which contains almost 700,000 molecules and 165 million AFM images. The accuracy of the predictions is remarkable, achieving a high score quantified by the cumulative BLEU 4-gram, a common metric in language recognition studies.

1 Introduction

Scanning Probe Microscopes have played a key role in the development of nanoscience as the fundamental tools for the local characterization and manipulation of matter with high spatial resolution. In particular, AFM operated in its frequency modulation mode allows the characterization and manipulation of all kind of materials at the atomic scale (*1, 2, 3*). This is achieved measuring the change in the frequency of an oscillating tip due to its interaction with the sample. When the tip apex is functionalised with inert closed-shell atoms or molecules, particularly with a CO molecule, the resolution is dramatically enhanced, providing access to the inner structure of molecules (*4*). This outstanding contrast arises from the Pauli repulsion between the CO probe and the sample molecule (*4, 5*) modified by the electrostatic interaction between the potential created by the sample and the charge distribution associated with the oxygen lone pair at the probe (*6, 7, 8*). In addition, the flexibility of the molecular probe enhances the saddle lines of the total potential energy surface sensed by the CO (*9*). These high resolution AFM (HR-AFM) capabilities have made possible to visualize frontier orbitals (*10*), to determine bond order potentials (*11*) and charge distributions (*12, 13*), and have opened the door to track and control on-surface chemical reactions (*14, 15*).

In spite of these impressive achievements (*16, 17*) one of the most important goals remains elusive: the molecular recognition. That is, the ability of naming a certain molecule exclusively by means of HR-AFM observations. Molecules have been identified combining AFM with other experimental techniques like scanning tunneling microscopy (STM) or Kelvin probe force microscopy (KPFM), and with the support of theoretical simulations (*10, 17, 18, 19, 20, 21*). Chemical identification by AFM of individual atoms at semiconductor surface alloys was achieved using reactive semiconductor apexes (*22*). In that case, the maximum attractive force between the tip apex and the probed atom on the sample carries information of the chemical species

involved in the covalent interaction. However, the scenario is rather different when using tips functionalised with the inert CO molecules where the main AFM contrast source is the Pauli repulsion and the images are strongly affected by the probe relaxation. So far, the few attempts to discriminate atoms in molecules by HR-AFM have been based either on differences found in the tip-sample interaction decay at the molecular sites (*6, 23*) or on characteristic image features associated with the chemical properties of certain molecular components (*6, 17, 24, 21, 25, 26, 10, 27, 28*). For instance, sharper vertices are displayed for substitutional N atoms on hydrocarbon aromatic rings (*24, 6, 23*) due to their lone pair. Furthermore, the decay of the CO-sample interaction over those substitutional N atoms is faster than over their neighboring C atoms (*6, 23*). Halogen atoms can also be distinguished in AFM images thanks to their oval shape (associated to their σ-hole (*25*)) and to the significantly stronger repulsion compared to atoms like nitrogen or carbon (*25*). However, even these atomic features depend significantly on the molecular structure (*6, 11*) and cannot be only associated to a certain species but to its moiety in the molecule. The huge variety of possible chemical environments renders the molecular identification by a mere visual inspection by human eyes an impossible task.

Artificial Intelligence (AI) techniques are precisely optimized to deal with this kind of subtle correlations and massive data. Deep learning (DL), with its outstanding ability to search for patterns, is nowadays routinely used to classify, interpret, describe and analyze images (*29, 30, 31, 32, 33, 34*), providing machines with capabilities hitherto unique to human beings or even surpassing them in some tasks (*35*). There are two main challenges to apply deep learning to achieve a complete molecular identification (structure and composition) through AFM imaging. The first one is the limited amount of experimental data available to train the models. In a previous work (*36*), we have explored the performance of an specifically designed Convolutional Neural Network (CNN), trained with a data set that includes 314,460 theoretical images –calculated with the latest HR-AFM modeling approaches (*6, 37*)– and only 540 images gener-

ated with a variational autoencoder from very few experimental images. This CNN, applied to a set of 60 molecular structures that include 10 different atomic species (C, H, N, P, O, S, F, Cl, Br, I), obtained almost perfect (99%) accuracy in the classification using simulated AFM images and very good accuracy (86%) for experimental AFM images. Encouraged by the success of this proof–of–concept, we have recently extended the available data sets of theoretical AFM images with the generation of QUAM–AFM (*38*), that aims to provide a solid basis for making results from DL applications to the AFM field reliable and reproducible (*39*). QUAM–AFM includes calculations for a collection of 686,000 molecules using 240 different combinations of AFM operation parameters (tip–molecule distance, cantilever oscillation amplitude and tilting stiffness of the CO-metal bond), resulting in a total of 165 million images (*38*).

The second challenge arises from the non–planar structure of the molecules, that mixes up in the molecular charge density –ultimately responsible for the AFM contrast– the effects of the geometry and the chemical composition, making it very difficult to disentangle them. Alldritt *et al.* (*40*) developed a CNN focused on the task of determining the molecular geometry. Results were excellent for the structure of quasi-planar molecules, even using the algorithm directly with experimental images. For 3D structures, they were able to recover information for the positions of the atoms closer to the tip. However, the discrimination of functional groups produced non conclusive results. At variance with this study, as we already mentioned above, a CNN (*36*) was able to solve the classification problem for 60 essentially flat molecules with almost perfect accuracy, being able to identify, for example, the presence of a particular halogen (F, Cl, Br or I) in molecular structures that, apart from this atom, were identical. Although encouraging, the clear success of this proof of concept does not provide a solution to the general problem of molecular identification. The classification approach can only identify molecules included in the training data set. Given the rich complexity provided by organic chemistry, even an extremely large data set, that already poses fantastic computational challenges (as the output

vector has the dimension of the number of molecules in the dataset), would fail to classify many of the already known or possibly synthesized molecules of interest.

In this work, we transform the problem of molecular identification into an image captioning challenge: the description of the content of an image using language. Automatic image captioning has been a field of intensive research for deep learning techniques over the last years (*41, 42, 43, 44*). It has been recently and successfully used (*45, 46*) for optical chemical structure recognition (*47*), the translation of graphical molecular depictions into machine-readable formats. These works are able to predict the SMILES textual representation (*48*) of a molecule from an image with its chemical structure depiction by using standard encoder-decoder (*46*) or transformer (*45*) models. In our case, we consider a stack of 10 constant-height HR-AFM images, each corresponding to different tip–sample distances, as the "image" and the International Union of Pure and Applied Chemistry (IUPAC) name of the molecule as the description or caption. Most of the current methods for automatic image captioning have two key components: (i) a CNN –a Neural Network (NN) with convolutional kernels as processing units– that represents the high-level features of the input images in a reduced dimensional space; and (ii) a Recurrent Neural Network or Elman network (*49*) (RNN) –a NN whose units are complex structures that have an inner state that stores the temporal context of a time series– that deals with language processing and predicts a single word at each time step (*50, 51, 52*). The IUPAC name determines unambiguously the molecular composition and structure. This is done by defining a hierarchical keyword list to name functional groups that are written following a systematic syntax that defines the structural position of each moiety or group in the molecule (*53*). Therefore, we tackle the molecular recognition challenge with a deep learning architecture that decomposes into two models. The first one predicts the main chemical groups that compose the molecule whereas the second model performs the IUPAC formulation. QUAM–AFM (*38*), is used to train and test the networks. Our approach predicts the exact name

in almost half of the cases and achieves a high accuracy according to the Bilingual Evaluation Understudy (BLEU) algorithm (*54*), the most commonly applied metric to score the accuracy of language-involved models.

2 A Deep Learning Approach for Molecular Identification

2.1 QUAM-AFM: Structures and AFM Simulations

One of the main challenges to automate the molecular identification through AFM imaging arise from the limited availability of data to fit the parameters of deep learning models. We use Quasar Science Resources S.L. - Universidad Autónoma de Madrid - Atomic Force Microscopy (QUAM-AFM) (*38*), a dataset of 165 million AFM images theoretically generated from 686,000 isolated molecules. Although the general operation of the HR-AFM is common to all instruments, operational parameter settings (cantilever oscillation amplitude, tip–sample distance, CO tilt stiffness) lead to variations in the contrast observed on the resulting images. The value of the first two can be adjusted by modifying the microscope settings to enhance different features of the image. However, the latter depends on the nature of the tip, i.e. the differences in the attachment of the CO molecule to the metal tip that have been consistently observed and characterised in experiments (*37, 55*). In order to cover the widest range of variants in the AFM images, six different values for the cantilever oscillation amplitude, four for the tilt stiffness of the CO molecule and 10 tip-sample distances were used to generate QUAM-AFM, resulting in a total of 240 simulations from each structure. We use the stack of 10 images resulting from the different tip–sample distances in a single input and the 24 parameter combinations as a data augmentation technique. That is, we feed the network with different image stacks randomly selected from the combinations of simulation parameters in each of the epochs for each of the molecules.

2.2 IUPAC Tokenization

Deep learning has already proven to have an extraordinary capacity to analyse data. This capacity is such that, in many cases, the biggest problem to be solved lies in defining an appropriate descriptor rather than in improving the existing analysis capacity. This is the case for AFM images, where the complexity to design the output of a model is due to the existence of infinite molecular structures. To establish a model output that is unambiguous, uniform and consistent for the terminology of chemical compounds, we have adopted the IUPAC nomenclature. Then, we have turned the standard classification problem (*36*) for a finite number of molecular structures into an image captioning task, developing a model that manages to formulate the IUPAC name of each molecule.

Most image captioning techniques to describe images through language consist of a loop that predicts a new word at each iteration (time step). Our goal is to transfer this idea to the identification of AFM images through the IUPAC formulation. Therefore, instead of predicting words at each time step, our model has to predict segments of the molecule's IUPAC name (see fig. 1). That is, the set of tokens used to decompose each name are sets of letters, numbers and symbols that we call *terms* and are used by the IUPAC nomenclature to denote functional groups, to assemble additive names or to specify connections. Different combinations of these terms generate IUPAC names for the molecules, as exemplified in fig. 1.

A systematic split of the IUPAC names in QUAM-AFM reveals that some of the terms have a very small representation, not enough to train a NN. We have discarded those that are repeated less than 100 times in QUAM-AFM, retaining a total of 199 terms (see table S1). Consequently, we have also removed the molecules that have any of these terms in their IUPAC name. In addition, we have dropped the molecules whose term decomposition has a length longer than 57, as there is not enough representation of such names in QUAM-AFM. Even so, the set of annotations still contains 678,000 molecules, that we have split into training, validation and test

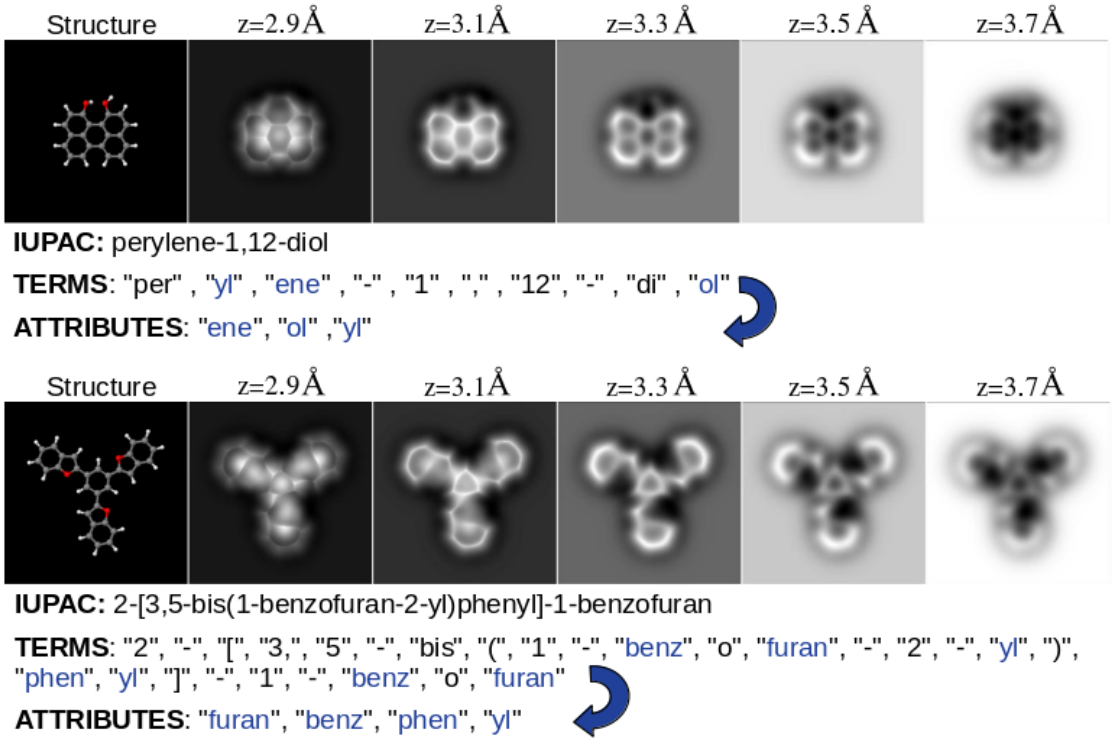


Figure 1: Two molecular structures with five of their associated AFM images at different tip-sample distances, the IUPAC name, the term decomposition of that name, and the associated attributes (a selection of 100 IUPAC terms that represent common functional groups, or chemical moieties, see text). The top structure shows that the attributes are sorted by length and alphabetically, not by the position in which they appear in the term decomposition. The bottom structure shows that attributes appear once even if they are repeated in the term decomposition.

subsets with 620,000, 24,000 and 34,000 structures, respectively.

Our first attempts based on feeding a single model with a stack of AFM images provides poor results predicting the IUPAC nomenclature. For this reason, we decompose the problem into two parts and assign each objective to a different NN (see figs. 1 and 2 and section 2.3 for a detailed description). We define the *attributes* as a 100-element subset of the IUPAC terms (see table S1) which mainly describes the most common functional groups in organic chemistry and, thus, are repeated a minimum number of times. The first network, named Multimodal Recurrent

Neural Network for attribute prediction (M-RNN$_A$), uses as input the stack of AFM images and its aim is to extract the attributes, predicting the main functional groups of the molecule (see figs. 1 and 2). The second network, named Atribute Multimodal Recurrent Neural Network (AM-RNN), takes as inputs both the AFM image stack and the attribute list with the aim of ordering them and complete the whole IUPAC name of the molecule with the remaining terms which are not considered attributes (see fig. 2B).

M-RNN$_A$ reports information neither on the order nor the number of times that the *attribute* appears in the formulation. However, this first prediction plays a key role in the performance of the model. Unlike most of the Natural Languaje Processing (NLP) challenges, the IUPAC name completely identifies the structure and composition of the molecule. Thus, a prior identification of the main functional groups, not only releases the CNN component of the AM-RNN from the goal of identifying these moieties, but, more importantly, almost halves the number of possible predictions of the AM-RNN. By feeding the AM-RNN with the attributes that are present in the IUPAC name (predicted by the M-RNN$_A$), we are also effectively excluding the large number of them that do not form part of it. This is an extremely simple relationship that the network learns and that improves significantly its performance.

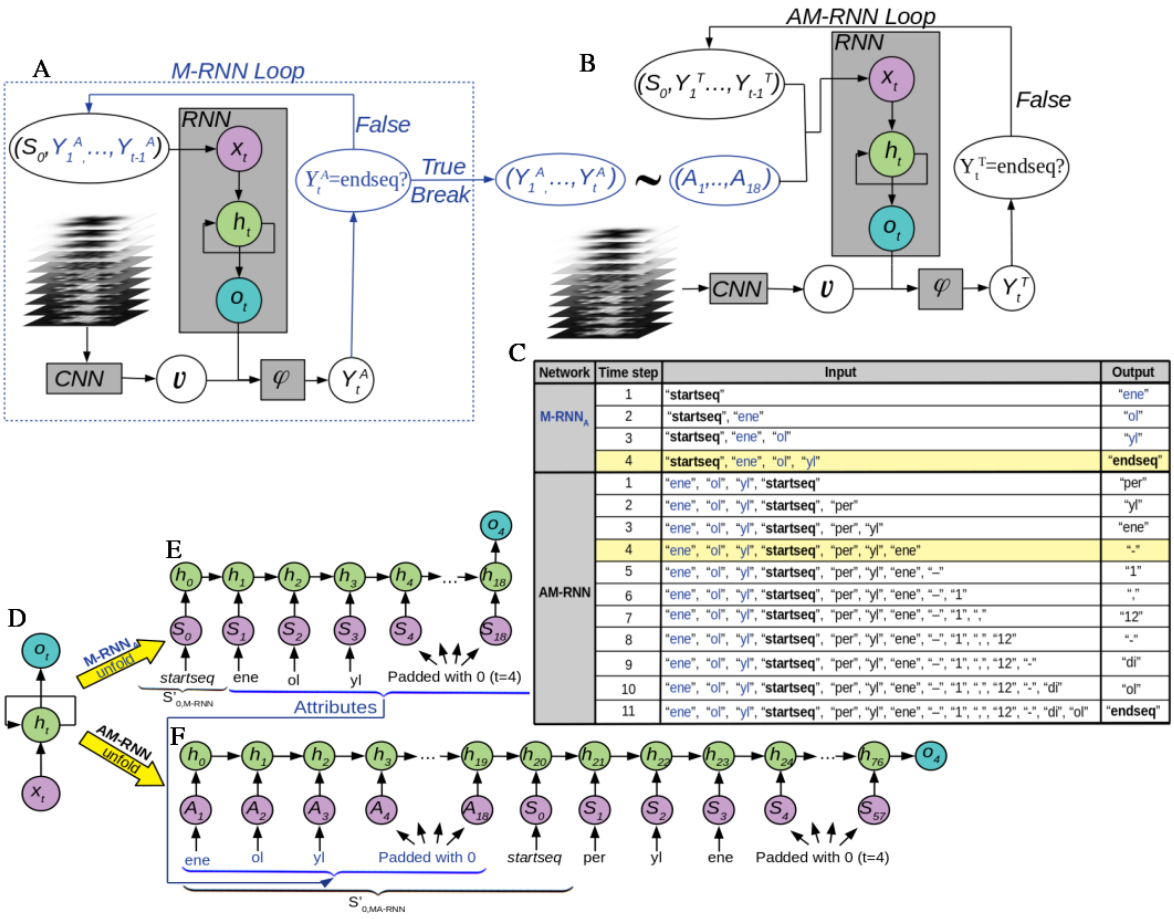
2.3 Multimodal and Attribute Multimodal Recurrent Neural Networks (M-RNN$_A$ and AM–RNN)

The standard approach for image captioning is based on an architecture that integrates a CNN and a RNN (*41, 56*). Here, we focus on the well–known Multimodal Recurrent Neural Network (M-RNN), which integrates three components (see fig. S1). The CNN encodes the input image into a high–level feature vector whereas the RNN component has two key objectives: Firstly, to embed a representation of each word based on its semantic meaning and, secondly, to store the semantic temporal context in the recurrent layers. The remaining component is the multimodal

(φ) component, which is in charge of processing both CNN and RNN outputs and generating the output of the model.

As discussed in section 2.2, we have developed an architecture composed of two M-RNNs (see fig. 2A and fig. 2B). The first one, the M-RNN$_A$, predicts the *attributes* that are incorporated as input to the second one, the AM-RNN, which performs the IUPAC name prediction. Although both AM-RNN and M-RNN$_A$ are based on the standard M-RNN (*41*), we introduce substantial modifications in each component. In fig. 2A and 2B, we show the inputs for each component. The input of the CNN component is a stack of 10 AFM images, whereas the input of the multimodal component φ consists of a concatenation of the outputs of the CNN and RNN components.

To explicitly define the inputs of the M-RNN components, it is worth recalling that a M-RNN processes time series, so it will perform a prediction (*attribute* or *term*) at each time step. Let us start by defining the inputs of the RNN component of the M-RNN$_A$. We encode the *attributes* of the model by assigning integer numbers (from 1 to 100) to each *attribute*. The input of RNN is a vector of fixed size 19, the maximum number of different attributes in the names of the molecules in QUAM-AFM (17) plus the *startseq* and *endseq* tokens. In the first step, it will contain $S_{0,\text{M-RNN}_A}$ = *startseq* to provide the model with the information that a new prediction starts. This input is padded with zeros until we obtain a length of 19 (see figs. 2A, 2C and 2E) and then processed by the RNN component while the stack of AFM images are processed by the CNN component, each of them encoding the respective input into a vector. The two resulting vectors are used to feed the multimodal component φ, where they are concatenated and processed in a series of fully connected layers to finally produce a vector of probabilities (see Figs. S1 and S2 for details on the RNN and φ layers). In this way the prediction at each time step corresponds to the most likely *attribute* Y_1^A which replaces the padding zero of the corresponding position in the input sequence of the RNN component in the next time step. This



Network	Time step	Input	Output
M-RNN$_A$	1	"**startseq**"	"ene"
	2	"**startseq**", "ene"	"ol"
	3	"**startseq**", "ene", "ol"	"yl"
	4	"**startseq**", "ene", "ol", "yl"	"**endseq**"
AM-RNN	1	"ene", "ol", "yl", "**startseq**"	"per"
	2	"ene", "ol", "yl", "**startseq**", "per"	"yl"
	3	"ene", "ol", "yl", "**startseq**", "per", "yl"	"ene"
	4	"ene", "ol", "yl", "**startseq**", "per", "yl", "ene"	"-"
	5	"ene", "ol", "yl", "**startseq**", "per", "yl", "ene", "–"	"1"
	6	"ene", "ol", "yl", "**startseq**", "per", "yl", "ene", "–", "1"	","
	7	"ene", "ol", "yl", "**startseq**", "per", "yl", "ene", "–", "1", ","	"12"
	8	"ene", "ol", "yl", "**startseq**", "per", "yl", "ene", "–", "1", ",", "12"	"-"
	9	"ene", "ol", "yl", "**startseq**", "per", "yl", "ene", "–", "1", ",", "12", "-"	"di"
	10	"ene", "ol", "yl", "**startseq**", "per", "yl", "ene", "–", "1", ",", "12", "-", "di"	"ol"
	11	"ene", "ol", "yl", "**startseq**", "per", "yl", "ene", "–", "1", ",", "12", "-", "di", "ol"	"**endseq**"

Figure 2: The architecture proposed for molecular identification through the IUPAC name is a composition of two networks (M-RNN$_A$ and AM-RNN) whose data flow is shown in panels A and B. The gray square boxes represent each component of the models: a convolutional neural network CNN, a recurrent neural network RNN, and the multimodal component φ. The arrows indicate the data flow within the model. M-RNN$_A$ predicts an attribute at each time step until the loop is broken with the *endseq* token (blue-line printed), whereas the AM-RNN predicts the sorted terms that give rise to the IUPAC name. Panel C shows the inputs and outputs at each time step predicted by the M-RNN$_A$ and AM-RNN networks from a 3D image stack corresponding to the perylene-1,12-diol molecule. Panels D, E and F show a representation of the RNN, in the same format used panels A and B, corresponding to the fourth time step in M-RNN$_A$ and AM-RNN for the perylene-1,12-diol molecule. This figure highlights the fact that the state of the RNN, in particular, the recurrent layer, depends on the previous predictions.

process is repeated until the *endseq* token is predicted, which breaks the loop. That is, for a given time step t, we feed the RNN component of M-RNN$_A$ with the input $(S_0, Y_1^A, ..., Y_{t-1}^A)$, that concatenates the *starseq* token S_0 with all the predictions already performed in previous time steps, and is padded with zeros until we obtain a length of 19 (see fig. 2E for the example of $t = 4$ in the indentification of perylene-1,12-diol molecule). Once the model has already predicted the N_A *attributes* it has to break the loop, so its last prediction must be the *endseq* token (see fig. 2C).

Once the prediction of the *attributes* has finished, the AM-RNN starts to operate in order to predict the IUPAC name of the molecule. For the input of the RNN component and the prediction flow, we follow the same reasoning applied to M-RNN$_A$, replacing $S_{0,\text{M-RNN}_A}$ by $S_{0,\text{AM-RNN}} = (Y_1^A, .., Y_{18}^A, \textit{startseq})$ (fig. 2B). Each RNN input is a vector of 76 components, arising from the concatenation of 18 *attributes* $(Y_1^A, .., Y_{18}^A) \sim (A_1, ..., A_{18})$ (padded if necessary) with the *startseq* token and the predictions performed at each previous time step, $(Y_1, ..., Y_{t-1})$, padded until we obtain a vector with length 57 –the maximum number of terms in the decomposition of the IUPAC names in QUAM-AFM– (see Figs. 2C and F). Similarly as in the M-RNN$_A$, the semantic input is processed by the RNN component while the AFM image stack is processed by the CNN, encoding the respective input into a vector. The multimodal component φ processes the CNN output v, concatenates the result with the output of the RNN, and process this combined result producing a vector of probabilities as output of the network. (see Figs. S1 and S2 for details). The position of the larger component in the vector provides us with the prediction of the new *term* Y_t. The process stops when the *endseq* token is predicted (see fig. 2C).

Denoting both $S_{0,\text{M-RNN}}$ and $S_{0,\text{AM-RNN}}$ by S_0', the data flow of both M-RNN and AM-RNN

is described by the following recurrence rules:

$$\begin{aligned} x_1 &= S_0' \\ v &= \text{CNN}(I) \\ h_t &= \text{RNN}(h_{t-1}, x_t) \\ Y_t &= \varphi(v, h_t),\ t \geq 1 \\ x_t &= (S_0', Y_1, ..., Y_{t-1}),\ t > 1 \end{aligned} \tag{1}$$

A more detailed description of each layer and the training strategy, far from trivial when combining a CNN and a RNN, can be found in sections S2 and S3, respectively.

3 Results

We have benchmarked the model by testing the trained networks with the 34,000 molecule test set, corresponding with 816,000 inputs from QUAM-AFM associated to variations of the simulation parameters. The predicted IUPAC names are identical to the annotations for 43% of the molecules. Taken into account the complexity of the problem, we can consider this as a good result. Notice that each matching means that the model has identified from the images, without any error, all the molecular moieties and it has also provided the exact IUPAC name, character by character, as shown in fig. 3. Our model is able to identify planar hydrocarbons, both cyclic or aliphatic, but also more complex structures as those including nitrogen or oxygen atoms that, due to their fast charge density decay (*6*), usually appear on the images as faint features (see for example fig. 3). Halogens, characterized on the images by oval features whose size and intensity are proportional to their σ–hole strength (*25*), can be also correctly labeled (figs. 3b, 3d and 3e). The model can even recognize the presence of the fluorine element, that does not induce a σ–hole and, when bonded to a carbon atom, produces an AFM fingerprint that is very similar to the one of a carbonyl group (compare fig. 3e with fig. 3f). More surprisingly, hydrogen positions are often guessed, what is striking since hydrogen atoms bonded to sp^2 carbon atoms are hardly

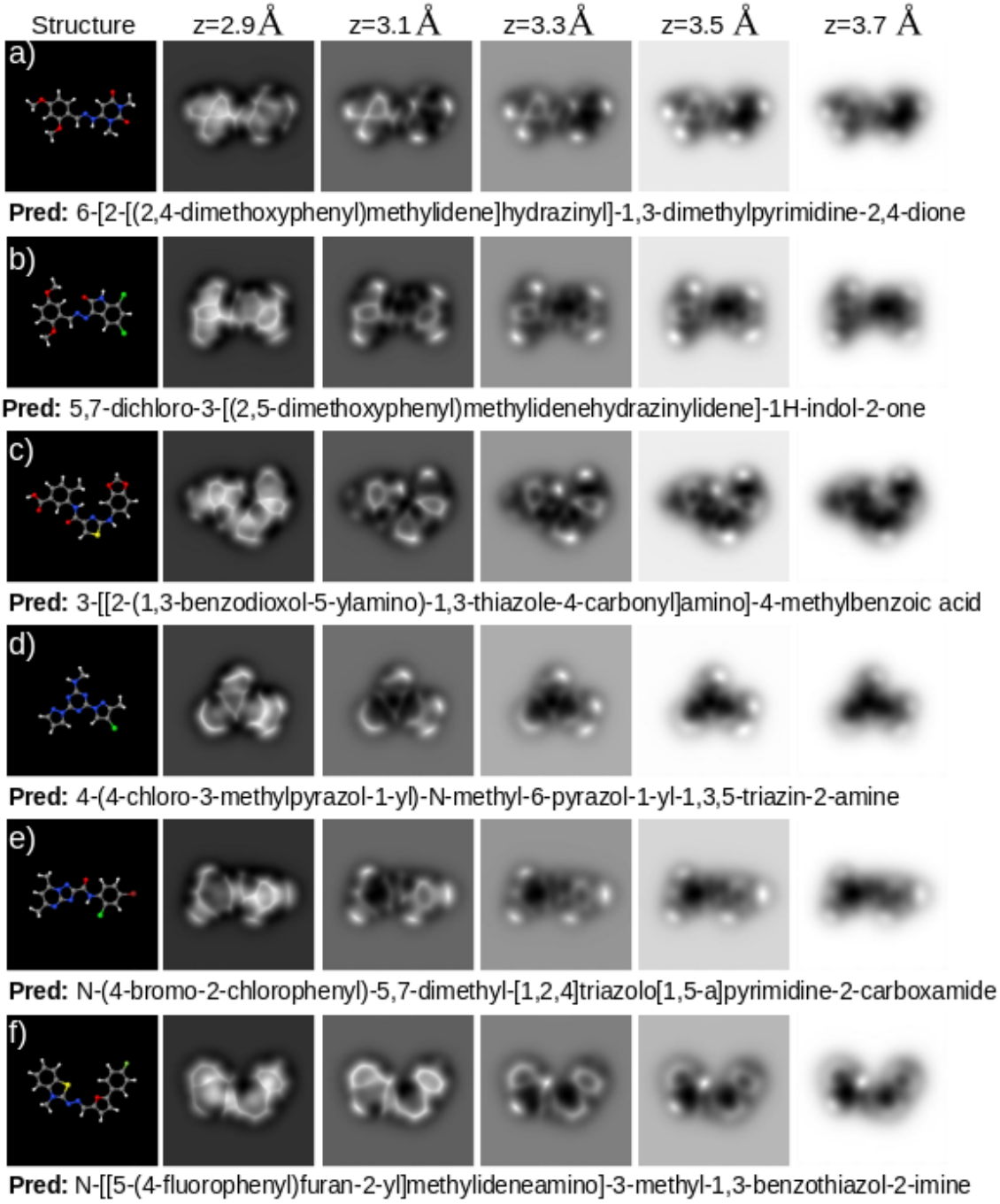


Figure 3: Set of perfect predictions (4-gram scores of 1.00). Each subfigure shows the molecular structure on the left, the AFM images at various tip–sample distances on the right and the prediction, which matches exactly the ground truth, below the images.

detected by the HR-AFM due to their negligible charge density (*28, 57*). Thus, many kinds of molecules, over half of our test set, including those showing non–trivial behaviors, have been correctly recognized by our model.

However, this statistic does not reflect the real accuracy of the model. A deeper analysis of the results shows that its quality and usefulness is much higher than the naked figure of 43% could indicate. Figure 4 shows that, even in those cases where the prediction is not correct, the majority of the examples still provide valuable information about the molecule. In order to quantify the accuracy of the prediction, we apply the n-grams of BLEU (*54*) (see fig. 4). This method, commonly used for assessing accuracy in NLP problems, calculates the accuracy based on *n–grams* of *terms* between predicted and reference sequences. An *n–gram* scores each prediction by comparing the sorted n–word groups appearing in the prediction with respect to the references. In our scenario, the comparison is with one single reference (ground truth), so it compares the common groups of *n terms* that appear in both the prediction and the reference (for example, perylene-1,12-diol, 4–gram reference groups include: "per, yl, ene, -", "yl, ene, -, 1","ene, -, 1, ,", etc).

First, we assess the accuracy of the M-RNN$_A$, the one that predicts the *attributes*, i.e., the molecular moieties. A perfect match on the 1–gram's means that every *attribute* in the reference appears in the prediction and that the prediction does not contain any other *attributes*. Our model scores a 0.95 under this assumption. This is a very high mark that means this network does recognize the molecular components on 95% of the cases. This result answers one of the more challenging open questions in the field (*10, 23, 6*): it demonstrates that the 3D HR-AFM data obtained with CO terminated apexes carries information of the chemical species present on the molecules, at least on the simulated images sets.

For the assessment of the overall prediction of the model, we propose the cumulative 4-gram, a common metric for the evaluation of linguistic predictions. This metric weights the

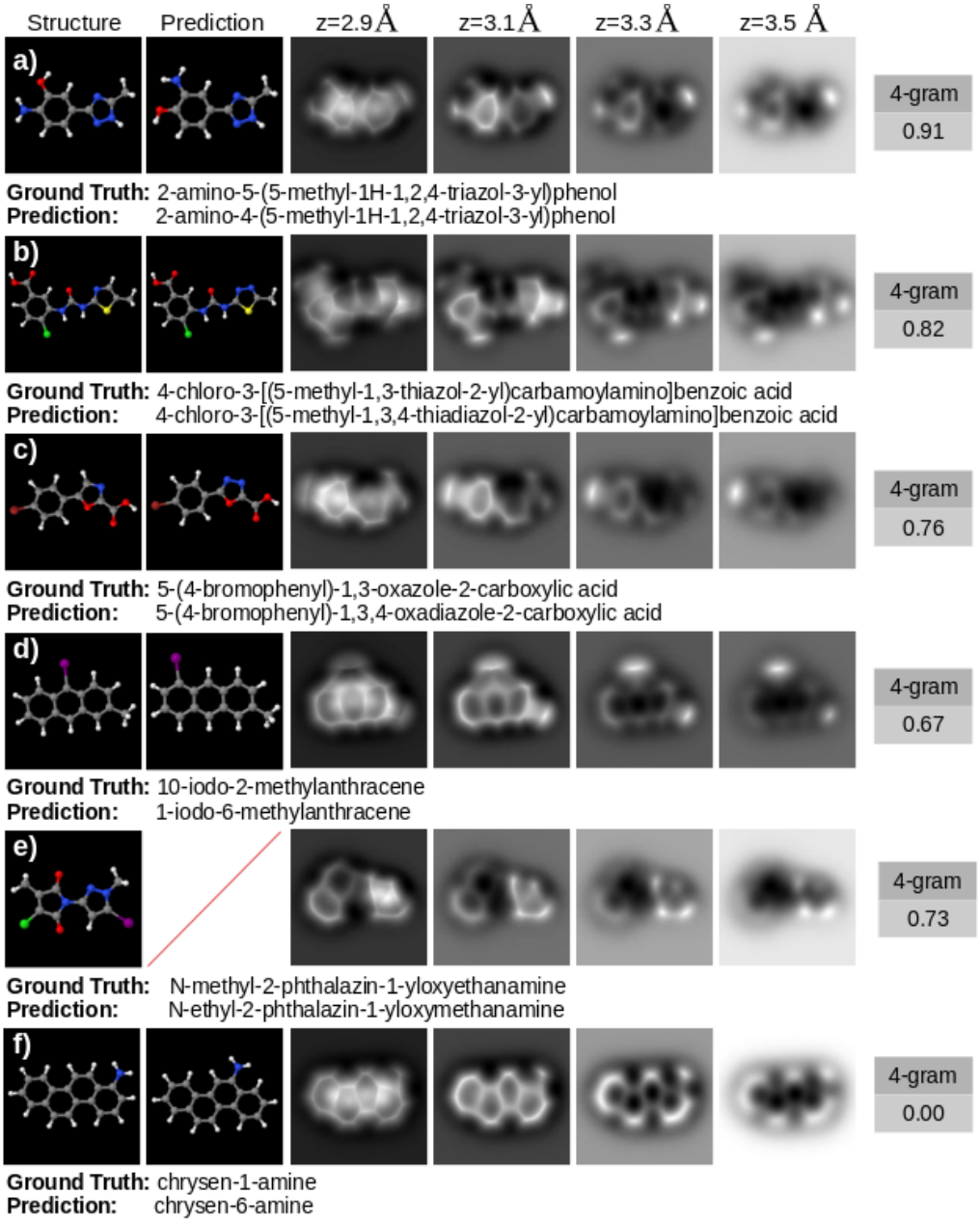


Figure 4: Examples of incorrect predictions, reflecting how the evaluation algorithm penalises errors. Each subfigure shows, from left to right, the simulated structure, the structure that matches the model prediction (where it exists), a set of AFM images at various tip–sample distances and the 4–gram score. Below each subfigure is the IUPAC name of the molecule (ground truth) and the prediction performed by the model (prediction).

Metric	1–gram	2–gram	3–gram	4–gram
Score	0.88	0.84	0.79	0.76

Table 1: BLEU cumulative n–gram scores obtained with AM-RNN applied to QUAM-AFM. The test has been performed on the set of 34,000 structures and their 24 combinations of simulation parameters.

scores obtained with the 1,2,3,4–grams and also performs a product with a function that penalises the different lengths between prediction and reference. BLEU scores (see table 1) reveal that AM-RNN also performs exceptionally well. Note that, in this case, the 1–gram shows the set of *terms* that are in both prediction and reference. That is, despite not providing the correct formulation, the model is able to predict 88% of the *terms* that the name contains, in agreement with the prediction capability showed by our first M-RNN$_A$, and indicating a great deal of chemical information about the molecule. In addition, AM-RNN scores 0.76 in the evaluation with the cumulative 4-gram, assessing large segments of the IUPAC name. Fig. 4 puts the accuracy of the model based on this assessment in context with a set of examples with different scores. Note that fig. 4f shows a frequently occurring case where, by applying a metric developed to assess translation in longer texts with several references, mistakes in predictions composed of only a few *terms* are overly penalised. Table 2 provides a systematic study of this limitation of the metric, showing an analysis of the score obtained by splitting the test set according to the number of terms into which the corresponding IUPAC name decomposes. The accuracy of the model is worse in molecules whose term decomposition is shorter. The reason for this seemingly contradictory fact is that the cumulative 4-gram metric penalises more for errors in short chains. As shorter strings contain fewer subgroups of 4 terms, the 4–gram scoring method penalises an error in a smaller chain more heavily than in a longer one (as shown in fig. 4a and fig. 4f).

Comparing the predictions with the references on a term–by–term basis, we find that the

25.1% of the errors are due to misclassification of one number term with another number, i.e. misplacing a group of atoms, and the 17.1%, 4.8%, 4.7% and 2.7% of the errors are due to a misclassification of the "-", "(" or ")", "yl" and "[" or "]" *terms*, respectively. Therefore, almost half of the errors made are located in the prediction of characters more related to the chemical formulation than to the information extracted from the images. Moreover, we must point out the fact that when the model predicts incorrectly, it sometimes generates IUPAC names that do not correspond to any molecule (see fig. 4e). These results indicate that it is not the capability of our model to recognise the molecules but the ability of the RNN component to properly write the name what is limiting the success rate. This conclusion is consistent with a recent work where automated IUPAC name translation from the SMILES nomenclature (*48*), that completely characterizes the structure and composition of a molecule, is done by a RNN (*58*), obtaining just a BLEU 4–gram score of 0.86.

Deep learning architectures are developed based on human intuition to improve the accuracy of the model. However, it is difficult to analyse in detail why the model succeeds or fails. When representing the input *terms* in the RNN component, we apply a word embedding that is trained with the rest of the model (see figs. S2 and S3). Previous research has shown that representations in this space capture the semantic meaning of words and establish algebraic relationships between them (*59, 60, 61*). It is truly remarkable to see that these results have been transferred to the formulation, grouping the *terms* according to their semantic meaning or according to the interactions described by the image stacks. We have verified this by projecting each of the *terms* into the 32–dimensional embedding space belonging to the AM-RNN, defining a L2 norm and computing the distances between the *terms*. These results show that *terms* with similar semantic meaning are close together (see fig. S9). For example, the closest *terms* to brom are chlor, fluor, and iod, or the *terms* closest to nona are octa, deca, undeca and dodeca. This also reflects in the fact that the *terms* that the model most commonly gets wrong are the closest ones, such as

Number of *terms*

Term Decomposition	0–10	10–20	20–30	30–40	40–50	50–60
4–gram score	0.59	0.73	0.78	0.79	0.75	0.66

Atom height difference

Distance	$<$0.5 Å	$<$1.0 Å	$<$1.5 Å	$\geq$1.5 Å
4–gram score	0.79	0.62	0.62	0.50

Table 2: Score with the BLEU cumulative 4–gram metric based on the characteristics of the molecules and their annotations. The top table divides the scores into subsets based on the length of the string of *terms* into which the IUPAC name is broken down. The bottom table divides the test set score into subsets based on the maximum difference in heights between atoms in the molecule (excluding hydrogens).

the errors in the prediction of the numbers that place atomic groups in specific positions (see figs. 4a, 4d and 4f), or the mistaken of one halogen for another. In other words, the erroneous terms have, in general, a similar semantic meaning.

Non–planar structures are a challenge for AFM-based molecular identification. Table 2 shows an analysis of the score obtained by splitting the test set according to different ranges of molecular torsion. In line with the conclusions reached in ref. (*40*), our model has a hard work to fully reveal the structure of molecules whose height difference between atoms exceeds 1.5 Å. This is an expected result as the microscope is highly sensitive to small variations on the probe–sample separation, and the interaction becomes highly repulsive on a distance range of 50–100 pm, inducing large CO tilting and image distortions. This makes very difficult to get a proper signal from lower atoms on molecules with non–planar configurations. However, most of these molecules would adopt a flatter configuration upon surface adsorption. We have tested our model by randomly selecting four of the non–planar structures whose prediction scores an arithmetic mean of 0.40 in the cumulative prediction of the 4–gram. We force them to acquire a flat structure and, then, we run the test again (see figs. S6 and S7). Prediction scores improve in a range of 0.2 to 0.55, resulting in a new mean cumulative 4–gram of 0.73. This

improvement represents semantically going from a prediction that barely provides any useful information about the molecule to one that in many cases gets it absolutely right. Thus, while the model already scores very high in the test with simulated images of gas–phase molecules, the performance would definitely improve with the flatter configurations expected for the adsorbed molecules measured in the experimental HR-AFM images.

4 Discussion

The results presented so far show that the stacks of 3D frequency shift images contain information not only on the structure of the molecules but also regarding their chemical composition. This information can be extracted by deep learning techniques, which, additionally, are able to provide the IUPAC name of the imaged molecules with a high success rate. Our combination of two M–RNNs is able to correctly recognize the molecule in many cases, even in those where it is difficult to discern between similar functional groups –as fluorine terminations with either carbonyl or even -H terminations–, or in image stacks where some moieties provide very subtle signals (see fig. 3). Some mistakes do appear from the chemical recognition point of view, especially in those molecules showing significant non-planar configurations where the performance is lower (see figs. S6 and S7 together with table 2). However, apart from these fundamental drawbacks, most of the errors in the predictions are related to the spelling of IUPAC names. That is, misplacement of functional groups or the incorrect use of parentheses, square brackets or hyphen characters, etc. It seems that these errors are frequent for RNNs dealing with the IUPAC nomenclature (*58*).

At this stage, it is worth considering if other DL architectures or alternative chemical nomenclatures could improve the molecular identification based on HR-AFM images. We have already pointed out that, leaving out the additional problem of extracting the chemical information from the images, a RNN only achieves a BLEU 4–gram score of 0.86 when translating from the

SMILES to the IUPAC name (*58*). Nomenclature translation has been addressed with architectures based on the novel transformer networks (*62*), obtaining a practically perfect accuracy (*63, 64*). Also, automatic recognition of molecular graphical depictions is able to correctly translate them to their SMILES representation with a 88% or 96% accuracy by using either a standard encoder-decoder (*46*) or a transformer (*45*) network. However, in our work we face a different problem since we deal with identification from AFM images instead of either molecular depictions, that contains all the chemical information needed to name a molecule, or translation between nomenclatures. Furthermore, the application of transformers to the identification from AFM images is not straightforward. First, tokenization must be consistent and each term must have a chemical meaning so that the embedding layers learn a meaningful information representation (see Fig. S9). This point has only been considered in ref. (*63*). More importantly, our method achieves high accuracy due to the initial attribute detection, forcing us to develop an architecture that is, in principle, incompatible with transformers, which are based on encoder-decoder networks.

Regarding other nomenclature systems for describing organic molecules, besides the already cited SMILES (*48*), there are other proposals such as InChI (*65*) or SELFIES (*66*), whose textual identifiers use the name of the atoms and bond connectivity. These systems miss relevant chemical information that is not provided by describing the molecule as a set of individual atoms rather than as moieties made up of atoms. Unlike these systems, the IUPAC nomenclature is focused on the classification of functional groups, an approach consistent with the characteristics shown by the AFM image features, that reflects in our proposal of a dual architecture composed by M-RNN$_A$ and AM-RNN. The SELFIES nomenclature establishes a robust representation of graphs with semantic constraints, solving some problems that arise in computer writing with other nomenclatures. However, the atom-based description would force an approach without attribute prediction, which is the key to obtain a high accuracy with our

model. Hence, it seems to be a trade-off between the limitations and improvements offered by these nomenclatures, suggesting that a dramatic improvement in performance is not expected, although further work is needed in order to reach a final conclusion.

Finally, we should point out that our analysis so far has been based on simulated HR-AFM images. In ref. (*36*), we showed that the experimental images contain features that are not reflected in the theoretical simulations. Data augmentation has been applied during the training of our model (see sections S3.2 and S3.4) to capture these effects. Although limited by the scarcity of experimental results suitable to apply our methodology, we have obtained very encouraging results. We have selected constant–height AFM images of dibenzothiophene and 2-iodotriphenylene from refs. (*27*) and (*67*), corresponding to 10 different tip–sample distances, covering a height range of 100 pm for dibenzothiophene (identical to the one spanned by the 3D stacks of theoretical images used to train our model) and 72 pm for 2-iodotriphenylene (see section S5 for details). Despite the strong noise in the images and the white lines crossing the images diagonally (see fig. S8), the prediction of dibenzothiophene is perfect, scoring 1.00 on the 4–gram, whereas for 2-iodotriphenylene the model predicts "2iodtriphenylene", missing a hyphen but providing all the relevant chemical information. Despite the good results obtained with these two experimental image stacks, no significant conclusions can be extracted given the very small size of the test. A larger, systematic analysis with a proper experimental data is necessary to further address the accuracy of our model.

5 Conclusions

In this work, we have shown how deep learning models, trained with the simulated HR-AFM 3D image stacks for 678,000 molecules included in the QUAM-AFM dataset, are able to perform full chemical-structural identification of molecules. Motivated by the unfeasibility of defining a classification in the usual sense of AI, we turned the problem into an image captioning problem.

Thus, instead of aiming to have a model that knows every atomic structure, we endow it with the ability to formulate. As a result the model is not only able to identify images that have not been previously shown to it, but it is also able to predict the IUPAC name of these unknown structures. We have devised a two–step procedure involving the combination of two M-RNNs. In a first step, the M-RNN$_A$ identifies the attributes, the most relevant functional groups present in the molecule. This initial step is already of importance because the algorithm provides useful information about the chemical characteristics of the molecule. In a second step, the AM-RNN, whose inputs arise from the M-RNN$_A$, sorts the information of the functional groups, adds extra characters (connectors, position labels, other tags, etc.) and the remaining functional groups that are not part of the *attributes* set. That is, the AM-RNN assigns the positions of the functional groups, completes the remaining terms and writes down the final IUPAC name of the molecule.

We have tested the model on a set of 816,000 AFM images belonging to 34,000 molecules that have not been shown before to the network. The bare results point out that in a 43% of the cases the predictions are exactly the same with respect to the reference in QUAM-AFM, character by character. To further asses the usefulness of the wrong predictions by the model, we apply the metrics defined by the BLEU n-gram. This is a robust methodology used in assessing the accuracy of our model that weights the accuracy of a prediction against a reference. The accuracy of the *attribute* prediction assessed with the 1-gram scores 0.95, whereas the overall accuracy of the model is determined with the cumulative 4-gram, scoring 0.76. This high value means that, even when the model does not achieve a perfect prediction, it provides valuable chemical insight, leading to a correct IUPAC name of a similar molecule in the vast majority of the cases. Finally, it is worth noting that this model could be applied to AFM images obtained in experiments. Due to the lack of a systematic and large set of experimental images, we do not have definitive conclusions yet, but we have obtained very encouraging results in two examples.

Acknowledgments

We deeply thank Dr. D. Martin-Jimenez and Dr. D. Ebeling for providing us with the experimental images of 2-iodotriphenylnene used to test the model. We would like to acknowledge support from the Comunidad de Madrid Industrial Doctorate programme 2017 under reference number IND2017/IND-7793 and from Quasar Science Resources S.L. P. Pou and R.P. acknowledge support from the Spanish Ministry of Science and Innovation, through project PID2020-115864RB-I00 and the "María de Maeztu" Programme for Units of Excellence in R&D (CEX2018-000805-M). Computer time provided by the Red Española de Supercomputación (RES) at the Finisterrae II Supercomputer is also acknowledged.

References and Notes

1. F. J. Giessibl, *Science* **267**, 68 (1995).

2. R. García, R. Pérez, *Surf. Sci. Rep.* **47**, 197 (2002).

3. F. J. Giessibl, *Rev. Mod. Phys.* **75**, 949 (2003).

4. L. Gross, F. Mohn, N. Moll, P. Liljeroth, G. Meyer, *Science* **325**, 1110 (2009).

5. N. Moll, L. Gross, F. Mohn, A. Curioni, G. Meyer, *New J. Phys.* **12**, 125020 (2010).

6. M. Ellner, P. Pou, R. Pérez, *ACS Nano* **13**, 786 (2019).

7. J. Van Der Lit, F. Di Cicco, P. Hapala, P. Jelinek, I. Swart, *Phys. Rev. Lett.* **116**, 096102 (2016).

8. P. Hapala, *et al.*, *Nat. Commun.* **7**, 11560 (2016).

9. P. Hapala, *et al.*, *Phys. Rev. B* **90**, 085421 (2014).

10. L. Gross, *et al.*, *Angew. Chem. Int. Ed.* **57**, 3888 (2018).

11. L. Gross, *et al.*, *Science* **337**, 1326 (2012).

12. L. Gross, *et al.*, *Science* **324**, 1428 (2009).

13. F. Mohn, L. Gross, N. Moll, G. Meyer, *Nat. Nanotechnol.* **7**, 227 (2012).

14. D. G. de Oteyza, *et al.*, *Science* **340**, 1434 (2013).

15. S. Clair, D. G. de Oteyza, *Chem. Rev.* **119**, 4717 (2019).

16. F. J. Giessibl, *Rev. Sci. Instrum.* **90**, 011101 (2019).

17. Q. Zhong, X. Li, H. Zhang, L. Chi, *Surf. Sci. Rep.* **75**, 100509 (2020).

18. K. Ø. Hanssen, *et al.*, *Angew. Chem. Int. Ed.* **51**, 12238 (2012).

19. D. Ebeling, *et al.*, *Nat. Commun.* **9**, 2420 (2018).

20. F. Schulz, *et al.*, *ACS Nano* **12**, 5274 (2018).

21. B. Schuler, G. Meyer, D. Peña, O. C. Mullins, L. Gross, *J. Am. Chem. Soc.* **137**, 9870 (2015).

22. Y. Sugimoto, *et al.*, *Nature* **446**, 64 (2007).

23. N. J. van der Heijden, *et al.*, *ACS Nano* **10**, 8517 (2016).

24. C. S. Guo, M. A. Van Hove, R. Q. Zhang, C. Minot, *Langmuir* **26**, 16271 (2010).

25. J. Tschakert, *et al.*, *Nat. Commun.* **11**, 5630 (2020).

26. P. Jelinek, *J. Phys.: Condens. Matter* **29**, 343002 (2017).

27. P. Zahl, Y. Zhang, *Energy Fuels* **33**, 4775 (2019).

28. P. Zahl, *et al.*, *Nanoscale* **13**, 18473 (2021).

29. A. Krizhevsky, I. Sutskever, G. E. Hinton, *Commun. ACM* **60**, 84–90 (2017).

30. K. Simonyan, A. Zisserman, *3rd Int. Conf. Learn. Rep.*, Y. Bengio, Y. LeCun, eds. (ICLR, Wall St, La Jolla, CA, USA, 2015).

31. K. He, X. Zhang, S. Ren, J. Sun, *Proc. Comput. Vision Pattern Recognit. (CVPR)* (IEEE Computer Society Press, Piscataway, NJ, USA, 2016), pp. 770–778.

32. C. Szegedy, S. Ioffe, V. Vanhoucke, A. A. Alemi, *31rd Proc. AAAI Conf. on Artificial Intelligence* (AAAI Press, Palo Alto, CA, USA, 2017), p. 4278–4284.

33. F. Chollet, *Proc. Comput. Vision Pattern Recognit. (CVPR)* (IEEE Computer Society Press, Piscataway, NJ, USA, 2017), pp. 1800–1807.

34. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, *Proc. Comput. Vision Pattern Recognit. (CVPR)* (IEEE Computer Society Press, Piscataway, NJ, USA, 2018), pp. 4510–4520.

35. K. He, X. Zhang, S. Ren, J. Sun, *Int. Conf. Comput. Vision (ICCV)* (IEEE Computer Society Press, Piscataway, NJ, USA, 2015), pp. 1026–1034.

36. J. Carracedo-Cosme, C. Romero-Muñiz, R. Pérez, *Nanomaterials* **11**, 1658 (2021).

37. A. Liebig, P. Hapala, A. J. Weymouth, F. J. Giessibl, *Sci. Rep.* **10**, 14104 (2020).

38. J. Carracedo-Cosme, C. Romero-Muñiz, P. Pou, R. Pérez, *J. Chem. Inf. Model.* **62**, 1214 (2022).

39. N. Artrith, *et al.*, *Nat. Chem.* **13**, 505 (2021).

40. B. Alldritt, *et al.*, *Sci. Adv.* **6**, eaay6913 (2020).

41. J. Mao, *et al.*, *arXiv preprint arXiv:1412.6632* (2014).

42. Q. You, H. Jin, Z. Wang, C. Fang, J. Luo, *Proc. Comput. Vision Pattern Recognit. (CVPR)* (IEEE Computer Society Press, Piscataway, NJ, USA, 2016), pp. 4651–4659.

43. M. Cornia, M. Stefanini, L. Baraldi, R. Cucchiara, *Proc. Comput. Vision. Pattern Recognit. (CVPR)* (IEEE Computer Society Press, Piscataway, NJ, USA, 2020), pp. 10578–10587.

44. L. Zhou, *et al.*, *34rd Proc. AAAI Conf. on Artificial Intelligence* (AAAI Press, Palo Alto, CA, USA, 2020), pp. 13041–13049.

45. K. Rajan, A. Zielesny, C. Steinbeck, *J. Cheminf.* **13**, 61 (2021).

46. D.-A. Clevert, T. Le, R. Winter, F. Montanari, *Chem. Sci.* **12**, 14174 (2021).

47. K. Rajan, H. O. Brinkhaus, A. Zielesny, C. Steinbeck, *J. Cheminf.* **12**, 60 (2020).

48. D. Weininger, *J. Chem. Inf. Comput. Sci.* **28**, 31 (1988).

49. J. L. Elman, *Cognit. Sci.* **14**, 179 (1990).

50. P. F. Brown, S. A. Della Pietra, V. J. Della Pietra, R. L. Mercer, *Comput. Linguist.* **19**, 263 (1993).

51. J. Chung, C. Gulcehre, K. Cho, Y. Bengio, *27th NeurIPS Workshop on Deep Learning* (MIT Press, Cambridge, MA, USA, 2014).

52. H. Sak, A. W. Senior, F. Beaufays, *arXiv preprint arXiv:1402.1128* (2014).

53. H. A. Favre, W. H. Powell, *Nomenclature of Organic Chemistry: IUPAC Recommendations and Preferred Names 2013* (The Royal Society of Chemistry, 2014).

54. K. Papineni, S. Roukos, T. Ward, W.-J. Zhu, *40th Proc. Annu. Meet. ACL* (Association for Computational Linguistics, Philadelphia, Pennsylvania, 2002), p. 311–318.

55. A. J. Weymouth, T. Hofmann, F. J. Giessibl, *Science* **343**, 1120 (2014).

56. O. Vinyals, A. Toshev, S. Bengio, D. Erhan, *Proc. Comput. Vision Pattern Recognit. (CVPR)* (IEEE Computer Society Press, Piscataway, NJ, USA, 2015), pp. 3156–3164.

57. T. K. Shimizu, *et al.*, *J. Phys. Chem. C* **124**, 26759 (2020).

58. K. Rajan, A. Zielesny, C. Steinbeck, *J. Cheminf.* **13**, 34 (2021).

59. J. Pennington, R. Socher, C. Manning, *Conf. Empirical Methods in Nat. Lang. Proc. (EMNLP)* (Association for Computational Linguistics, Doha, Qatar, 2014), pp. 1532–1543.

60. T. Chen, S. Kornblith, M. Norouzi, G. Hinton, *37th Int. Conf. Mach. Learn. (ICML)* (PMLR, 2020), pp. 1597–1607.

61. S. Minaee, *et al.*, *ACM Comput. Surv.* **54** (2021).

62. A. Vaswani, *et al.*, *Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NIPS)* (Long Beach, CA, USA, 2017), pp. 5998–6008.

63. L. Krasnov, I. Khokhlov, M. Fedorov, S. Sosnin, *Sci. Rep.* **11**, 14798 (2021).

64. J. Handsel, B. Matthews, N. J. Knight, S. J. Coles, *J. Cheminf.* **13**, 79 (2021).

65. S. Heller, A. McNaught, S. Stein, D. Tchekhovskoi, I. Pletnev, *J. Cheminf.* **5**, 7 (2013).

66. M. Krenn, F. Häse, A. Nigam, P. Friederich, A. Aspuru-Guzik, *Mach. Learn.: Sci. Technol.* **1**, 045024 (2020).

67. D. Martin-Jimenez, *et al.*, *Phys. Rev. Lett.* **122**, 196101 (2019).

Supplementary Information for Molecular Identification from AFM images using the IUPAC Nomenclature and Attribute Multimodal Recurrent Neural Networks

Jaime Carracedo-Cosme,[1,2] Carlos Romero-Muñiz,[3]
Pablo Pou,[2,4] Rubén Pérez,[2,4,*]

[1]Quasar Science Resources S.L., Camino de las Ceudas 2, E-28232 Las Rozas de Madrid, Spain
[2]Departamento de Física Teórica de la Materia Condensada,
Universidad Autónoma de Madrid, E-28049 Spain
[3]Departamento de Física Aplicada I, Universidad de Sevilla, E-41012, Seville, Spain
[4]Condensed Matter Physics Center (IFIMAC),
Universidad Autónoma de Madrid, E-28049 Madrid, Spain

* ruben.perez@uam.es

Contents

S1 IUPAC tokenization **3**

S2 M-RNN$_A$ and AM-RNN models: A layer description **5**

S3 Model learning **8**

S3.1 Molecular Classes for Transfer Learning 8

S3.2 Pre–training the CNN component 9

S3.3 M–RNN and AM–RNN optimization 10

S3.4 M–RNN and AM–RNN training 11

S4 Influence of the molecular torsion in the model performance: gas–phase versus flat configurations **13**

S5 Experimental test **16**

S6 Embedding **18**

References and Notes **21**

S1 IUPAC tokenization

In order to tokenize the IUPAC nomenclature we selected a set of terms consisting of prefixes, suffixes, numbers, etc., whose combinations generates the 686,000 IUPAC names corresponding with the molecules belonging to Quasar Science Resources - Universidad Autónoma de Madrid - Atomic Force Microscopy Image Dataset (QUAM-AFM) (*1*). Deep learning models require large datasets in which each target is repeated many times to be learned during the training. We have analysed the number of times that each term appears in QUAM-AFM and removed those terms that are repeated less than 100 times. This reduces the total number of terms considered to a total of 199, shown in table S1. Consequently, we have discarded molecules whose decomposed IUPAC name contains any of these terms. Similarly, extremely long IUPAC names have very little representation in QUAM-AFM, so we also discard molecules whose IUPAC name decomposes into more than 57 terms. In this way, we have slightly reduced the set of available inputs to a total of 678.000 molecules.

acen	acet	acid	acr	alde	alen	amate	amide	amido	amim
amine	amino	amo	ane	ani	anil	ano	anone	anthr	ate
ato	aza	aze	azi	azido	azin	azo	azol	benz	brom
but	carb	chlor	chr	cyclo	ene	eno	eth	fluor	form
furan	furo	hyde	hydr	ic	ida	ide	idin	idine	ido
imid	imidin	imine	imino	ind	ine	ino	iod	iso	itrile
ium	meth	mine	naphth	nitr	nium	nyl	oate	oic	ol
ole	oli	olin	oline	olo	om	one	oso	oxo	oxol
oxy	oxyl	oyl	phen	phth	prop	pter	purin	pyr	pyrrol
quin	sulf	thi	tri	urea	yl	ylid	yridin	zin	zine
dodeca	undeca	lambda	phosph	cinnam	xanth	porph	coron	deca	guan
hept	amic	pent	enal	octa	anal	nona	mido	nida	azet
tetr	ulen	anol	hypo	pine	ysen	anth	tere	acyl	yrin
inin	tris	pino	mid	hex	nia	bis	per	nio	pin
ite	rin	alo	en	di	11	yn	an	cy	de
15	10	13	14	12	in	18	21	az	al
bi	et	ep	id	ox	il	or	16	17	on
(	6	)	-	[	a	]	N	’	1
7	4	5	3	9	2	8	H		,
o	b	e	c	C	d	O	g	f	

Table S1: Table of terms for International Union of Pure and Applied Chemistry (IUPAC) decomposition. The elements above the bold line are the subset of 100 attributes considered. The grey cell does not correspond to any term, it has been coloured in order to distinguish it from the term that spells an empty space between two words.

As explained in the main text, we combine two different Multimodal Recurrent Neural Network (M-RNN)s to achieve molecular identification. The first network, named Multimodal Recurrent Neural Network for attribute prediction (M-RNN$_A$), uses as input the stack of Atomic Force Microscopy (AFM) images and its aim is to extract the *attributes*, a 100-element subset of the terms which are mainly used to designate functional groups and not positions (see table S1). The second network, Atribute Multimodal Recurrent Neural Network (AM-RNN), consists of an adaptation of the M-RNN (*2*) for the IUPAC nomenclature prediction, where we add an input of semantic attributes predicted by the M-RNN$_A$ (see section S2 for details). This strategy differs from the concept of *attention* (*3, 4, 5*), implemented in other Natural Languaje Procesing (NLP) problems, and that has led to the development of the transformers (*6, 7, 8, 9*). In this case we assume that the relationship between each IUPAC name and each molecule is biunivocal, (although there are some exceptions we suppose that the names of each compound have been entered under the same rules in Pubchem). This approach makes the problem posed different from most of the challenges faced in image captioning where there are multiple descriptions for the same image. Therefore, a prior identification of the main functional groups, not only releases the Convolutional Neural Network (CNN) component of the AM-RNN from the task of identifying these moieties, but, more importantly, almost halves the number of possible predictions of the AM-RNN: By feeding the AM-RNN with the attributes that are present in the IUPAC name (predicted by the M-RNN$_A$), we are also effectively excluding the large number of them that do not form part of it. This strategy improves significantly its performance.

S2 M-RNN$_A$ and AM-RNN models: A layer description

As previously mentioned, the architecture developed for molecular identification with the IUPAC nomenclature is composed of two models, M-RNN$_A$ and AM-RNN, (see Figure 2 of the main text). Each one is composed by a CNN, a RNN and a multimodal componet φ. Figure S1 displays the type of layers that constitute each component of both M-RNN$_A$ and AM-RNN. The architecture of the CNN component is identical in both models, while RNN and φ share the same structure and, except for the RNN layer, the same type of layers with different number of units (e.g. kernels in convolutional layers, vector length in fully connected layers, etc) according to the specific purpose of the model. Figure S2 provides the details, highlighting the

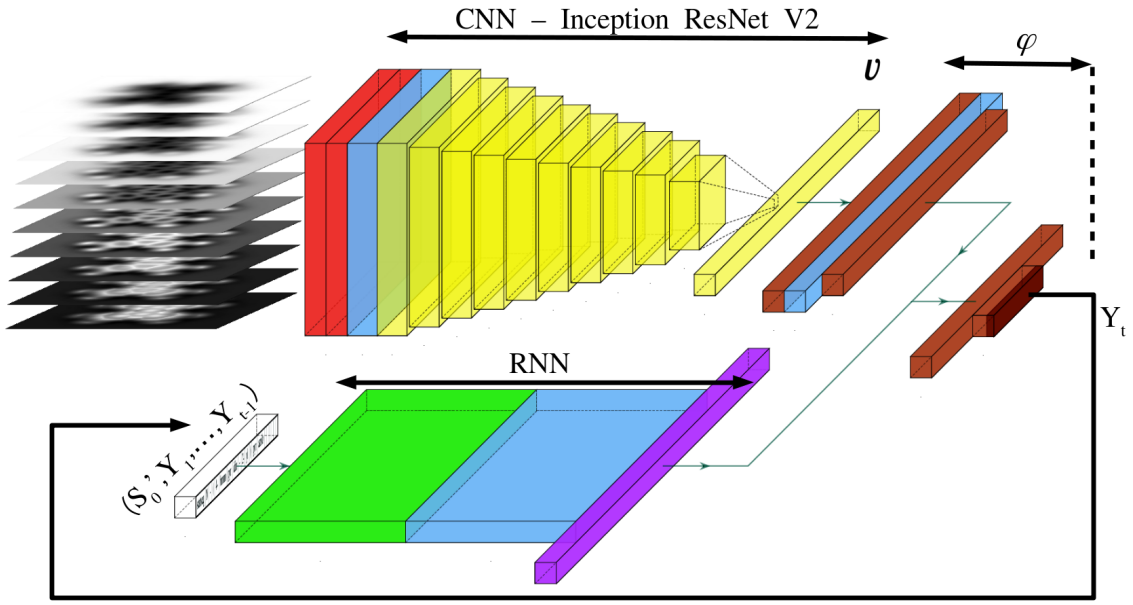


Figure S1: Graphical layer representation of M-RNN$_A$ and AM-RNN with the layers that constitute their three components, CNN, RNN and φ. The CNN component follows the Inception ResNet V2 model, where the first 2D–convolutional layers have been replaced by two 3D convolutional layers (to process the image stack, shown in red), followed by a dropout layer (blue). The yellow blocks are just a pictorial representation of each block of the original Inception ResNet V2 model. Notice that the last fully connected layer of the model, which is specific for the original classification task, has been removed, obtaining an output vector (v) with length 1539. The RNN and φ components include embedding (green), dropout (blue), fully connected (brown) and recurrent (purple) layers. The purple box represents a Gated Recurrent Unit (GRU) layer in M-RNN$_A$, whereas in the AM-RNN it represents a Long Short-Term Memory (LSTM) layer. S'_0 represents the input for the RNN component in the first time step: the *startseq* token in the M-RNN$_A$ and the concatenation of the *attributes* with the *starseq* token in AM-RNN. The subsequent inputs include the *attributes* (*terms*) predicted in previous time steps by M-RNN$_A$ (AM-RNN). Although both M-RNN$_A$ and AM-RNN have the same structure, each layer of RNN and φ components has different dimensions (see Figure S2 for a detailed description).

differences between the layers of the RNN and φ components of the two models.

The CNN component consists of a modification of the Inception ResNet V2 model (*10*), identical for both M-RNN$_A$ and AM-RNN. This well–known model has been developed to be applied to 2D images whereas in our case we process 3D maps (stacks of 10 AFM images with various tip–sample distances). Therefore we have replaced the first 2D–convolutional layers in Inception ResNet V2 by two 3D convolutional layers, each one with 32 filters, (3,3,3) kernel size and (2,1,1) strides, followed by a dropout layer. We have verified that this dropout layer is essential for the model to generalize to different images, such as the experimental ones. In addition, we have removed the last fully connected layer of the model, which is specific for the original classification task, obtaining an output vector (v) with length 1539.

The goal of the RNN component is to use sequential data to add new *terms* to the formulation. To this end, the architecture of this component is developed according to two key

Model	Component	Operator	Units	Activation	Connections
M-RNN$_A$	RNN	Input$_{RNN}$	19		
		Embedding	32		Input$_{RNN}$
		Dropout$_1$	0.2		Embedding
		GRU	128		Dropout$_1$
	ψ	FC$_1$	1024	ReLU	v
		Dropout$_2$	0.2		FC$_1$
		FC$_2$	512	ReLU	Dropout$_2$
		Concat			FC$_2$, GRU
		FC$_3$	256	ReLU	Concat
		FC$_4$	103	Softmax	FC$_3$
AM-RNN	RNN	Input$_{RNN}$	76		
		Embedding	32		Input$_{RNN}$
		Dropout$_1$	0.2		Embedding
		LSTM	256		Dropout$_1$
	ψ	FC$_1$	1024	ReLU	v
		Dropout$_2$	0.2		FC$_1$
		FC$_2$	512	ReLU	Dropout$_2$
		Concat			FC$_2$, GRU
		FC$_3$	256	ReLU	Concat
		FC$_4$	202	Softmax	FC$_3$

Figure S2: Layer–by–layer details of the RNN and φ components integrated in M-RNN$_A$ and AM-RNN. v denotes the output vector of the CNN component. The layers are the same for both models, except for the recurrent one, a GRU in M-RNN$_A$ (attribute prediction) and an LSTM in AM-RNN (term prediction).

objectives: Firstly, to embed a representation of each *term* based on its semantic meaning and, secondly, to store the semantic temporal context in the recurrent layers. To perform the representation of each *term* in a vector space, the RNN component has an embedding layer that is able to capture relationships between *terms* (see section S6 for a more detailed analysis). The embedding layer is followed by a dropout layer that acts as a regularizer and finally the data goes through a recurrent layer which, with the temporal context generated by all the predictions made in previous time steps, makes a proposal of predictions that is processed by the multimodal component (see fig. S1). The recurrent layer is a GRU in M-RNN$_A$ (attribute prediction) and an LSTM in AM-RNN (term prediction). The multimodal component φ first processes the CNN output v in two fully connected layers with a dropout between them. Subsequently, this output is concatenated with the output of the RNN (see fig. S1). Finally, the result of the concatenation feeds two fully connected layers, the first one activated with Rectified Linear Unit Activation Function (ReLU) activation function whereas the second one is activated with Soft Approximation of Max (Softmax) activation function, that converts the outputs from the layer into a vector with components that represent probabilities that sum to one.

S3 Model learning

Both M-RNN$_A$ and AM-RNN could, in principle, be trained in an end–to–end process, however, this would require more than a year of training and it has been found that this type of training for M-RNN results in not providing detailed descriptions (*11*). To avoid it, we train each component of the models in different stages, fixing the weights of the CNN and RNN alternatively while training the rest of the model.

In the first stage, both CNN and RNN components are initialised with random weights, so if we fix the weights of the CNN while training the RNN, the CNN would perform a random representation of the input image stack, and consequently, the weights of the RNN component would be updated under random rules. An analogous reasoning can be applied for the reverse case, in which we would fix the weights of the RNN and train the CNN. We solve this issue initialising the CNN component with pre–trained weights. To determine this weights, we perform a classification with the CNN based on classes of molecules that shared the same chemical composition (number of different chemical species and number of atoms for each specie, excluding the H atoms), as described in detailed in section S3.1.

S3.1 Molecular Classes for Transfer Learning

The classification of AFM images defining the model output as each individual molecule is an impossible task because QUAM-AFM does not include enough images of each particular structure and has an excessive number of molecules (classes). Thus, we simplify the problem by grouping molecules based on their chemical composition. Hence, we define the class of a molecule by the type of atomic species that it contains and the number of repeated atoms of each of these species. To obtain a representative number of images of each class, we exclude the hydrogens from the species list (see fig. S3), so that molecules with completely different structures such as pyrazine, pyridazine, but-2-enedinitrile or butanedinitrile belong to the same class (C_4N_2). This results in a total of 2339 classes for the molecule structures considered in QUAM-AFM.

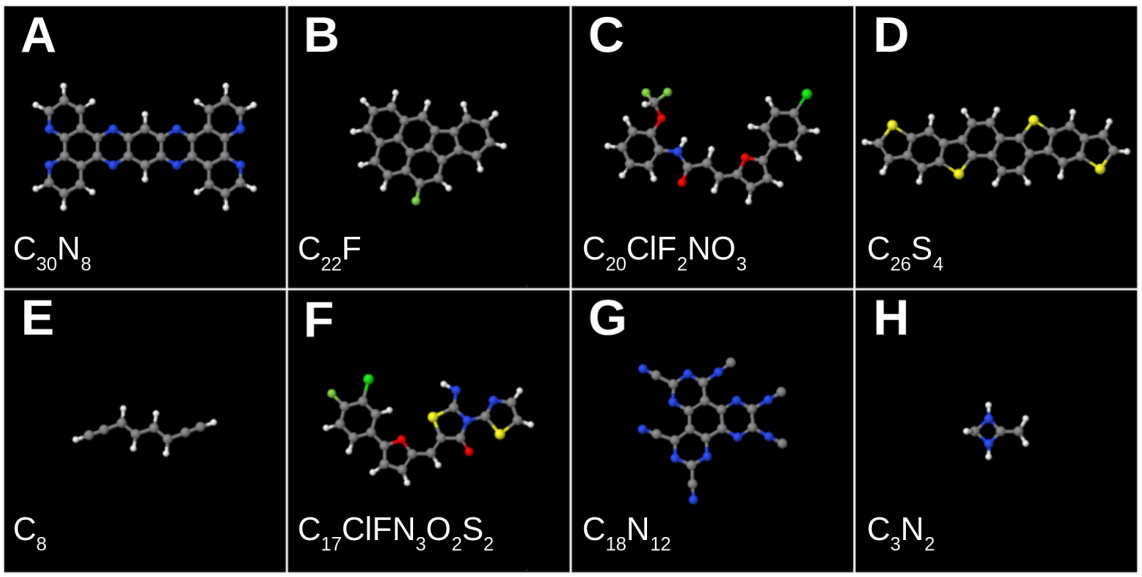


Figure S3: Atomic structures belonging to different classes for classification according to their chemical species.

S3.2 Pre–training the CNN component

As mentioned above, it is necessary to pre-train the CNN component with a classification that groups the molecules in the QUAM-AFM dataset in classes describing its chemical composition, irrespective of their structure. We perform this classification with the same Inception ResNet V2 model (*10*) that we used for the CNN element in the two M-RNNs . To this end, we replace the first convolutional layer by two 3D convolutional layers followed by a dropout layer and, instead of removing the output layer as described in the main text for the NLP target, we modify its number of units to 2339, corresponding to the number of classes defined in section S3.1.

In each epoch, we select a single combination of simulation parameters from those included in the QUAM-AFM dataset (*1*) for each molecule. Therefore, the model receives inputs of the same molecules but different image stacks at each epoch. In addition, we apply an Image Data Generator (IDG) (see fig. S4) to the input image stack in order to simulate experimental features that are not captured in the theoretical simulations, as discussed in ref. (*12*). The values selected for the distortions are randomly chosen in ranges of [1,360]–degree rotations, ± 0.15 zoom range, ± 0.1 shear range, and ± 0.1 both vertical and horizontal shift range, as illustrated in fig. S4. When a molecule is rotated or moved during an AFM experiment, all

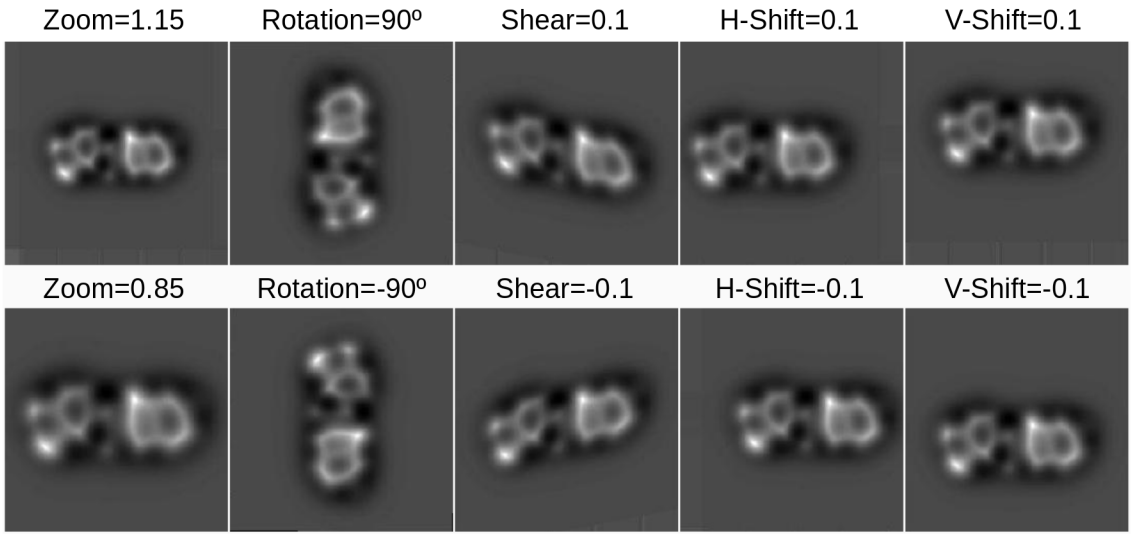


Figure S4: Results of applying the IDG to the same AFM image of a N-(5-amino-4-methylpyridin-2-yl)-6-fluoro-1-benzothiophene-2-carboxamide molecule.

resulting images show similar variations, so we apply the same deformation parameters to the ten images that compose each stack. These variations, coupled with the dropout layer in the CNN component, ensure that the model is robust regarding soft variations of the inputs that are present in experimental images.

We train the CNN minimizing the error of the Negative Log Likelihood loss function with the Adaptive Moment estimator (Adam) optimizer (*13*). To speed up the convergence, we apply a batch–normalization (*10, 14*), setting the mean to zero and the variance to one in the input layers (*15, 16*). We found that this normalization not only makes the training faster but also improves the classification results, reaching an accuracy of 0.97 in the test prediction.

S3.3 M–RNN and AM–RNN optimization

Because of the analogies between M-RNN$_A$ and AM-RNN, their loss functions and trainings are similar, hence we define the optimization problem for both at the same time with the notation used in the main text. The function to be maximized is the probability of obtaining a correct sentence $S = (S_1, ..., S_N)$ given an input (I, S'_0):

$$\theta^* = arg\max_{\theta} \sum_{(I,S)} \log p(S|I, S'_0; \theta) \tag{1}$$

where θ represents the model parameters, and I is the 3D image stack. Note that the prediction of the IUPAC nomenclature, according to the decomposition performed on a set of terms,

M-RNN$_A$

Stage 1. CNN fixed			Stage 2. RNN fixed			Stage 3. CNN fixed		
Lr	Epochs	Batch	Lr	Epochs	Batch	Lr	Epochs	Batch
10^{-3}	5	64	10^{-3}	-	32	10^{-3}	20	64
10^{-4}	3		10^{-4}	6		10^{-4}	10	

AM-RNN

Stage 1. CNN fixed			Stage 2. RNN fixed			Stage 3. CNN fixed		
Lr	Epochs	Batch	Lr	Epochs	Batch	Lr	Epochs	Batch
10^{-3}	10	64	10^{-3}	-	32	10^{-3}	40	64
10^{-4}	5		10^{-4}	8		10^{-4}	30	

Figure S5: Details of the training scheme for each stage of each of the models. Lr is a short for learning rate.

must depend on the predictions already performed, i.e. the prediction must take into account previously predicted terms in order to have semantic meaning, so it is a time series. During the training, we feed the model at each time step t with the target of previous time steps $(S'_0, S_1, ..., S_{t-1})$ and not with the predictions performed $(Y_1, ..., Y_{t-1})$, which in some cases are wrong. We refer to each input of the model at the time step t as the pair $(S'_0, S_1..., S_{t-1}; I)$. Note that, in this way, the final prediction length of each molecule depends on t. Thus, the prediction of S depends on the prediction of each specific term, which in turn depends not only on I but also on all the predictions performed in previous time steps. Since the model predicts a single term of the sequence at each time step, it is natural to apply the chain rule to model the joint probability over the sequential *terms*. Hence, the probability of obtaining a correct prediction for the complete sequence is described by the sum of the logarithmic probabilities over the terms. Therefore, the maximum log–likelihood function is as follows:

$$L(S, I) = \sum_{t=1}^{N} \log p(S_t | I, S'_0, S_1, ..., S_{t-1}; \theta). \tag{2}$$

As the deep learning optimization techniques consist of searching for a minimum rather than a maximum, the loss function is described by the sum of the negative log–likelihood:

$$L(S, I) = -\sum_{t=1}^{N} \log p(S_t | I, S'_0, S_1, ..., S_{t-1}; \theta). \tag{3}$$

S3.4 M–RNN and AM–RNN training

The training of both M-RNN$_A$ and AM-RNN proceeds in three stages in which the weights of the CNN and RNN components are fixed (non-trainable) alternatively. In the first stage, the

model initialises all its weights randomly except those of the CNN component, which are pre–trained with the chemical classification explained in section S3.2, and focuses on the training of the RNN. Although specialised in a different classification, the CNN component output already represents high–level features of each input stack. The model is then fed with the input (I, S). In the same way as we do for the CNN pre-training described in section S3.2, a random combination of the simulation parameters described in (I) is selected for each input at each epoch. Thus, although the CNN's weights are fixed, the high–level representation of each structure is different in each epoch. This selection, coupled with the dropout layer of the φ component, ensures that the RNN component does not overfit for specific representations of the input images.

The aim of the second stage is to specialise the weights of the CNN component in the semantic prediction. To this end, the weights of the RNN component are fixed and the IDG described in section S3.2 is applied to the input stack. Furthermore, the selection of simulation parameters is randomly chosen for each input I. In this stage the prediction is performed for a single time step of each pair (S, I). After completion of this second stage, the CNN component provides specific details for the IUPAC formulation rather than to the chemical classification. Finally, the third training stage repeats the process of the first one, fixing the weights of the CNN. Further details for the training at each stage (number of epochs, batch size, learning rate) are shown in fig. S5.

S4 Influence of the molecular torsion in the model performance: gas–phase versus flat configurations

HR-AFM shows an outstanding lateral resolution for quasi-planar molecules, but it is more limited to discern correctly molecules with a significant torsion. This is due to the nature of the contrast and the probe flexibility. The exponential behavior of the Pauli repulsion with respect to the tip-sample distance makes this contribution the most relevant contrast source. Thus, each atom in the sample will behave as an umbrella of around 2-3 Å veiling any other electronic charge density below this region (except some cases where deformations in the charge density may occur). Furthermore, the CO molecule will tilt under this repulsion and will block the access of the probe to the region underneath.

Previous works (*17*) suggest that it is difficult to retrieve information from parts of the molecule that are located more than 1.5 Å below the topmost atoms. According to this, we expect our algorithm to provide a poorer performance when finding out the IUPAC names of molecules showing significant out-of-plane distortions. In order to quantify this, we have carried out some tests to directly show how the model accuracy improves for planar structures. We have selected four molecules showing a torsion of about 1.8 Å and recalculated them in a forced planar configuration. These planar configurations have been determined by DFT calculations where we fix the z-position of all the atoms in the molecule and let them move in the xy plane in order to reach the ground state configuration compatible with this constraint. If we feed the algorithm with the image stacks of the planar configuration the Bilingual Evaluation Understudy (BLEU) 4-gram score increases significantly in all cases, even when the chemical nature of the molecule makes the recognition by our model more difficult. Figure S6 shows two molecules where the prediction for the non planar failed but where we obtain a highly accurate result (an almost perfect match in the cases shown) for the planar configurations, with only an error in the misplacement of one of the functional groups. Figure S7 shows the result for two other molecules that contain functional groups that are difficult to discern such as fluorine atoms, easily mistakable with other terminations like diketones which may display faint AFM features. Although the BLEU 4-gram for the planar configuration is not high there is a clear improvement compared to the non–planar cases.

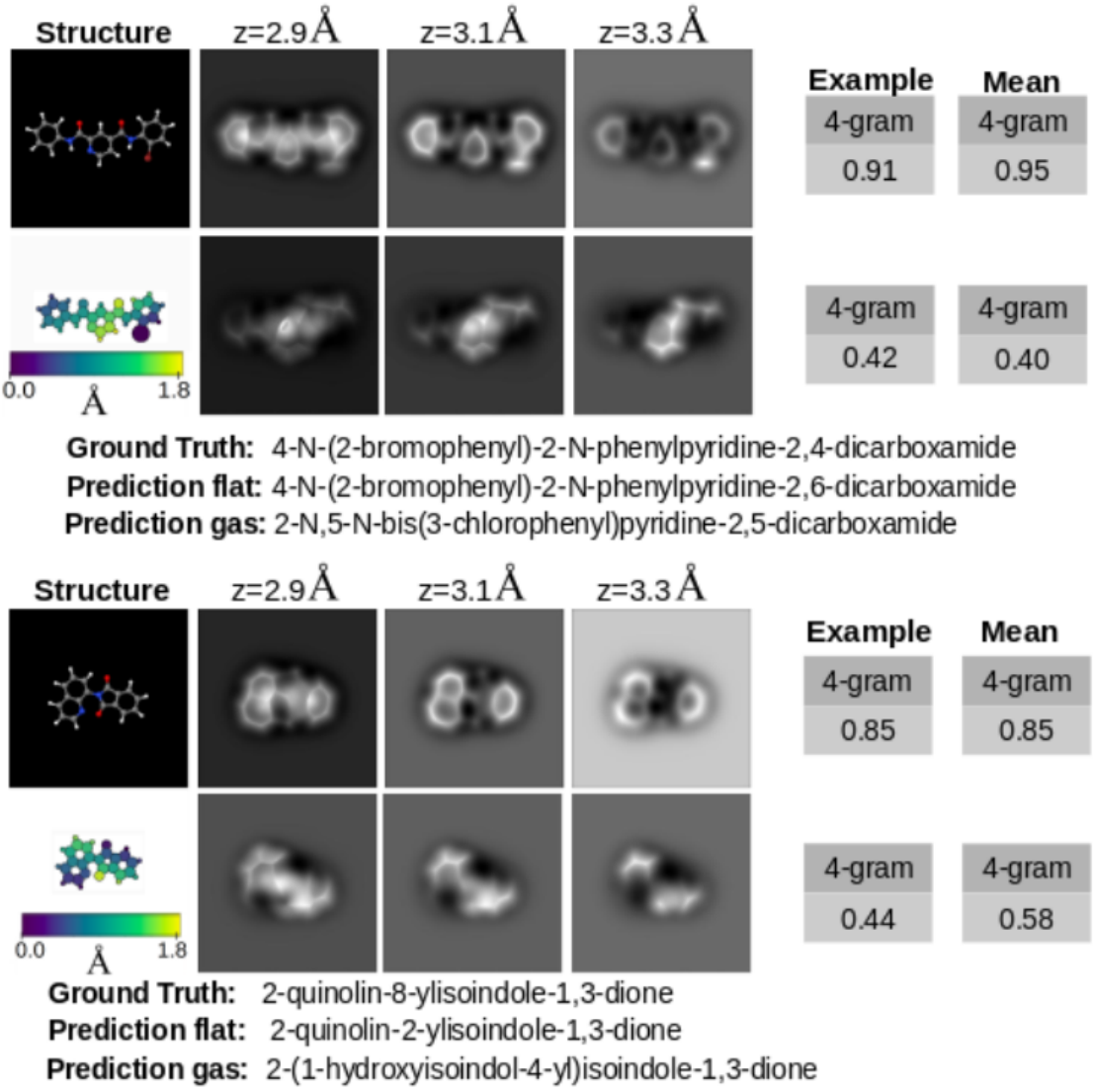


Figure S6: Comparison of the model predictions for the gas–phase and the forced planar configuration of a representative molecule, whose ball–and–stick depiction and height map are shown on the left. The upper AFM images correspond to the simulation with the structure in a completely planar configuration while the lower ones correspond to the images for the gas–phase structure. To the right of the AFM images is the 4–gram score corresponding to the prediction shown (flat and gas–phase structure, respectively). Further to the right is the average of the 4–gram scores obtained for images generated with the 24 combinations of operational parameters considered in QUAM-AFM. The prediction example shown is the one that scored closest to the mean.

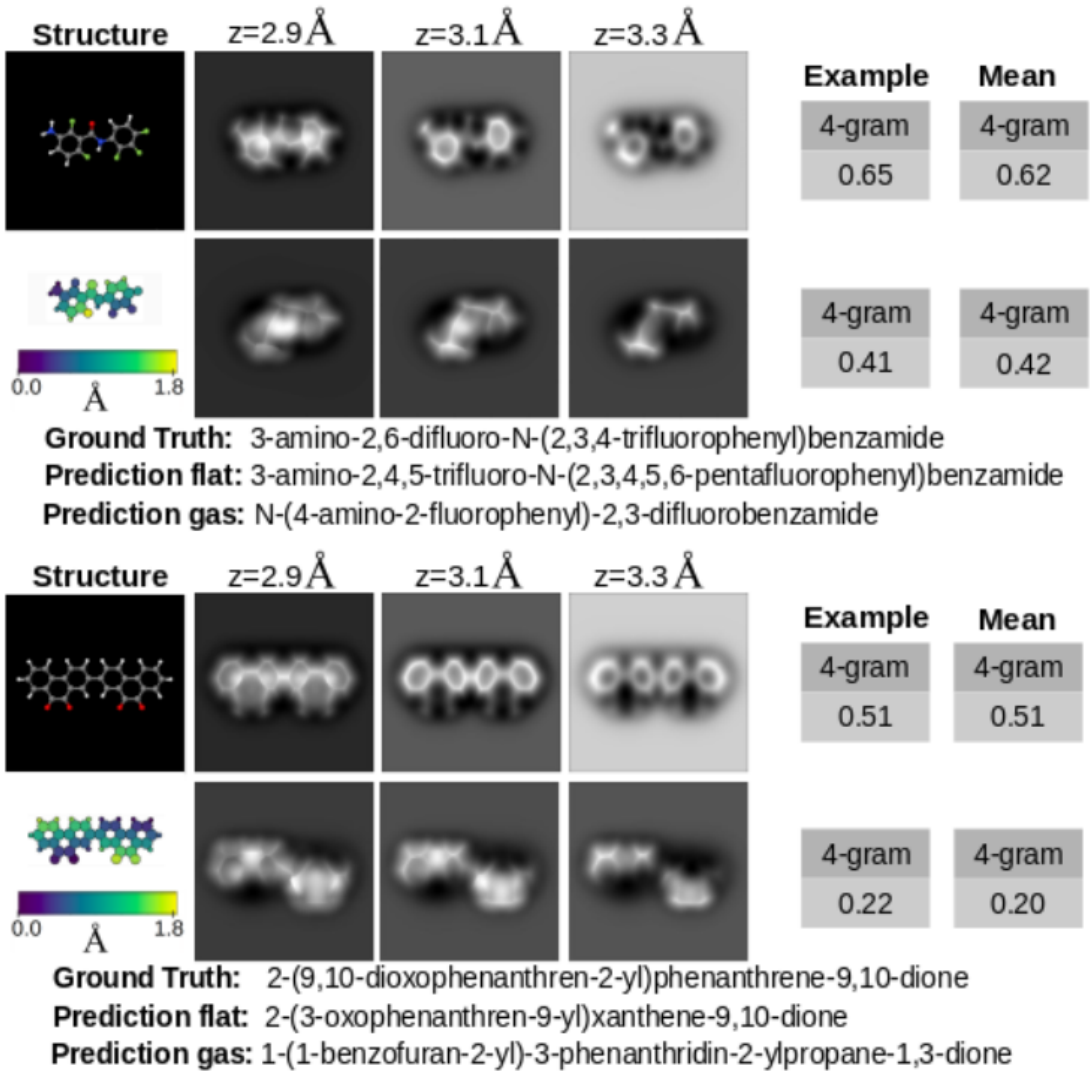


Figure S7: As fig. S6 for two molecules that include chemical chemical groups, like F atoms (top) and diketones (bottom), that display faint AFM features and thus are more difficult to recognise by the model.

S5 Experimental test

We have performed a test with 3D stacks of experimental AFM images for dibenzothiophene and 2-iodotriphenylene (see fig. S8). In the first case, despite the strong noise in the images and the white lines crossing the images diagonally, we obtain a perfect prediction. In the second case, with images covering a tip-sample distance range of 72 pm (smaller that the 100 pm range used for the training), the prediction performed by the model is "2iodtriphenylene", which is very close to the ground truth, just missing a hyphen but providing all the relevant chemical

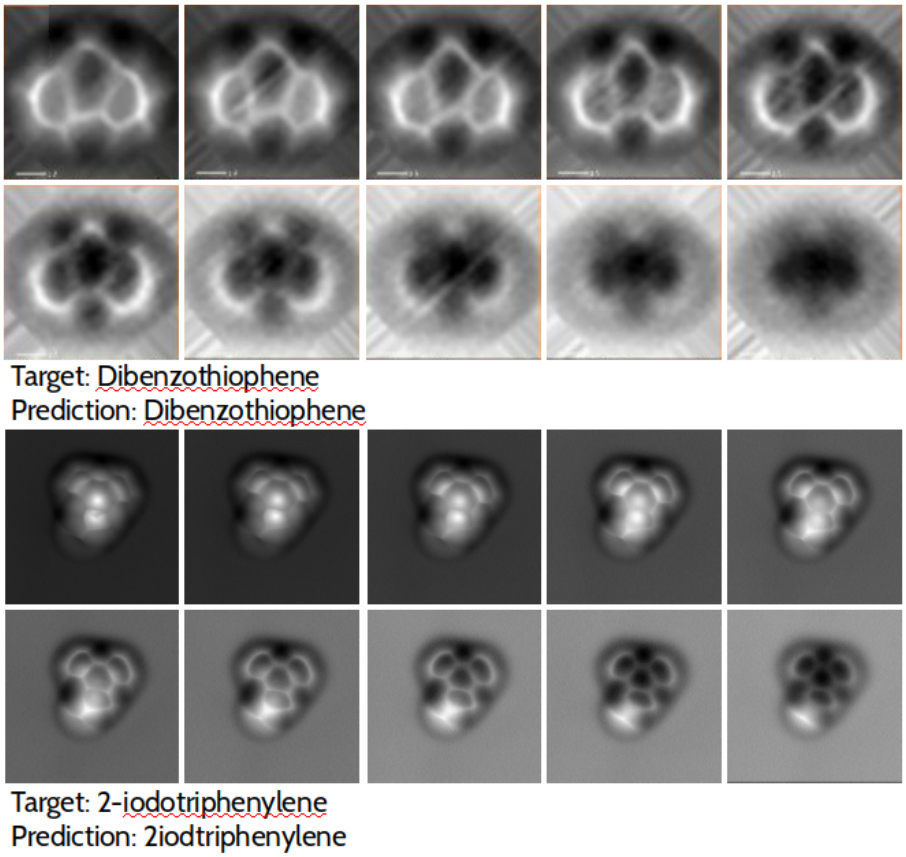


Figure S8: Experimental images used to test the molecular identification with our model. The top set of images corresponds to the 10 images of dibenzothiophene taken at 10 different tip-sample distances (covering a height range of 100pm) in ref. (*18*) (reproduced courtesy of the American Chemical Society, ACS), using a bias voltage $V = 40$ mV and a tip oscillation amplitude of approximately 50 pm. Images were aligned and Laplace filtered in the original publication to highlight the molecular structure while maintaining the feature shapes as well as possible. The bottom set corresponds to 10 constant–height frequency shift images of 2-iodotriphenylene, covering a height range of 72 pm (with a height variation of 8 pm between two consecutive images), published in the supplementary material of ref. (*19*) (reproduced courtesy of the American Physical Society, APS). A bias voltage $V = -0.57$ mV and an oscillation amplitude of $A = 52$ pm have been used for the measurements.

information. Notice that, in both cases, experimental images have been scaled close to the size of the unit cell used in QUAM-AFM before feeding the model. Without this scaling, the score in the 4-gram evaluation falls dramatically for both cases, suggesting that the typical sizes of chemical moieties learned during the training are an important component in the success of the identification process. This sensitivity to variations in image sizes can possibly be reduced by increasing the zoom range of $\pm 15\%$ applied in our data augmentation during the training.

S6 Embedding

The weights of the embeddings of the RNN components of models applied to image captioning (*20,21*) are usually pretrained with Word Embeding to represent the semantic high-level features in a vectorial space. The usefulness of this technique lies in the representation of words in a vector space, in which words with similar semantic meaning are represented close together, as shown in fig. S9. In this way, the models manage to use synonyms providing versatility and improving its expressive capacity. The versatility of languages to express the same idea in different ways is further exploited in image captioning providing the same input image with different annotation outputs. Then those models succeed in learning that different words have

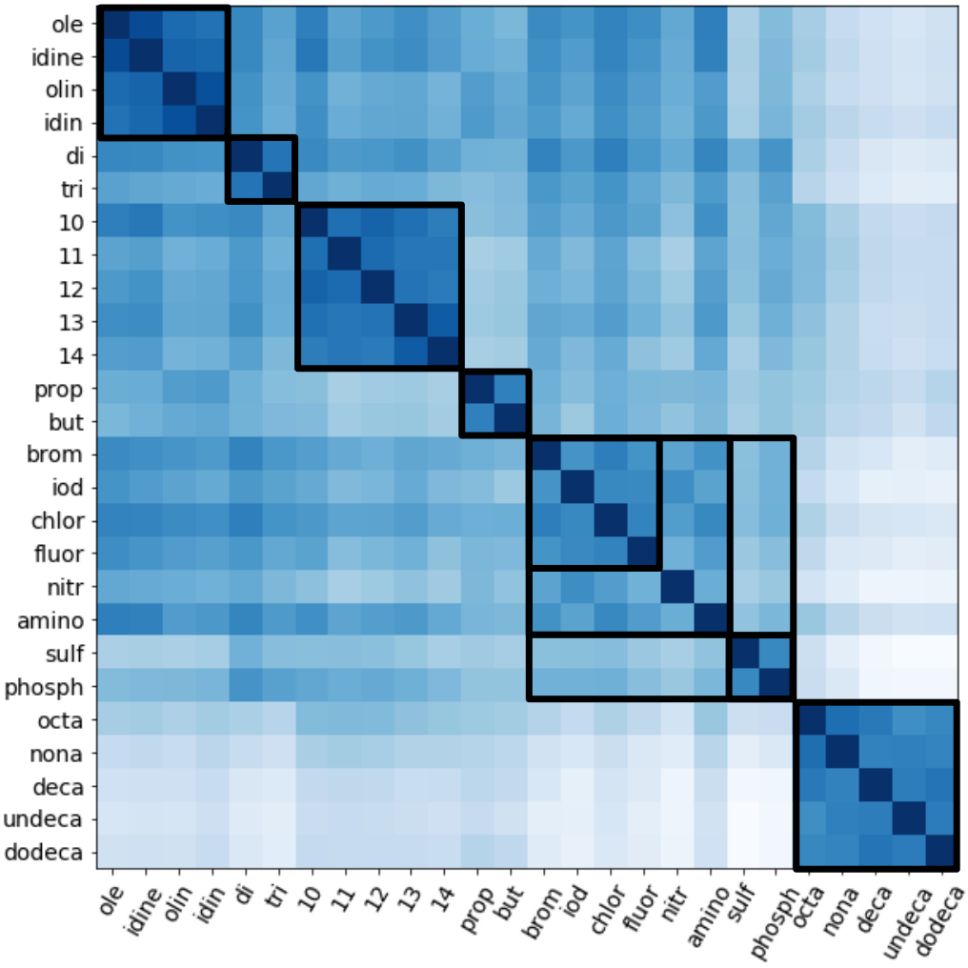


Figure S9: Distances in the L2 sense between some of the terms in the AM-RNN embedding layer. The higher intensity of blue corresponds to smaller distances while the lighter ones correspond to longer distances. The distance from a term to itself is zero, so it matches the darkest blue.

similar meaning.

However, there are no trivial synonyms in our case, due to the close relationship between the IUPAC names assigned to the functional groups in a molecule. Thus, any change in the output sequence will result likely in an error, as a different molecule would be predicted. Additionally, there is no freely available pre-trained Word Embeding for the IUPAC nomenclature, and therefore all weights of the RNN component are initialised randomly. Consequently, we train the embedding layer with the rest of the RNN component (stages 1 and 3 described in section S3.4). Here we find that, although the IUPAC nomenclature does not use synonyms, the representations made in this vector space have certain semantic relationships. To check this, we define the distance in the L2 sense and calculate the distances among the terms (of the AM-RNN). We find, for example, that the terms represented closest to *brom* are *chlor, fluor* and *iod*. In addition to the terms associated with halogens, those designating numbers are also represented in close proximity. It is also noteworthy that the terms closest to *nona* are *octa, deca, undeca* and *dodeca*, as shown in fig. S9. It should be recalled that each of the terms is represented by a number to feed the network, so the model is not learning lexical relations but syntactic relations.

References and Notes

1. J. Carracedo-Cosme, C. Romero-Muñiz, P. Pou, R. Pérez, *J. Chem. Inf. Model.* **62**, 1214 (2022).

2. J. Mao, *et al.*, *arXiv preprint arXiv:1412.6632* (2014).

3. D. Bahdanau, K. Cho, Y. Bengio, *arXiv preprint arXiv:1409.0473* (2014).

4. A. P. Parikh, O. Täckström, D. Das, J. Uszkoreit, *arXiv preprint arXiv:1606.01933* (2016).

5. C. Raffel, M.-T. Luong, P. J. Liu, R. J. Weiss, D. Eck, *International Conference on Machine Learning* (PMLR, 2017), pp. 2837–2846.

6. A. Vaswani, *et al.*, *Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NIPS)* (Long Beach, CA, USA, 2017), pp. 5998–6008.

7. J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, *arXiv preprint arXiv:1810.04805* (2018).

8. T. B. Brown, *et al.*, *arXiv preprint arXiv:2005.14165* (2020).

9. Y. Pan, T. Yao, Y. Li, T. Mei, *Proc. Comput. Vision. Pattern Recognit. (CVPR)* (IEEE Computer Society Press, Piscataway, NJ, USA, 2020), pp. 10971–10980.

10. C. Szegedy, S. Ioffe, V. Vanhoucke, A. A. Alemi, *31rd Proc. AAAI Conf. on Artificial Intelligence* (AAAI Press, Palo Alto, CA, USA, 2017), p. 4278–4284.

11. J. Gu, J. Cai, G. Wang, T. Chen, *32rd Proc. AAAI Conf. on Artificial Intelligence* (AAAI Press, Palo Alto, CA, USA, 2018).

12. J. Carracedo-Cosme, C. Romero-Muñiz, R. Pérez, *Nanomaterials* **11**, 1658 (2021).

13. F. Chollet, *Deep Learning with Python*, vol. 1 of *10* (MANNING, Shelter Island, NY 11964, 2015), first edn.

14. P. Y. Simard, Y. A. LeCun, J. S. Denker, B. Victorri, *Neural networks: tricks of the trade* (Springer-Verlag Berlin Heidelberg, 1998), pp. 239–274.

15. S. Ioffe, C. Szegedy, *32th Int. Conf. Mach. Learn. (ICML)* (JMLR.org, 2015), vol. 37, p. 448–456.

16. Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, *Proc. IEEE* **86**, 2278 (1998).

17. B. Alldritt, *et al.*, *Sci. Adv.* **6**, eaay6913 (2020).

18. P. Zahl, Y. Zhang, *Energy Fuels* **33**, 4775 (2019).

19. D. Martin-Jimenez, *et al.*, *Phys. Rev. Lett.* **122**, 196101 (2019).

20. Q. You, H. Jin, Z. Wang, C. Fang, J. Luo, *Proc. Comput. Vision Pattern Recognit. (CVPR)* (IEEE Computer Society Press, Piscataway, NJ, USA, 2016), pp. 4651–4659.

21. O. Vinyals, A. Toshev, S. Bengio, D. Erhan, *Proc. Comput. Vision Pattern Recognit. (CVPR)* (IEEE Computer Society Press, Piscataway, NJ, USA, 2015), pp. 3156–3164.