THE JOURNAL OF PHYSICAL CHEMISTRY A

pubs.acs.org/JPCA Article

Molecule Identification with Rotational Spectroscopy and Probabilistic Deep Learning

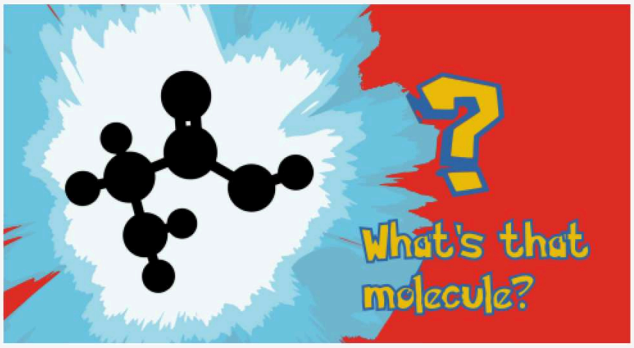
Published as part of The Journal of Physical Chemistry virtual special issue "Machine Learning in Physical Chemistry".

Michael McCarthy and Kin Long Kelvin Lee*

Cite This: *J. Phys. Chem. A* 2020, 124, 3002–3017 Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: A proof-of-concept framework for identifying molecules of unknown elemental composition and structure using experimental rotational data and probabilistic deep learning is presented. Using a minimal set of input data determined experimentally, we describe four neural network architectures that yield information to assist in the identification of an unknown molecule. The first architecture translates spectroscopic parameters into Coulomb matrix eigenspectra as a method of recovering chemical and structural information encoded in the rotational spectrum. The eigenspectrum is subsequently used by three deep learning networks to constrain the range of stoichiometries, generate SMILES strings, and predict the most likely functional groups present in the molecule. In each model, we utilize dropout layers as an approximation to Bayesian sampling, which subsequently generates probabilistic predictions from otherwise deterministic models. These models are trained on a modestly sized theoretical dataset comprising ∼83 000 unique organic molecules (between 18 and 180 amu) optimized at the ωB97X-D/6-31+G(d) level of theory, where the theoretical uncertainties of the spectoscopic constants are well-understood and used to further augment training. Since chemical and structural properties depend strongly on molecular composition, we divided the dataset into four groups corresponding to pure hydrocarbons, oxygen-bearing species, nitrogen-bearing species, and both oxygen- and nitrogen-bearing species, training each type of network with one of these categories, thus creating "experts" within each domain of molecules. We demonstrate how these models can then be used for practical inference on four molecules and discuss both the strengths and shortcomings of our approach and the future directions these architectures can take.

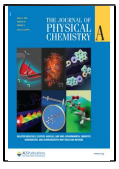


INTRODUCTION

The ability to determine the elemental composition and three-dimensional structure of an unknown molecule is highly relevant in nearly all fields of chemistry. Microwave spectroscopy has many favorable attributes in this regard because its spectral resolution is intrinsically very high and because rotational transition frequencies sensitively depend on the geometry of the molecule. For these reasons, it has been used with good success in characterizing mixtures containing both familiar and unknown species. With the development of broadband chirped-pulse methods,[1−4] microwave instruments can routinely sample an octave or more of frequency bandwidth while simultaneously achieving parts per million resolution at low pressure. Under these conditions it is relatively straightforward to distinguish between two molecules with very similar structures, and as a consequence, gas mixtures containing in excess of 100 different compounds have been analyzed[5] using highly automated experimental techniques and methodologies.[6−11]

As the throughput of microwave spectrometers continues to increase, data analysis rather than acquisition has become the primary obstacle to translating spectral information into chemical knowledge. From a scientific and analytical standpoint, the ability to analyze complex mixtures in real time would substantially improve the rate of discovery, especially with respect to identifying unknown species in heavily congested spectra. Typically, deconvolution of a mixture is performed by spectrally separating rotational transitions of an

Received: February 17, 2020
Revised: March 19, 2020
Published: March 26, 2020



https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002–3017

individual chemical constituent, isotopic species, or excited state on the basis of the small number of spectroscopic constants that are needed to reproduce its rotational spectrum. Most prominent among these are the three rotational constants ($A(BC)$), which are inversely proportional to the principal moments of inertia and thus encode the distribution of mass in three-dimensional space. Indeed, these parameters are widely used in experimental molecular structure determination through a variety of methods.[12,13]

The conventional process of identifying an unknown molecule by microwave spectroscopy involves comparing the magnitudes of the three experimental rotational constants with those predicted by electronic structure calculations for a series of candidate molecules, most of which are selected on the basis of chemical intuition. Intuition, in this case, requires consideration of the likely elemental composition, starting precursors, experimental conditions, and well-characterized molecules with similar rotational constants. For small systems, where there are relatively few possible combinations, chemical intuition can often narrow down the list of candidates quickly, and the number of electronic structure calculations required is small. For heavier and larger molecules, the number of possible structures grows rapidly with respect to composition and structural diversity. In truly unknown analytical mixtures where information is limited—such as those encountered in electrical discharge experiments[5] and astronomical observations—the combinatorics becomes intractable, and chemical intuition is both highly inefficient and incomplete with respect to capturing the full range of possible outcomes.

Machine learning (ML) is an attractive tool to assist in the identification of newly discovered molecules. At a high level, ML methodologies learn a set of parameters, θ, that are then used to estimate some property y that can help assist with the identification of a molecule on the basis of its spectroscopic data x. Here, y ideally represents a three-dimensional molecular structure, which—using rare-isotope spectroscopy—can be directly confirmed experimentally on the basis of the expected shifts in rotational constants. Other discerning factors that can be substituted for y include possible elemental composition, presence of functional groups, and the number of non-hydrogen atoms.

To identify molecules solely from available spectroscopic information, we require an ML methodology that can satisfy two criteria: first, it must encapsulate all of the possible structural and chemical space for a given set of $A(BC)$, as molecules with different compositions and structures can have similar rotational constants; second, the method must provide some estimate of uncertainty. The first criterion ensures that the method can break the partial degeneracy of $A(BC)$ where the composition is not necessarily known and may represent entirely different molecules and structures. The second criterion is necessary to infer possible carriers; it is impossible to deterministically know the exact carrier simply from $A(BC)$, and instead, it must be taken from a distribution of possible candidates.

Probabilistic neural networks[14] are an extremely felicitous class of ML techniques that provide solutions relevant to both criteria. Built on top of conventional deep learning models, which learn from a training set of data and provide the maximum likelihood estimate, probabilistic approaches ultimately yield a distribution of weighted predictions and their associated likelihoods. With a sufficiently large and diverse dataset of information, a probabilistic neural network model can be trained to transform spectroscopic parameters x into discerning information y. Formally, the problem of molecular identification then becomes that of estimating the conditional likelihood $p(y|x, \theta)$—the likelihood that an unknown molecule with parameters x can be identified with information y on the basis of learned parameters θ.

As a proof of concept for the usefulness of probabilistic deep learning in molecule identification, we combine ensembles of relatively simple neural network architectures with computationally cost-effective approximations to Bayesian sampling via dropout layers.[15] Each model within the ensemble is trained on electronic structure calculations comprising a specific chemical composition (e.g., pure hydrocarbons, oxygen-bearing molecules) as a way to break the chemical/structural degeneracy of $A(BC)$, such that each respective model becomes an "expert". In essence, each model yields conditional predictions that correspond to a particular composition, for example, predicted y for a given set of constants x if the molecule is a pure hydrocarbon. The first model we consider translates spectroscopic parameters into Coulomb matrix eigenvalues as a way of decoding spectroscopy data into machine representations that encode molecular structure and chemical properties. The predicted eigenspectra are subsequently used by three independent models that predict the possible molecular formulas, functional groups present, and SMILES encoding for a given composition. The ability to determine composition and functionalization is not only useful for identification but also deepens the connection with other analytical techniques such as mass spectrometry and infrared spectroscopy. We show that the latter, in particular, can only be accessed through our new deep learning framework and unlocks a new facet of rotational spectroscopy. The early sections of this paper will detail the expected results and performance of each model and, where applicable, comparison with a baseline ML model. In the last section, we discuss how the information from these models can be collectively interpreted in order to infer the identity of unknown molecules.

METHODOLOGY

Molecule Generation. In order to train the deep learning models, we required a dataset of molecules that span a sufficiently large volume of structural and chemical space. Initial structures were generated via two mechanisms: parsing of SMILES strings published in the PubChem database and systematic generation with the Open Molecule Generator (OMG).[16] For both cases, we systematically generated hundreds of formulas pertaining to simple organic species with an even number of electrons that constitute $H_wC_xO_yN_z$, where $1 \leq w \leq 18$, $1 \leq x \leq 8$, and $y, z \leq 3$ with $w \geq x + y + z$. As the number of isomers grows combinatorially with the number of atoms, many formulas generate up to hundreds of thousands of possible SMILES strings. Therefore, to keep the number of quantum-chemical calculations tractable, we truncated the largest lists and instead randomly sampled up to 2000 SMILES strings with uniform probability as a method of taking representative species for a given formula. Over the course of training we observed that the dataset underrepresented pure hydrocarbon species; subsequently, we bolstered the hydrocarbon set by generating isomers up to $H_{20}C_{10}$ using OMG.

Cartesian coordinates were generated from the SMILES strings using OpenBabel[17] and subsequently refined using

https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002−3017

electronic structure calculations with Gaussian 16[18] at the ωB97X-D/6-31+G(d) level of theory, optimizing the geometry corresponding to the lowest singlet state. This method was chosen on the basis of earlier benchmarking from which Bayesian uncertainties were obtained for several low-cost methods and basis sets, comparing the theoretical equilibrium rotational constants with vibrationally averaged experimental values.[19] Those results showed that ωB97X-D/6-31+G(d) provided an excellent compromise between low theoretical uncertainty, good accuracy with respect to experimental constants, and low computational cost. Additionally, as we shall see later, the uncertainties were also used to augment our training sets.

Data Preprocessing. Upon completion, the electronic structure calculation outputs were parsed, extracting relevant information such as the electronic energy, spectroscopic constants, harmonic frequencies, electric dipole moments, the corresponding canonical SMILES string using OpenBabel,[17] and the Cartesian coordinates of the molecule in the principal axis orientation. The results were then filtered to remove nonconvergent structures, transition state structures, and duplicate species by comparing the rotational constants and dipole moments. To facilitate the ensemble of models, we categorized the molecules in the dataset into four groups on the basis of their composition: pure hydrocarbons (HC), oxygen-bearing species (HCO), nitrogen-bearing species (HCN), and oxygen- and nitrogen-bearing species (HCON).

The optimized Cartesian coordinates were used to calculate the corresponding Coulomb matrix,[20] whose elements M_{ij} are defined by

$$M_{ij} = \begin{cases} 0.5Z_i^{2.4} & \text{for } i = j \\ \dfrac{Z_iZ_j}{|\mathbf{R}_i - \mathbf{R}_j|} & \text{for } i \neq j \end{cases} \quad (1)$$

where i and j are atom indices and Z_j and $\mathbf{R}_j$ are the atomic number and coordinates of atom j, respectively. The matrix maps the three-dimensional charge distribution of a molecule into a symmetric two-dimensional projection of shape $n \times n$, where n is the number of atoms. This machine representation of molecular structure simultaneously is unique in representation (apart from enantiomers) and encodes a significant amount of chemical information.

Because the experimental data typically consist of only up to eight parameters, there is a need to reduce the dimensionality of our molecular representation: a set of rotational constants is unlikely to effectively sample all possible Coulomb matrix configurations. Instead, we choose to use the eigenvalues $\lambda = [\lambda_1 \ldots \lambda_n]$ of the Coulomb matrix. While the absolute positions of atoms are lost, the maximum value and decay of the magnitude of the eigenspectrum reflect the type of atoms present, as well as the general size of the molecule: smaller molecules display "shorter" eigenspectra compared with larger species, and molecules that contain more non-hydrogen atoms exhibit slower-decaying eigenspectra with larger-magnitude eigenvalues. Despite a reduction in dimensionality, Figure 1 shows that eigenspectra can still readily differentiate between even similar molecules—fulvene is a higher-energy isomer of benzene, pyridine is isoelectronic with benzene, and benzaldehyde is a functionalized derivative of benzene. While the magnitudes of the leading eigenvalues are similar, they are differentiable particularly toward the tail end of the eigenspectra; for example, the eigenspectrum continues for benzaldehyde, whereas the eigenspectrum of pyridine truncates earlier.

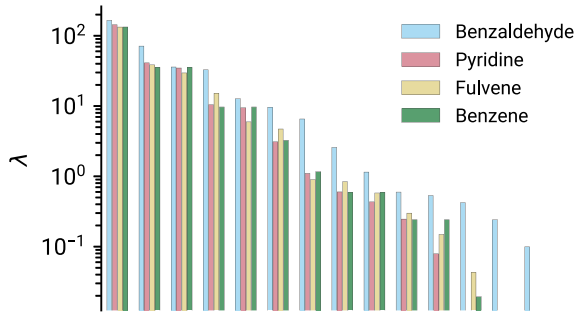


Figure 1. Comparison of the eigenspectra of four structurally and chemically similar species. The eigenspectra are truncated after the first 15 nonzero elements.

Figure 2 compares all of the pairwise Euclidean (L_2) distances between molecules within the dataset. In both the Coulomb matrix representation and the reduced eigenspectrum representation, the distributions peak far from zero and thus are expected to be readily differentiable by machine learning models. The distribution of distances is similar to those seen in other large organic molecule datasets such as QM9.[21,22]

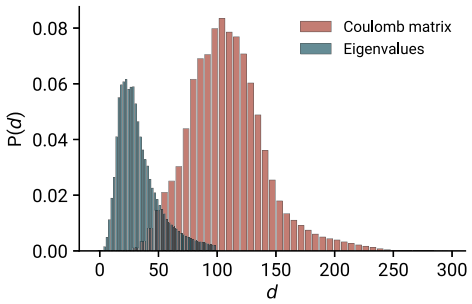


Figure 2. Pairwise similarities of molecules within the dataset as measured by the Euclidean (L_2) distance.

In order for the model to process the formula, SMILES strings, and functional groups, we converted these labels into corresponding vector representations. For each molecule, the chemical formula was encoded into a length-4 vector, where the index corresponds to the atom symbol and the value to the number of the corresponding atom. With respect to SMILES strings, we used one-hot encoding similar to that demonstrated in several other studies:[23,24] each SMILES string is encoded in a two-dimensional matrix where rows correspond to characters and each column index represents one of the 29 SMILES symbols within our dataset corpus. The first column index is reserved for blank spaces, which are used to pad shorter SMILES strings up to 100 characters. The resulting SMILES arrays have a shape of 100 × 30. Finally, the functional group labels are generated on the basis of OpenBabel canonical SMILES strings by performing functional group substructure searches with the SMARTS language implemented in RDKit.[25] The functional groups within each molecule are subsequently represented as a multilabel, "multihot" encoding. A full table summarizing the encodings can be found in the Supporting Information.

https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002−3017

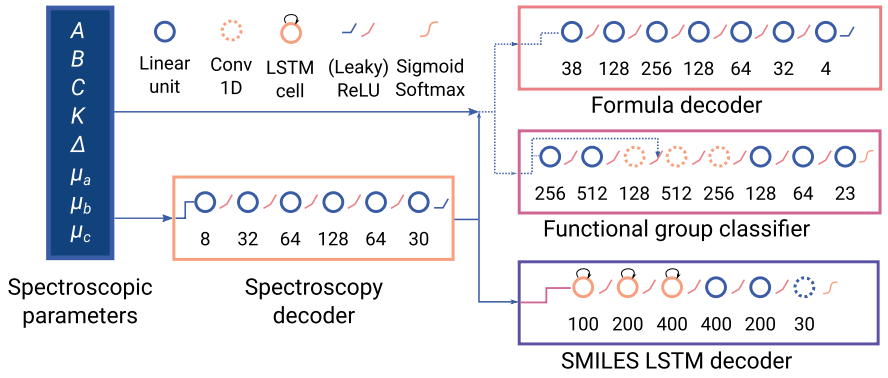


Figure 3. Graph depiction of the models considered in this work, with data flowing from left to right. Nodes represent layer types, with the corresponding output size written below each node. Blue dotted lines represent the concatenated output of the spectroscopy decoder and the parameter inputs. The pink line within the SMILES LSTM decoder model corresponds to the time-shifted sequences of eigenvalues (see the text).

Neural Network Details. The neural network models described in this work were implemented with PyTorch,[26] with training performed on an Nvidia GV100 GPU on the "Hydra" computing cluster at the Smithsonian Institution High Performance Computing Cluster (Smithsonian Institution, https://doi.org/10.25572/SIHPC). In all cases, training was performed using the Adam optimizer.[27,28] Training was performed on an 80:20 split, where 20% of the dataset was held for validation between training epochs. At the end of each epoch, the training set was shuffled, such that each minibatch was different between passes.

To improve generalization and uncertainty in each of the models, we also adopted two augmentation strategies. First, it became apparent during development that the dataset was extremely imbalanced, despite the random and unbiased sampling approach we adopted during its creation. This was a direct consequence of the number of possible isomers for certain functional groups over others: for example, there are many more ways to form an amine (i.e., primary, secondary, tertiary) than a nitrile, which was frequently observed during inference. This is commonly encountered in multilabel classification, where there are insufficient examples of under-represented labels for models to learn from and subsequently predict. To alleviate this, we duplicated species with functional groups that had fewer than 5000 samples and added Gaussian noise to the rotational constants and dipole moments to create "new" synthetic samples to balance the dataset.

The second augmentation strategy was to apply data transformations between training epochs, which is done to mitigate overfitting and—of particular importance in our application—to decrease model overconfidence. This method is commonly used in image-based applications, whereby adding Gaussian noise or random rotations improves the effective dataset size and prevents overfitting. In our case, the rotational constants were augmented by the theoretical uncertainty associated with the electronic structure method used (ωB97X-D/6-31+G(d)): the values of $A(BC)$ were scaled by a ratio δ sampled from a posterior likelihood $p(\delta)$ that represents the spread in discrepancy between the theoretical equilibrium rotational constants and the experimental vibrationally averaged values.[19] In principle, this allowed for model training to be performed on a "vibrationally-averaged" dataset that would otherwise be too costly to compute for the entire dataset. For the Coulomb matrix eigenspectra, we included Gaussian noise scaled by an exponential decay factor that preserved the tail seen in eigenspectra.

In all of the architectures explored in this work, each fully connected layer is paired with a dropout layer: for most cases, dropout layers act as a method of enforcing regularization during training by deactivating connections through each pass according to some probability p.[29] As an alternative purpose, Gal and Ghahramani[15] showed that during the prediction phase dropout layers can empirically approximate Bayesian sampling in Gaussian processes, provided that p is sufficiently large to introduce enough stochasticity while maintaining accuracy. This approach emulates ensemble-based methods, whereby dropping different neurons with each forward pass effectively creates a subnetwork. In our regression and recurrent models, these dropout units remain active during the prediction phase as a way to estimate the model uncertainty with $p \approx 0.3$ (i.e., each layer drops around 30% of the units with each pass).

While dropout is a computationally efficient and simple way of determining uncertainty, this approach is known to underestimate model uncertainty.[30] Consequently, a single deep learning model with dropouts may not necessarily capture the full range of possible molecules based only on spectroscopic constants. As we shall see later, there are structural and chemical subtleties associated with molecules of varying compositions (e.g., oxygen-bearing species vs pure hydrocarbons) that force models to place varying importance on different parameters. To help alleviate this, we also employ an ensemble of networks—in general applications, this approach involves dividing the training data among multiple networks. As each network is exposed to a different dataset, the trained weights and biases differ, with the joint prediction having a smaller generalization error than a single network.[14,31] In our application, each network is exposed to a specific composition of molecules that fall under the four categories mentioned previously, with the goal of preserving domain specificity; that is, the same set of rotational constants can result from different chemical compositions, and this needs to be reflected in the model sampling. The premise is to learn and predict a given molecular property if the unknown molecule were to contain a particular composition.

Figure 3 shows the overall flow of data through the network models considered in this work. A user provides spectroscopic data that *can* be experimentally derived, which is then used by the network to perform inference on the range of possible molecular formulas, generate viable SMILES strings, and predict the likelihood of functional groups present. In the case of the regression models, the architectures are relatively simple

https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002−3017

Table 1. Summary of Training Parameters for the Four Models[a]

model	α	Λ	N	no. of epochs	loss	no. of parameters
spectroscopy decoder	3×10^{-3}	10^{-1}	100	80	MAE	20896
formula decoder	5×10^{-3}	2×10^{-2}	30	20	MAE	15120
SMILES decoder	10^{-3}	10^{-1}	500	30	KL divergence	579588
functional classifier	1×10^{-5}	3×10^{-1}	300	50	cross-entropy	668443

[a] α and Λ are the learning rate and weight decay, respectively, defined in the Adam model.[27,28] N is the minibatch size.

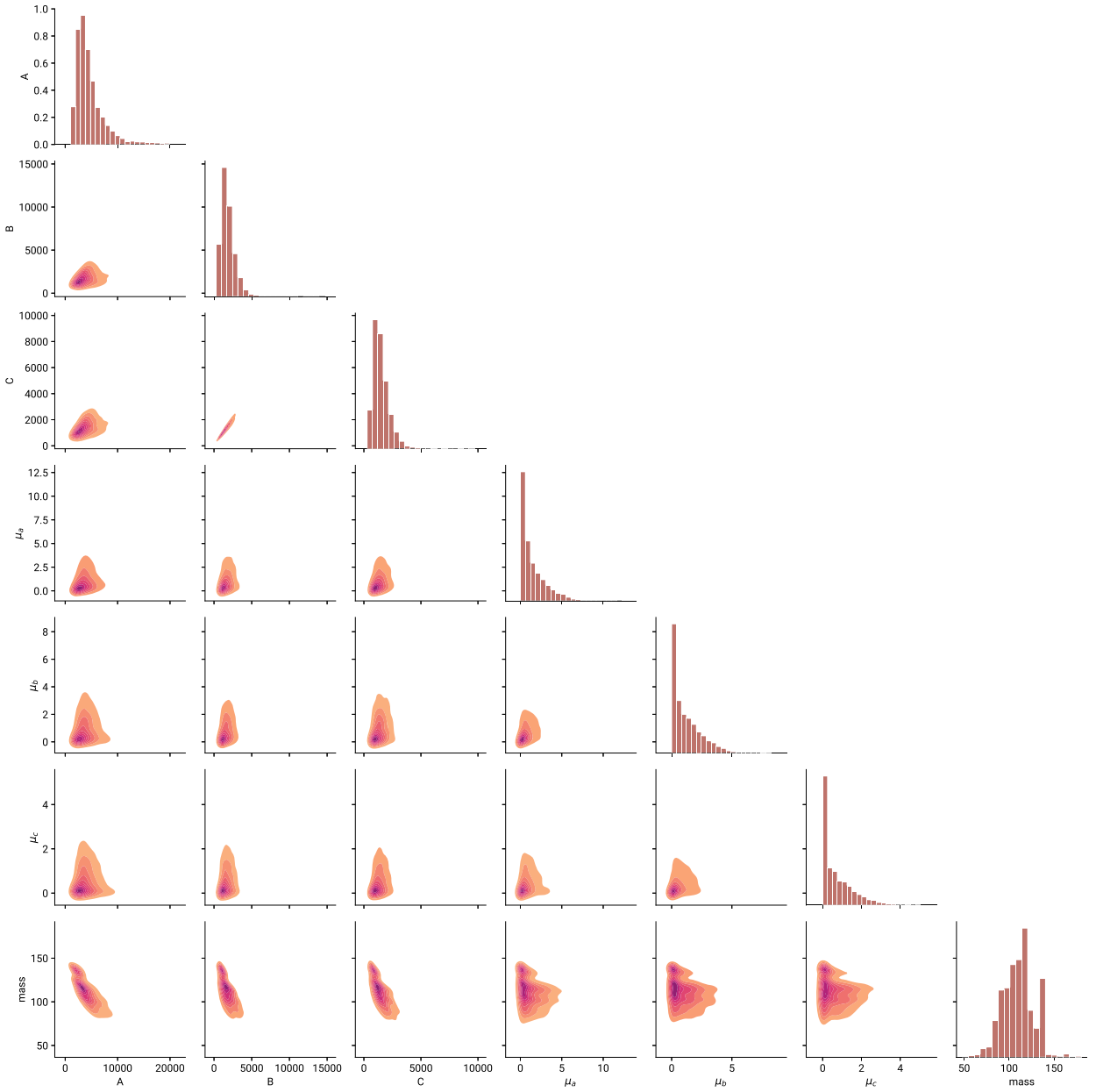


Figure 4. Visualization of the parameter space spanned by the dataset used for model training. Lighter species with rotational constants greater than 20 000 MHz are excluded in the visualization. Diagonal plots are histograms of features, while off-diagonal elements show density contours for pairs of features. Absolute values of the dipole moments are shown.

multilayer perceptrons (MLPs), up to seven layers deep and a maximum of 256 units wide, using leaky rectified linear unit (LeakyReLU) activation[32,33] with a negative gradient of 0.3. The model training is performed by minimizing the mean absolute error between the model output and the regression targets (eigenspectra λ and composition).

For the SMILES decoder, each long short-term memory (LSTM) cell used hyperbolic tangent and sigmoid functions for the cell and recurrent activations, respectively. The output of the SMILES decoder corresponds to an array of shape 100 × 30, with each row corresponding to the likelihood distribution of a given SMILES character. The model was

https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002−3017

Table 2. Summary Statistics for the Parameters Relevant to This Study[a]

parameter	mean	std. dev.	min.	P25	P50	P75	max.
A (MHz)	5623.44	9949.31	820.76	2968.19	4141.94	6169.82	673708.06
B (MHz)	1851.40	1735.75	228.55	1136.62	1592.84	2259.91	337985.92
C (MHz)	1494.23	1199.04	225.28	951.87	1302.37	1823.59	213579.84
μ_a (D)	1.45	1.53	0.00	0.33	0.93	2.06	17.73
μ_b (D)	1.17	1.18	0.00	0.25	0.81	1.74	11.81
μ_c (D)	0.72	0.80	0.00	0.09	0.46	1.14	6.59
κ	−0.67	0.37	−1.00	−0.93	−0.81	−0.55	1.00
Δ (amu Å^2)	−61.47	49.17	−388.87	−91.97	−48.27	−24.57	0.00
M (amu)	108.38	18.72	14.03	96.17	108.18	119.16	180.16

[a]κ, Δ, and M are the asymmetry parameter, the inertial defect, and the molecular mass, respectively. P25, P50, and P75 correspond to the 25th, 50th, and 75th percentiles.

subsequently trained by minimizing the Kullback−Leibler (KL) divergence:[34]

$$D_{\mathrm{KL}} = \sum_{N}\sum_{M} p(y|\lambda_m)\log\frac{p(y|\lambda_m)}{p_\theta(y|\lambda_m)} \tag{2}$$

where $p_\theta(y|\lambda)$ represents the model softmax output and $p(y|\lambda)$ the one-hot SMILES encoding for a given eigenspectrum λ. The loss is calculated by summation over N minibatches comprising M spectra. To help mitigate model overconfidence,[35] we performed label smoothing on the SMILES encoding[36] whereby the one-hot-encoded ground-truth $p(y|\lambda)$ (which is a Dirac δ function) is smoothed by weighted uniform noise $\varepsilon u(k)$:

$$p'(y|\lambda) = (1-\varepsilon)\delta_{k,y} + \varepsilon u(k) \tag{3}$$

where k corresponds to the character label, the uniform noise $u(k)$ is equal to $^1/_{30}$, and the weighting value ε is equal to 0.1. Consequently, the learning targets are no longer binary, forcing the model to produce higher-entropy/uncertainty predictions.

In the case of the functional groups, the task was to perform multilabel classification; training was performed by minimizing the binary cross-entropy loss. The architecture we propose here includes three one-dimensional (1D) convolution layers, under the premise that the convolution kernels will learn characteristic relationships between the eigenspectrum and the spectroscopic parameters. Indeed, preliminary testing with simple MLP models (without convolution) performed significantly worse than the k-nearest-neighbor baseline. In terms of activation functions, each convolution unit uses LeakyReLU (α = 0.3), whereas linear layers use parametric ReLU (PReLU) activations,[37] with the exception of the final output layer, which uses sigmoid activation. To characterize the classification performance, we computed the F_1 score[38] across the full validation at the end of training:

$$F_1 = \frac{2PR}{P+R} \tag{4}$$

where P and R are the precision and recall scores; the former measures the number of times a correct label is applied out of all attempts, while the latter reports the ratio of the number of correct labels predicted to the number of all possible examples of a given label:

$$P = \frac{N_{\mathrm{TP}}}{N}$$

$$R = \frac{N_{\mathrm{TP}}}{N_{\mathrm{TP}} + N_{\mathrm{FN}}} \tag{5}$$

where N_{TP}, N_{FN}, and N are the number of true positives, the number of false negatives, and the total number of samples, respectively.[38]

As these models represent a proof of concept, we have not extensively characterized or optimized either the hyperparameters or the architecture, with the exception of parameters encountered during training such as the learning rate and the minibatch size. The training parameters used are organized in Table 1. In terms of the number of training epochs, each model was trained until the loss appeared to have effectively converged, and there was no clear evidence for overfitting in neither the training/validation loss nor the prediction results. A large value of the weight decay (Λ) was used for each model, as it drastically decreased the model overconfidence—a known consequence of using dropouts to approximate Bayesian sampling.[39]

RESULTS AND DISCUSSION

Electronic Structure Calculations. Figure 4 shows a correlation plot of the dataset parameters. With the exception of the rotational constants and molecular mass, which are codependent, we see that all of the parameters are effectively uniformly distributed and span a representative space along their respective dimensions. The rotational constants, particularly B and C, decrease sharply with the molecular mass. The average species in our dataset is a near-prolate symmetric top ($\kappa < 0$) that is nonplanar ($\Delta \ll 0$) with nonzero dipole moments along each axis and a mass of 108 amu.

Table 2 shows the summary statistics for the dataset, which provide another perspective besides that seen in Figure 4. The mean and median (P50) are in qualitative agreement: most molecules in the dataset are near the prolate limit (i.e., $A \gg B \approx C$) according to the asymmetry parameter κ. The average molecule possesses dipole moments along all three principal axes, on the order of 1 D for μ_a and μ_b. With regard to the extremities, the lightest molecule in the dataset is CH_2, with the correspondingly largest rotational constants; the heaviest molecules considered (180 amu) correspond to a formula $H_8C_8O_3N_2$.

Spectroscopy Decoder. The first step in our approach involves taking experimental data as input and encoding them as Coulomb matrix eigenspectra, which are responsible for

https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002−3017

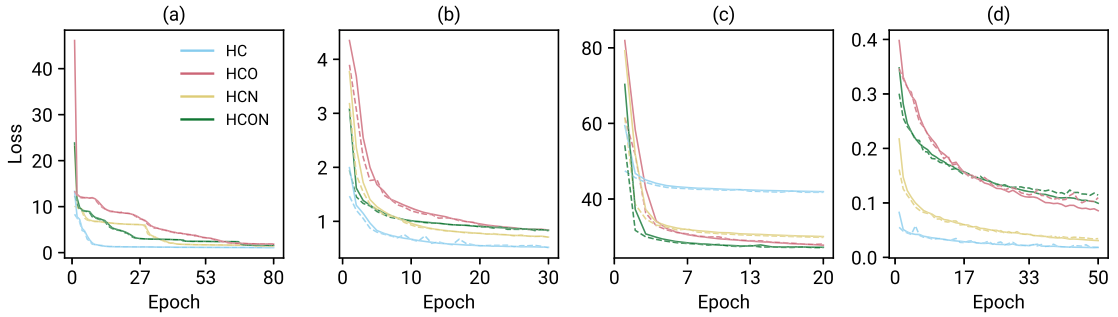


Figure 5. Epoch training (solid lines) and validation (dashed lines) losses for each of the models considered averaged across minibatches. Each color corresponds to a model composition: blue for pure hydrocarbons, red for oxygen-bearing molecules, yellow for nitrogen-bearing molecules, and green for oxygen- and nitrogen-bearing molecules. Panels (a) and (b) show the MAE loss, while (c) shows the KL divergence and (d) shows the binary cross-entropy.

translating spectroscopic constants into structural and chemical information. On a GV100, training over 80 epochs was completed for all four models in ∼30 min. Figure 5a shows the training progress of the decoder model over 80 epochs, where the color traces represent ensemble subnetworks trained on a particular composition. The loss profiles appear turbulent, which reflects the difficulty in conditioning the network parameters to map the spectroscopic constants to the corresponding eigenvalues. Presumably, the learning rate would be a highly critical factor in the ultimate performance of this decoder model—currently, the final mean absolute errors (MAEs) are on the order of less than 1% of the typical leading eigenvalues, which we believe provides sufficient accuracy for the subsequent decoding steps. The loss profiles suggest that the models are currently neither under- nor overfitted and thus could be readily extended in learning capacity.

As a concrete example, Figure 6 compares the ground-truth eigenspectrum for benzene (C_6H_6)—a highly symmetric (D_{6h})

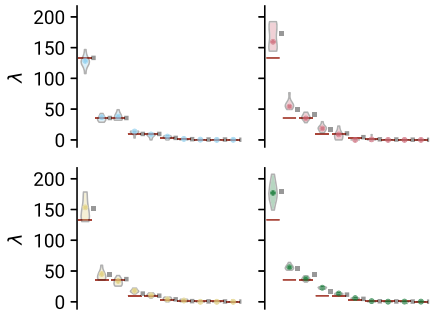


Figure 6. Comparison of the eigenspectrum of benzene (red lines) with predictions by each spectroscopy decoder model (violin plots); colors represent the same models as those in Figure 5. The thickness of each violin plot represents the distribution of values predicted after 1000 iterations of sampling. The solid circles represent the distribution means for each eigenvalue. Black squares represent predictions using *k*-nearest-neighbors regression based on five neighbors.

oblate top with $B \approx 5700$ MHz—and the corresponding model predictions obtained using the spectroscopic parameters of benzene. The violin plots represent distributions of possible eigenvalues given the input spectroscopic constants. Qualitatively, we see that the pure hydrocarbon model (blue) provides the closest match to the ground truth, which is contained within the uncertainty of each eigenvalue. The other models produce similar eigenspectra, with subtle differences in the magnitudes of the eigenvalues: for example, the oxygen- and nitrogen-bearing model (green) systematically predicts large leading eigenvalues, which reflects the type of molecules with which this model was trained.

As a point of comparison, the solid squares in Figure 6 show predictions from a *k*-nearest-neighbors algorithm as implemented in the Scikit-learn library,[40] which acts as a baseline for accuracy, using the same training process as for the neural networks. With five neighbors using the L_2 distance as the measure, we attain similar results to the neural network model means, although in the case of the HCO composition the lead eigenvalues are overpredicted. Although the accuracies of the two machine learning techniques are similar, the *k*-nearest-neighbors results are deterministic and therefore do not provide an estimate of uncertainty. Because we are interested in performing statistical inference, it is important that uncertainties between steps are propagated appropriately.

The advantage of simpler, supervised machine learning techniques is often interpretability. However, we show that the eigenspectrum decoder can still be readily interpreted with respect to the input parameters. By design, the eigenspectrum decoder should translate input spectroscopic parameters into Coulomb matrix eigenvalues, and through unsupervised training the model learns which parameters are more important or discriminating than others, which can be quantified via input gradients. Figure 7a shows the distribution of gradients for each spectroscopic parameter after repeated iterations of adding Gaussian noise into the hydrocarbon

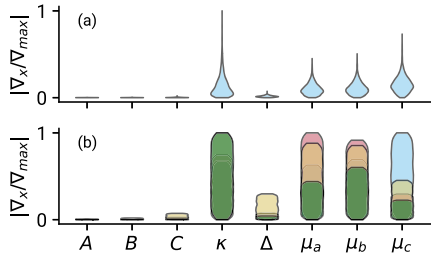


Figure 7. Violin plots of normalized unsigned gradients computed through back-propagation of the hydrocarbon model following 3000 iterations. The two panels represent (a) Gaussian and (b) uniform noise as inputs (x) to the model. Plot colors correspond to the same compositions as described in Figure 5. In panel b, the input gradients of each model composition are overlaid to show differences in model response.

https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002−3017

decoder model. Here the Gaussian noise represents some semistructured, barely semantic information, and the corresponding gradients provide an indication of how that information affects the model outputs. We see that the most informative parameters are κ followed by μ_c, μ_b, and μ_a, with the dipole moments on average providing more information than κ. This suggests that the model is most effectively utilized when the user has knowledge of the axes along which the dipole moments are nonzero, the asymmetry parameter κ, and, to a much lesser extent, the inertial defect (Δ).

Whereas Figure 7a focuses on the hydrocarbon distribution, Figure 7b shows the gradient distributions for each model composition using uniform noise as inputs, which should measure the true response of the network absent any semanticity. While dipole moments and asymmetry parameter are consistently the most defining features, each model responds quantitatively differently to each parameter as a direct consequence of the different types of bonding and structure within each composition. Perhaps most indicative of this is the importance of Δ for nitrogen-bearing molecules (yellow), suggesting that planarity is a much more defining characteristic and carries more variation for nitrogen molecules than for the other compositions. The most defining feature of pure hydrocarbons is the dipole moment along the C principal axis; structurally, this can be rationalized as an indirect measure of the number of carbon atoms along the C axis, which act as the primary source for polarization. Thus, one of the benefits of using an unsupervised ensemble learning approach is that each model can fluidly adapt to features best suited for that particular chemical composition.

Formula Decoder. Among the quantities that we wish to determine are possible chemical compositions. Following conversion of the spectroscopic data into eigenspectra, the formula decoder model seeks to predict which and how many atoms are possible for a given eigenspectrum. Figure 5b shows the training loss for the formula decoder over 20 epochs: all four models show similar loss profiles, which quickly converge by approximately 10 epochs. On the GV100, this corresponds to approximately 13 min of training time for all four models. In contrast to the spectroscopy decoder model, the learning capacity of the formula decoder model appears to be adequate, as indicated by the closely matching training and validation curves. As the models are not overfitted, it is likely that the learning capacity could be increased, and this should be considered in future architecture searches. It is important to note, however, that bias terms in the final layers were found to dominate the model outputs if unmitigated (or in our case, removed) and detrimentally affect model generalization.

Continuing with benzene as an example, Figure 8 demonstrates the performance of the combined spectroscopy and formula decoder models. Each iteration involves predicting eigenspectra corresponding to the benzene constants, whereby the spectra are then passed as input into the formula decoder model. Two general trends can be seen in Figure 8: first, the largest uncertainty is seen in the number of hydrogens; second, the number of heavy atoms is effectively conserved across the models—removal of carbon compensates for the inclusion of oxygen and nitrogen. Both observations can be interpreted in terms of the physical properties learned by the decoder models. In the former case, hydrogen atoms are significantly lighter and therefore do not contribute much to the magnitude of rotational constants, which is appropriately reflected with a correspondingly large uncertainty. The latter trend sees that all

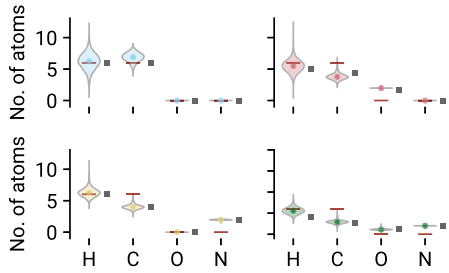


Figure 8. Predictions of chemical composition by each respective formula decoder model. Red lines represent the ground truth (C_6H_6). Scatter points correspond to the expected values for each atom type after 2000 samples. Colors refer to the same model compositions as in Figure 5. Black solid squares indicate predictions from k-nearest-neighbors regression based on five neighbors.

four models conserve the effective combined mass of the molecule: there are a limited number of ways that mass can be distributed to yield the same set of rotational constants within the constraints of atomic composition and mass. In all cases, the expected number of heavy atoms is roughly six, which matches that of benzene. These two observations not only lend confidence to the performance of the model but, more importantly, indicate that the model predictions can be rationalized with chemical intuition. While the formula decoder has marginally less accuracy than the baseline k-nearest-neighbors model (black solid squares), we believe that the ability to interpret the model uncertainty in terms of molecular structure is invaluable when identifying unknown molecules.

A complementary interpretation of the predicted formulas is to generate synthetic "mass spectra", as shown in Figure 9,

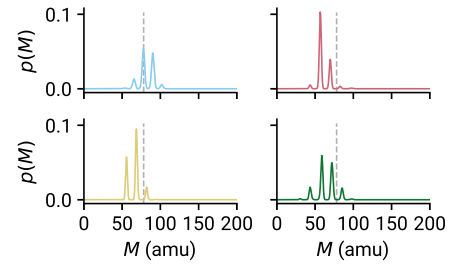


Figure 9. Simulated mass spectra based on the quantized compositions in Figure 8 predicted by the (a) pure hydrocarbon, (b) HCO, (c) HCN, and (d) HCON models. The probability distributions were obtained by Gaussian kernel estimates with a bandwidth (σ) of 1.5 mass units. The dashed lines indicate the mass of benzene (78.11 amu).

which can be helpful when assaying unknown mixtures that have available mass-resolved (e.g., mass spectrometry) data for comparison. The predicted compositions shown in Figure 8 are quantized and used to calculate the molecular mass. Kernel density estimation is subsequently used to predict the likelihood of a given mass. The mass spectra can be interpreted in two ways: the maximum likelihood estimate (MLE) gives a point estimate of the most likely mass that corresponds to the input spectroscopic parameters, while the distribution reflects the uncertainty in the model. In the case of the pure hydrocarbon model, the MLE predicts a mass close to the ground truth (~79.6 amu vs 78.11 amu), and masses with more or fewer than six carbons are considerably less likely. The

https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002−3017

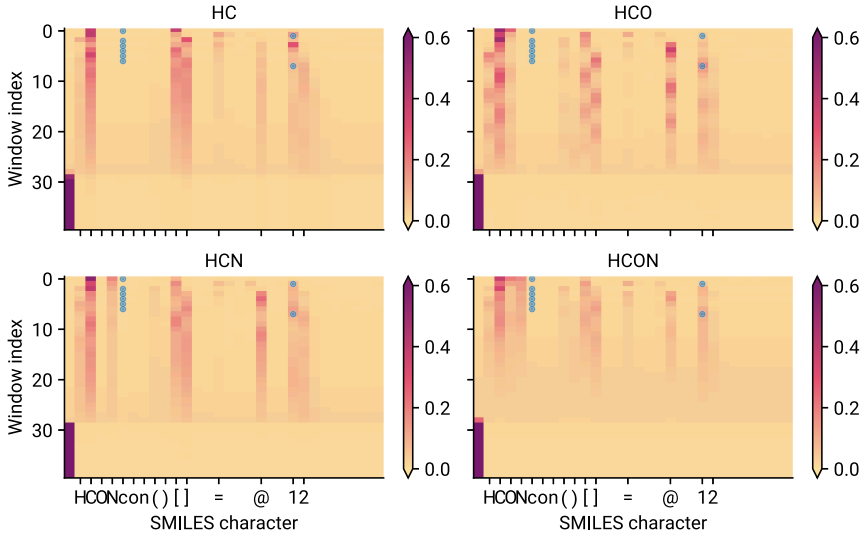


Figure 10. Heat maps of SMILES character probabilities predicted using the spectroscopic parameters of benzene averaged over 2000 samples, truncated for the first 40 sequences. The four panels represent the same model compositions as in previous figures, as indicated in the panel titles. In each panel, the abscissa and ordinate correspond to the SMILES character encoding index and the sequence window index, respectively; the first SMILES encoding corresponds to an empty character. Progressively darker colors correspond to higher probabilities for a given symbol. Blue circles indicate the ground-truth encoding for benzene (c1ccccc1).

HCO and HCON models yield MLEs near the target but do not reproduce the mass of benzene exactly; the eigenspectra encode structure and composition, and the offset masses pertain to possible structures of a given composition that *can* correspond with the specified parameters.

SMILES Decoder. Figure 5c shows the loss—as the minibatch mean KL divergence—for the SMILES LSTM decoder model training. Each model appears to converge quickly, reaching an asymptote within several epochs, and required ∼50 min of computation on a GV100. The ultimate training and validation accuracy of each model is quite exceptional: the worst performer, the pure hydrocarbon model, yields a KL divergence averaged across the entire sequence of ∼0.15, which is close to the minimum possible value of zero.[34] Thus, according to this loss metric, the model is able to reproduce the long sequences of encodings accurately without under- or overfitting.

Where the composition is a helpful quantity, the ultimate goal is to determine possible structures that can be assigned to the spectroscopic parameters. There are a variety of formats in which this information can be conveyed, for example, as simple Cartesian structures, or as reconstructions of the Coulomb matrix based on the predicted eigenvalues, or as string identifiers such as SMILES[41] and InChI.[42] The string representations are particularly attractive encodings because they are machine- and human-parsable and in certain forms (e.g., canonical SMILES) can discriminate enantiomers. For our purposes, we chose canonical SMILES as a target because of its simplicity: in contrast, the syntax for InChI is extremely specific and unlikely to be fully reproduced with the limited amount of experimental information. SMILES strings, even when incomplete, can be used to infer likely functional groups and, with programs such as OpenBabel, can be used to generate initial-guess Cartesian structures for subsequent optimization with electronic structure methods. Because of its wide use in cheminformatics for drug discovery and reaction screening, there have been multiple applications of deep learning that utilize SMILES; recurrent approaches such as LSTM[43] and GRU architectures[44] are best suited for sequence-to-sequence translation, whereby one SMILES string is used to predict another.[45]

In our application, we convert sequences of eigenspectra into SMILES characters with LSTM units: each window of eigenvalues is used to predict the likelihood of each symbol within our SMILES corpus, and through the recurrent nature of the LSTM architecture, the hidden outputs of each window are used to predict the likelihoods of following windows. The rationale is to recover nuances of SMILES syntax; for example, a closing bracket may appear several or many characters after an opening bracket, which indicates side branching in a chain. Similarly, a closing bracket should not appear prior to an opening one. Figure 10 visualizes the outputs from the SMILES decoder model based on benzene parameters, truncated to the first 40 sequence windows: the heat maps represent the averaged likelihood of a character within our corpus (abscissa) for a given sequence window (ordinate). These averages are useful for illustrating what semantics are learned by the LSTM model. We see that in all four compositions, the string terminates at a sequence length of approximately 30 characters, whereby the likelihood maximizes on the whitespace character. This indicates that the model learns an appropriate length and complexity of a SMILES string from its eigenspectrum. Another general observation is that the most likely character in early windows regardless of model composition is aliphatic carbon (C)—because all molecules contain mostly carbon, associating a high likelihood with carbon becomes inevitable. Later in the sequence, other characters become more likely, including other elements and bonding specifications. One of the more important features is that the ordering of parentheses appears to be successfully learned by the model, whereby the likelihood of a closing bracket is zero initially and remains zero until an opening bracket has a nonzero probability of appearance—this is the intended consequence of using a LSTM model.

The major obvious shortcoming of our model, however, is that it fails to reproduce the SMILES code of benzene

https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002−3017

Table 3. Four SMILES Strings with the Highest Conditional Likelihoods for Each Model Based on 2000 Iterations of Sampling, Decoded with the Beam Search Algorithm (Predictions Are Based on the Spectroscopic Constants of Benzene)

HC	HCO	HCN	HCON
CCCCCCCCCCCC	CCCCCCCCCC	cCCCCCCCCCCCCC	OCCCCOO
CCCCCCCCC	OCCCCCCCCC	CCCCCCCCCCCCCC	NCCCCOO
CCCCCCCC	OCCCCCCCCCC	NCCCCCCCCCCCCCCC	CCCCOOO
CCCCCCCCCCCCC	OCCCCCCCCCCC	nCCCCCCCCCCCCCC	OCCCCONC

(c1ccccc1, indicated by the blue circles). Upon inspection of our training set, it appears that the aromatic carbon symbol is significantly underrepresented, as only ∼3200 molecules contain it, and therefore it is unsurprising that little to no likelihood is predicted by the models. On a more general note, the encoding for benzene is highly unique because of its molecular symmetry (D_{6h}), and thus, it is unlikely that a generalized LSTM model can successfully reproduce the specificity required for molecules like benzene; in other words, there is "no free lunch".[14,46] This example highlights the limitation of our SMILES decoder model, where highly symmetric—and typically small—molecules are poorly reproduced because of their high specificity and symmetry compared with large asymmetric species. We contend, however, that these molecules are the most difficult to identify and in need of an inferential approach contrasting the smaller molecules that can be more readily deduced via combinatorial searches.

Following calculation of the character likelihoods, we employed a beam search algorithm to decode the sequences into SMILES strings. This was performed by starting with n of the most likely characters at the beginning of the sequence and finding characters along the sequence that maximize the conditional likelihood. In spite of this, we found many of the resulting strings to have invalid SMILES syntax, particularly with respect to the placement and ordering of parentheses, and often to be chemically vague. Another issue we observed concerns the coherence time of the sequences: in many samples, the character likelihoods decay gradually into approximately uniform likelihoods, as the eigenvalues are effectively zero and the LSTM model fails to produce any information. This criterion is used during the beam search, where the sequence likelihood is compared against a uniform distribution using the KL divergence: as it approaches zero, sampling is terminated early to prevent oversampling from uninformative sequences.

As can be seen in Table 3, the strings with the highest conditional likelihood are unfortunately chemically and structurally uninformative. The most striking issue is that aliphatic carbon is significantly oversampled, most likely because the dataset contains organic molecules and—with little information available—the most likely character within a SMILES sequence will be carbon. Another problem is the length of the sequences: even with early termination, the sequences produced are far too long to match the rotational constants of benzene. In the models containing oxygen and nitrogen, we see that these elements are incorporated into the sequence, albeit with extremely low likelihood (e.g., CCCCOOO in the HCON model). To improve this approach, future attempts should consider changing different aspects of the problem. For example, the eigenspectrum is not necessarily an optimal feature representation to decode into SMILES strings, and decoding could be advanced by projection onto a more informative space (i.e., principal components) or other machine-readable representations.[47,48] The neural network architecture could also be substantially improved, for example by using transformer architectures.[49] Finally, the information content of SMILES could be encoded in different ways, such as lossless compression.[50] While we used a one-hot approach that was successfully demonstrated by other groups[24,45,51] for direct SMILES-to-SMILES translation, it is likely that the uncertainty is too high in our application for unique and informative mapping. Various forms of SMILES compression, such as DeepSMILES,[45] would greatly simplify the encoding complexity and decrease the machine learning requirements—an avenue for future exploration.

Functional Group Classification. As the SMILES LSTM decoder—in its current state—was unable to produce useful information for molecular identification, we investigated the possibility of simpler yet indicative sources of information. Combining 1D convolution and linear layers, we built a model that uses the eigenspectra and the spectroscopic parameters to perform multilabel classification, which predicts the likelihood that selected functional groups are present. This is premised by the fact that the parameters, in particular the dipole moment vectors, contain some information about functional groups that are the primary drivers for polarization in a molecule. Combined with the eigenspectra, there should be sufficient information to reliably distinguish between similar yet different functional groups (e.g., OH groups within carboxylic acids and primary alcohols).

Figure 5d shows the training and validation binary cross-entropy profiles over 40 training epochs. On a GV100, model training took approximately ∼11 min to complete. Once again, the training and validation losses are nearly identical, indicating that the models are neither over- nor underfitted. The hydrocarbon model demonstrates exceptionally low loss, which is ascribed to low chemical complexity, as few functional groups are possible. While the HCO and HCON models show the largest loss values, considering the full breadth of possible functional groups (15 labels for the former and 23 for the latter), we believe that each model is performing within the full capacity of the architecture.

In multilabel classification, the binary cross-entropy alone is not informative of the model performance. Using k-nearest-neighbors as an unsupervised baseline classifier, we performed approximately the same multilabel classification task as with the neural network model. For comparison, we use the F_1 score, which is the harmonic mean of the precision and recall scores; the former measures the number of times the correct label is predicted out of the total number of samples, whereas the latter is equal to the number of times the correct label is predicted divided by the number of examples of that label. An F_1 score of unity represents the case where every label was correctly predicted at every possible instance, and not simply from random chance.

Figure 11 compares the F_1 scores calculated by the two approaches for each functional group within a composition. In

https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002−3017

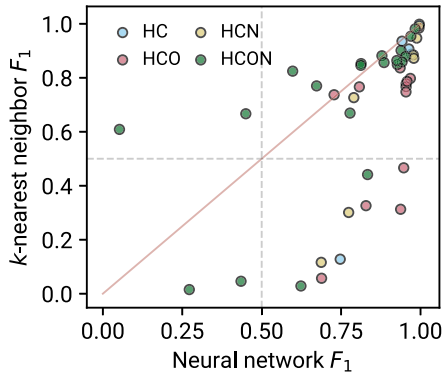


Figure 11. Comparison of validation F_1 scores from the neural network approach (abscissa) and a *k*-nearest-neighbors classifier (ordinate). Each scatter point corresponds to a functional group encoding, and colors represent the compositions used to train the respective models. The solid red trace indicates where both models perform equally well.

many cases, both classifier models show excellent performance (top right quadrant) where F_1 is close to unity. However, there are several functional groups within the HCON composition that are not predicted well by either classification model, namely, vinyl groups, carbonyls, and alcohol groups. Table 4 shows the worst-performing functional groups with respect to F_1 scores: we see that the neural network has consistently higher recall scores than precision scores, indicating that these functional groups are subject to false positives. In comparison, the *k*-nearest-neighbors approach results in higher precision scores, albeit with significantly lower recall scores and more reluctance to predict these groups. It is likely that the features are weakly discriminative with respect to these oxygen functional groups compared with their nitrogen counterparts, which correspond to much higher F_1 scores. This is true for the allene functional group, which suffers from consistently lower predictability across all four compositions by both classification models.

Table 4. Lowest 10 F_1 Scores for the Neural Network Approach, Comparing the Precision (P) and Recall (R) Scores for Both Classification Models

		neural network			*k*-nearest-neighbors		
model	functional group	*P*	*R*	F_1	*P*	*R*	F_1
HCON	allene	0.08	0.21	0.05	0.78	0.50	0.61
HCON	vinyl	0.38	0.63	0.27	0.46	0.01	0.02
HCON	ether	0.49	0.56	0.43	0.43	0.02	0.05
HCON	amino acid	0.52	0.63	0.45	0.84	0.55	0.67
HCON	alkyne	0.67	0.76	0.60	0.86	0.80	0.82
HCON	carboxylic acid OH	0.65	0.69	0.62	1.00	0.01	0.03
HCON	peroxide	0.73	0.80	0.67	0.78	0.76	0.77
HCN	vinyl	0.70	0.72	0.69	0.67	0.06	0.12
HCO	phenol	0.72	0.76	0.69	0.50	0.03	0.06
HCO	allene	0.74	0.75	0.73	0.78	0.70	0.74

The key observation from Figure 11 is the superior performance by the neural network classifier. We can conclude that the neural network approach is better suited for molecular identification, with three distinct advantages over the baseline model: (1) improved precision and recall in all except one functional group, (2) uncertainty quantification through dropouts, and (3) portability and scalability. The last advantage is particularly important toward real-time inference; the *k*-nearest-neighbor classifier needs to traverse the full training set (9 GB of data) for inference, thus scaling poorly with the dataset size and limiting portability. On the other hand, the neural network classifier is significantly compressed (~2.6 MB on disk) and can be used in distributed systems and GPUs.

In terms of the performance of the neural network, Table 5 shows the top 15 F_1, precision, and recall scores with their respective compositions and functional groups. These metrics show that in the best-case scenarios, the classifier is able to predict the presence of a functional group with ~85% precision simply from a set of spectroscopic parameters. Most importantly, it is difficult to establish a human judgment baseline, as it is highly unlikely that an expert would be able to derive such information simply by inspecting rotational constants and dipole moments. This is extended to the vast majority of the functional groups included in our study: Table 4 shows the worse performers with respect to F_1 scores, such that >75% of the predictors are accurate to 70%.

Table 5. Top 15 Performing Functional Groups and Their Associated Statistics for Each Neural Network Classifier Composition, Based on the Validation Dataset

model	functional group	*P*	*R*	F_1
HCN	aromatic carbon	0.96	0.93	0.99
HCN	alkyne	0.84	0.74	0.98
HCN	nitrile	0.98	0.97	0.98
HC	allene	0.84	0.75	0.97
HCON	aromatic carbon	0.96	0.95	0.97
HCO	peroxide	0.96	0.95	0.97
HC	alkyne	0.92	0.88	0.96
HCO	aldehyde	0.90	0.84	0.96
HCO	carbonyl	0.95	0.94	0.95
HCO	carbonyl-carbon	0.94	0.93	0.95
HCO	ketone	0.88	0.82	0.95
HCON	carbonyl	0.91	0.86	0.95
HCO	alcohol	0.92	0.90	0.95
HCON	carbonyl-nitrogen	0.86	0.79	0.94
HC	aromatic carbon	0.92	0.90	0.94

Figure 12 continues to use benzene as a demonstration, where each panel shows the predicted functional groups for a given composition. Because of the large number of labels, we refer the reader to Table 6 for a list of the labels within each label group. The output of this classifier, as shown by each bar, predicts the likelihood that a particular functional group is present in the molecule given the Coulomb matrix eigenvalues and spectroscopic parameters. A full ordered list of the functional groups is given in Table S1. In the case of the pure hydrocarbon model, the most likely groups predicted are aliphatic carbon and vinyl groups, followed by aromatic carbon and then alkyne with much lower probability and allene as the least likely. Although the model incorrectly ascribes lower probability to the correct (aromatic carbon) label, it does infer a high likelihood of unsaturation via the vinyl group. On the other hand, the nitrogen (yellow) and mixed (green) models predict a high likelihood of aromaticity. Interestingly, the oxygen-bearing (red) model predicts a large likelihood for

https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002−3017

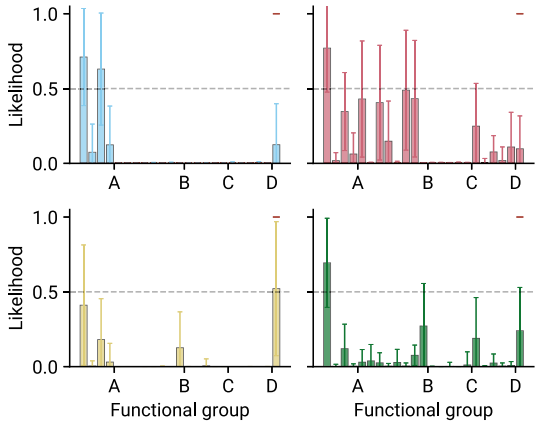


Figure 12. Predicted mean likelihoods of each functional group by the four model compositions. Error bars represent 1σ in model uncertainty. The dotted lines mark an arbitrary cutoff of 50% likelihood. The absicssa labels represent types of functional groups to the left of the label (see Table 6): (A) carbon saturation, (B) carbonyls, (C) nitrile/nitro, (D) alcohol/acid. The last group corresponds to aromatic carbon.

Table 6. Ordering of the Functional Group Labels

label group	functional group
A	aliphatic
	allene
	vinyl
	alkyne
B	carbonyl
	carbonyl-nitrogen
	carbonyl-carbon
	aldehyde
	amide
	ketone
	ether
C	amine
	amino acid
	nitrate
	nitro
D	alcohol
	carboxylic acid
	enol
	phenol
	peroxide
no label	aromatic carbon

many functional groups, particularly those pertaining to carbonyls (between A and B).

The probabilistic approach adopted here allows a user to consider not only the probability of a functional group but also the model confidence. Furthermore, because the labeling is generated by matching SMARTS substructures, one can easily create arbitrarily specific functional group classification schemes; in the current implementation we chose to use quite general SMARTS coding to maximize coverage, but this could be tuned to produce highly specific labels (e.g., heteroatomic ring structures). In the proceeding sections, we will discuss how predictions from each of the models can be combined to infer the identity of an unknown molecule or at least suggest tests to be conducted.

Example Applications. In this section, we will apply the formula and functional group decoder models to four known molecules in order to demonstrate our anticipated workflow/thought process. Generally speaking, the formula decoder sets the boundaries for viable compositions and, combined with the predicted functional groups, should significantly limit the search space. We note that these examples were not chosen on the basis of their performance but rather as a way to highlight the strengths and weaknesses of the models outlined in this work and how the predictions from each model can be combined to piece together information about an unknown molecule. Figure 13 shows the predictions for four different species by the formula decoder and functional group classifier for each model composition. Inference with all four models was performed on an Nvidia GV100 GPU with 5000 samples per molecule at approximately 4−6 s/molecule.

Starting with cyanophenylacetic acid ($C_9H_7NO_2$), an aromatic molecule with nitrile and carboxylic groups, we see that the number of atoms predicted by the HCON formula decoder (green) is fairly accurate, although the number of nitrogens is overpredicted and not captured by the model uncertainty, thereby showing that the model remains overconfident in spite of data augmentation. The corresponding HCON functional group classifier correctly predicts six out of seven groups—missing only the carboxylic acid group, which on the basis of the F_1 scores in Table 4 is one of the groups that is poorly captured by the HCON classifier. Additionally, there are three other false-positive predictions: a vinyl group, a nitrogen atom α to a carbonyl, and an amide group. This result reinforces the fact that the current model implementation is more likely to generate false positives (i.e., low precision scores).

The next example, aminobutyne, is a typical unsaturated nitrogen-bearing molecule. In this case, the HCN formula decoder overestimates the number of nitrogens, although it captures the number of hydrogens and carbons perfectly. The functional group classifier correctly predicts the presence of aliphatic carbon, an alkyne group, although it ascribes a low likelihood to an amine group. Unfortunately, this is an example in which the functional group classifier is misleading in its prediction: from this, we recommend that these classifier models be used to guide what groups *may be present* rather than completely ruling out groups entirely.

Propanediol is an example where both the formula decoder and functional group classifier provide accurate predictions. In the latter, both the aliphatic content and alcohol functional groups are correctly predicted, along with a false-positive ether group. We see here that each model composition recognizes the highly saturated nature of the input species—predicting low likelihoods for unsaturated groups (e.g., vinyl, alkene, etc.) and dominated solely by aliphatic carbon. This example highlights how predictions from each composition can jointly inform the user what common functional groups are present.

Finally, fulvene is an isomer of benzene (C_6H_6). The formula decoder once again captures the number of atoms well, although the expected number of carbons is slightly higher than the actual number. In contrast to the benzene example (Figure 12), none of the models predict a significant likelihood of aromatic carbon being present and instead see a high likelihood of unsaturated alkenes (compared with the propanediol result).

On the basis of the four examples, we can conclude three aspects that will guide the interpretation of these models. First, the formula decoders are likely to underestimate the total number of non-hydrogen atoms, although they constrain the

https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002−3017

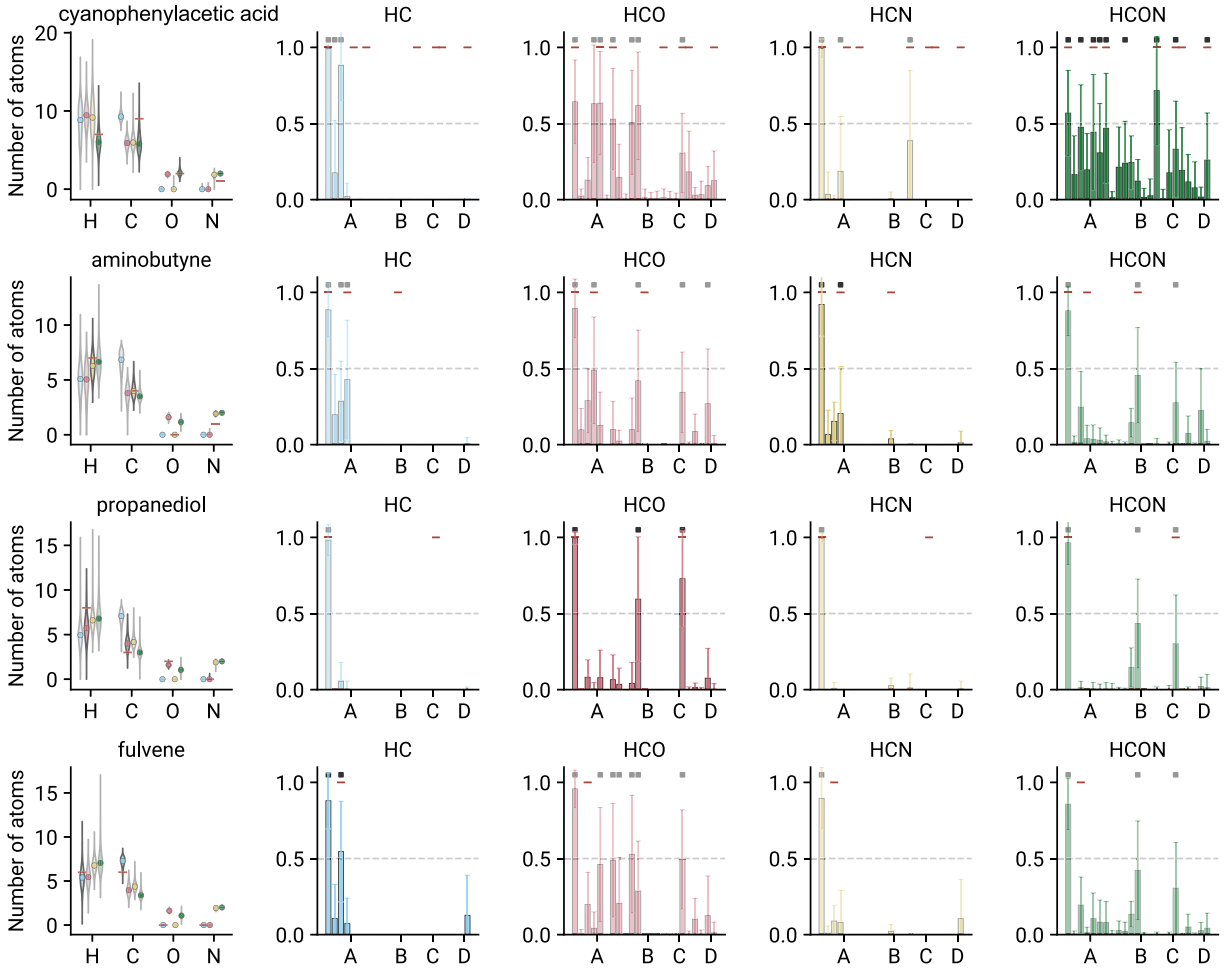


Figure 13. Mosaic of the predicted distributions of molecular composition (leftmost panels) and likelihoods of functional groups for four selected species. In each panel, red lines indicate the ground truth. In the functional group predictions, bars represent the mean prediction with 1σ uncertainties shown in the error bars, and black squares indicate functional groups with likelihoods greater than 0.5. Darker shading corresponds to the correct model composition. The absicssa labels represent types of functional groups to the left of the label: (A) carbon saturation, (B) carbonyls, (C) nitrile/nitro, (D) alcohol. The last group corresponds to aromatic carbon.

possible formula to within an atom. When considering the possible range of formulas, it is therefore recommended to test the mean formula first, followed by modifications to the numbers of heavy atoms (in particular oxygen and nitrogen) according to the uncertainty of each atom. Because the uncertainties are often underestimated—as can be seen in the nitrogen and oxygen predictions—we recommend extending the sampling of the number of atoms by ±1 beyond the limits of the uncertainties. Second, the functional group classifiers appear to be more likely to produce false positives than false negatives, on the basis of Figure 13 as well as some of the precision and recall scores shown in Table 4. Thus in testing of functional groups, we recommend prioritizing the high-likelihood functional groups that fall under the composition constraints and systematically ruling each group out over the course of the identification process. These can be confirmed experimentally often by rare isotopic substitution, for example by shifting alcohol groups with deuterium. Third, when the composition is unclear, it is important to consider predictions from all four compositions, in particular functional groups that are common to other compositions. The most decisive trend involves saturated/unsaturated species: in Figure 13, unsaturated species are predicted to have unsaturated content regardless of the model composition, while saturated species generally result in no unsaturated groups at all (as in the case of propanediol).

Model Considerations and Limitations. In the examples provided so far, the models are provided a complete set of spectroscopic parameters with absolute precision. In real applications, this may not always be the case, for example, when combinations of parameters are being used to fit effective Hamiltonians (e.g., $B + C$ for a prolate symmetric top) or when the dipole moments are not known. The advantage of our probabilistic approach is the ability to perform inference even under these circumstances: because each model provides an estimate of the conditional likelihood $p(y|x)$, each parameter within x can simply be varied in proportion to its uncertainty and with spectroscopic intuition. To pose an example, we discuss a situation commonly encountered in our laboratory:[5] a prolate symmetric top is fit with $B + C$ without immediately obvious K structure, and only a-type transitions are measured, thus leaving A poorly constrained and μ_b and μ_c unknown. The parameters can be repeatedly perturbed with Gaussian noise weighted by the parameter uncertainty in a bootstrap fashion. Because of the probabilistic nature of our models, the uncertainties propagate from the input values through the eigenspectrum to the predicted quantities; each pass is

https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002−3017

equivalent to computing the conditional likelihood of a formula or functional group with respect to λ and $A(BC)$.

A detail that arose during the training of these models was the importance of a balanced dataset, which was particularly apparent in the functional group classifier. Despite our efforts to balance the dataset prior to training, the models produced are still susceptible to biases that are created inadvertently by unbiased sampling as we have done. This was seen, for example, in our tests on smaller molecules, which are underrepresented with respect to larger species simply because of the number of possible isomers for the latter case. Future attempts of these models will need to be highly mindful of these subtleties at the possible expense of selection bias.

One of the significant drawbacks of our approach toward probabilistic neural networks is the overuse of dropout layers: although they are necessary for the probabilistic aspect of our solution, it is likely that they over-regularize parameter learning and consequently decrease the full learning capacity of each model. In principle, one could use a reduced dropout probability during training—as long as there is no overfitting—and use a larger dropout rate for inference. There are also methods to calibrate uncertainties by empirical scaling[52] that could rectify the model uncertainties, thereby mitigating "overdropping". Regardless, dropout acts only as an approximation to Bayesian sampling, and for a truly probabilistic approach inference must be performed by sampling from posterior distributions of learned parameters. A major difficulty in implementing true Bayesian deep learning models is the computational cost associated with training and inference; every forward pass must involve sampling from hundreds to thousands of parameter distributions that replace scalar values, and every backward pass must compute, propagate, and update gradient information to the same number of parameters.[14] Bayesian networks are an active area of study, and attractive solutions are being developed, including probabilistic backpropagation,[53,54] bootstrap methods,[55] and approximate[56] and variational[57] inference. The ability to move to a Bayesian model would remove the need for an ensemble approach, which would significantly improve the ease of interpretation of the model. Here an ensemble is required because of the difficulty for single network models to generalize and be predictive with a large variety of input parameters, whereas Bayesian models are resistant to overfitting.

Overall, the proof-of-concept models we have shown here highlight the viability of probabilistic deep learning models in molecule identification with rotational spectroscopy. While there is room for improvement, the approaches we have described provide a promising framework for performing inference on unknown molecules: we can reliably constrain the possible range of compositions and functional groups present *simply from a set of eight spectroscopic parameters*. These constraints—in conjunction with user expertise—can be used to guide systematic electronic structure calculations to provide possible candidates for identification. The framework described here has significant implications for the use of rotational spectroscopy in complex mixture analysis. In addition to providing a systematic method for identification, each decoder model connects rotational spectroscopy with other analytical techniques: through the formula decoder, we are able to predict mass spectra, and with the functional group classifier, we unlock an aspect of chemistry that was not previously accessible solely with rotational spectroscopy, as functional groups are typically determined using infrared techniques. We believe that further development of this methodology will solidify rotational spectroscopy as a universal analytical tool.

CONCLUSIONS

In this work, we have demonstrated a series of proof-of-concept probabilistic deep learning models that aim to assist with molecular carrier inference. The architectures we have described are relatively simple and lightweight neural network models. In our demonstrations, we have shown that the approximate formula can be determined and functional groups that are likely to be present can be identified from spectroscopic data routinely available from broad-band chirped-pulse experiments—the spectroscopy decoder, formula decoder, and functional group classifier can be collectively used to infer discriminating factors about the unknown molecule that should systematically lead to its identification. Although the SMILES LSTM decoder could not generate sufficiently coherent SMILES sequences, our results show that the models proposed here are able to learn some of the semantics, although it is unclear whether there is sufficient specific information contained within the eigenvalues to perform a direct translation to canonical SMILES strings. Instead, it may be worthwhile to consider compressed SMILES encodings or other representations of molecular structure.

The models we have presented as part of this work are computationally scalable and, with appropriate algorithmic optimizations, could provide a step toward near-real-time unknown molecule inference. Furthermore, the probabilistic framework we have detailed can be readily accommodated to "real" situations, particularly those where certain spectroscopic parameters are highly uncertain, by the use of bootstrapped parameters during inference. We anticipate that these models will be highly invaluable in future broad-band assays of unknown complex mixtures using rotational spectroscopy.

ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jpca.0c01376.

A table containing information regarding the functional group labels (PDF)

AUTHOR INFORMATION

Corresponding Author

Kin Long Kelvin Lee − *Center for Astrophysics | Harvard & Smithsonian, Cambridge, Massachusetts 02138, United States;* orcid.org/0000-0002-1903-9242; Email: kin_long_kelvin.lee@cfa.harvard.edu

Author

Michael McCarthy − *Center for Astrophysics | Harvard & Smithsonian, Cambridge, Massachusetts 02138, United States;* orcid.org/0000-0001-9142-0008

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jpca.0c01376

Notes

The authors declare no competing financial interest.

The dataset used for the model training can be made available upon request. The Python code used to train, test, and perform

https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002−3017

inference can be found on https://github.com/laserkelvin/rotconml.

■ ACKNOWLEDGMENTS

The authors acknowledge financial support from NSF (Grants AST-1615847 and AST-1908576) and NASA (Grants NNX13AE59G and 80NSSC18K0396) and computing resources from the Smithsonian Institution High Performance Cluster (SI/HPC, "Hydra" (https://doi.org/10.25572/SIHPC)).

■ REFERENCES

(1) Brown, G. G.; Dian, B. C.; Douglass, K. O.; Geyer, S. M.; Shipman, S. T.; Pate, B. H. A broadband Fourier transform microwave spectrometer based on chirped pulse excitation. *Rev. Sci. Instrum.* **2008**, *79*, 053103.
(2) Park, G. B.; Field, R. W. Perspective: The first ten years of broadband chirped pulse Fourier transform microwave spectroscopy. *J. Chem. Phys.* **2016**, *144*, 200901.
(3) Wehres, N.; Heyne, B.; Lewen, F.; Hermanns, M.; Schmidt, B.; Endres, C.; Graf, U. U.; Higgins, D. R.; Schlemmer, S. 100 GHz Room-Temperature Laboratory Emission Spectrometer. *Proc. Int. Astron. Union* **2017**, *13*, 332−345.
(4) Finneran, I. A.; Holland, D. B.; Carroll, P. B.; Blake, G. A. A direct digital synthesis chirped pulse Fourier transform microwave spectrometer. *Rev. Sci. Instrum.* **2013**, *84*, 083104.
(5) Lee, K. L. K.; McCarthy, M. Study of Benzene Fragmentation, Isomerization, and Growth Using Microwave Spectroscopy. *J. Phys. Chem. Lett.* **2019**, *10*, 2408−2413.
(6) Crabtree, K. N.; Martin-Drumel, M.-A.; Brown, G. G.; Gaster, S. A.; Hall, T. M.; McCarthy, M. C. Microwave spectral taxonomy: A semi-automated combination of chirped-pulse and cavity Fourier-transform microwave spectroscopy. *J. Chem. Phys.* **2016**, *144*, 124201.
(7) Martin-Drumel, M.-A.; McCarthy, M. C.; Patterson, D.; McGuire, B. A.; Crabtree, K. N. Automated microwave double resonance spectroscopy: A tool to identify and characterize chemical compounds. *J. Chem. Phys.* **2016**, *144*, 124202.
(8) Zaleski, D. P.; Prozument, K. Automated assignment of rotational spectra using artificial neural networks. *J. Chem. Phys.* **2018**, *149*, 104106.
(9) Seifert, N. A.; Finneran, I. A.; Perez, C.; Zaleski, D. P.; Neill, J. L.; Steber, A. L.; Suenram, R. D.; Lesarri, A.; Shipman, S. T.; Pate, B. H. AUTOFIT, an automated fitting tool for broadband rotational spectra, and applications to 1-hexanal. *J. Mol. Spectrosc.* **2015**, *312*, 13−21.
(10) Western, C. M. PGOPHER: A program for simulating rotational, vibrational and electronic spectra. *J. Quant. Spectrosc. Radiat. Transfer* **2017**, *186*, 221−242.
(11) Riffe, E. J.; Shipman, S. T.; Gaster, S. A.; Funderburk, C. M.; Brown, G. G. Rotational Spectrum of Eugenol As Analyzed with Double Resonance and Grid-Based Autofit. *J. Phys. Chem. A* **2019**, *123*, 1091−1099.
(12) Bohn, R. K.; Montgomery, J. A.; Michels, H. H.; Fournier, J. A. Second moments and rotational spectroscopy. *J. Mol. Spectrosc.* **2016**, *325*, 42−49.
(13) *Equilibrium Molecular Structures: From Spectroscopy to Quantum Chemistry*; Demaison, J., Boggs, J. E., Császár, A. G., Eds.; CRC Press: Boca Raton, FL, 2011.
(14) Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press, 2016.
(15) Gal, Y.; Ghahramani, Z. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. *arXiv (Statistics.Machine Learning)*, October 4, 2016, 1506.02142, ver. 6. https://arxiv.org/abs/1506.02142 (accessed 2020-02-17).
(16) Peironcely, J. E.; Rojas-Chertó, M.; Fichera, D.; Reijmers, T.; Coulier, L.; Faulon, J.-L.; Hankemeier, T. OMG: Open Molecule Generator. *J. Cheminf.* **2012**, *4*, 21.
(17) O'Boyle, N. M.; Banck, M.; James, C. A.; Morley, C.; Vandermeersch, T.; Hutchison, G. R. Open Babel: An open chemical toolbox. *J. Cheminf.* **2011**, *3*, 33.
(18) Frisch, M. J.; et al. *Gaussian 16*, rev. A.01; Gaussian, Inc.: Wallingford, CT, 2016.
(19) Lee, K. L. K.; McCarthy, M. Bayesian Analysis of Theoretical Rotational Constants from Low-Cost Electronic Structure Methods. *J. Phys. Chem. A* **2020**, *124*, 898−910.
(20) Rupp, M.; Tkatchenko, A.; Müller, K.-R.; von Lilienfeld, O. A. Fast and Accurate Modeling of Molecular Atomization Energies with Machine Learning. *Phys. Rev. Lett.* **2012**, *108*, 058301.
(21) Ramakrishnan, R.; Dral, P. O.; Rupp, M.; von Lilienfeld, O. A. Quantum chemistry structures and properties of 134 kilo molecules. *Sci. Data* **2014**, *1*, 140022.
(22) Ruddigkeit, L.; van Deursen, R.; Blum, L. C.; Reymond, J.-L. Enumeration of 166 Billion Organic Small Molecules in the Chemical Universe Database GDB-17. *J. Chem. Inf. Model.* **2012**, *52*, 2864−2875.
(23) Gupta, A.; Müller, A. T.; Huisman, B. J. H.; Fuchs, J. A.; Schneider, P.; Schneider, G. Generative Recurrent Networks for De Novo Drug Design. *Mol. Inf.* **2018**, *37*, 1700111.
(24) Hirohara, M.; Saito, Y.; Koda, Y.; Sato, K.; Sakakibara, Y. Convolutional neural network based on SMILES representation of compounds for detecting chemical motif. *BMC Bioinf.* **2018**, *19*, 526.
(25) Baltruschat, M.; Kelley, B.; Swain, M.; Tosco, P. *RDKit: Open-Source Cheminformatics Software*, 2019; https://zenodo.org/record/3603542.
(26) Paszke, A. et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*; Wallach, H., Larochelle, H., Beygelzimer, A., d'AlchéBuc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc., 2019; pp 8026−8037.
(27) Loshchilov, I.; Hutter, F. Decoupled Weight Decay Regularization. *arXiv (Computer Science.Machine Learning)*, January 4, 2019, 1711.05101, ver. 3. https://arxiv.org/abs/1711.05101 (accessed 2020-02-17).
(28) Kingma, D. P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv (Computer Science.Machine Learning)*, January 30, 2017, 1412.6980, ver. 9. https://arxiv.org/abs/1412.6980 (accessed 2020-02-17).
(29) Hinton, G. E.; Srivastava, N.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. R. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv (Computer Science.Neural and Evolutionary Computing)*, July 3, 2012, 1207.0580, ver. 1. https://arxiv.org/abs/1207.0580 (accessed 2020-02-17).
(30) Li, Y.; Gal, Y. Dropout Inference in Bayesian Neural Networks with Alpha-divergences. *arXiv (Computer Science.Machine Learning)*, March 8, 2017, 1703.02914, ver. 1. https://arxiv.org/abs/1703.02914 (accessed 2020-02-17).
(31) Zhou, Z.-H.; Wu, J.; Tang, W. Ensembling neural networks: Many could be better than all. *Artificial Intelligence* **2002**, *137*, 239−263.
(32) Nair, V.; Hinton, G. E. Rectified Linear Units Improve Restricted Boltzmann Machines. In *ICML '10: Proceedings of the 27th International Conference on Machine Learning*, Haifa, Israel, 2010; Omnipress, 2010; pp 807−814.
(33) Maas, A. L.; Hannun, A. Y.; Ng, A. Y. Rectifier Nonlinearities Improve Neural Network Acoustic Models. In *ICML '13: Proceedings of the 30th International Conference on Machine Learning*, Atlanta, GA, 2013; Curran Associates, Inc., 2013; p 6.
(34) Kullback, S. *Information Theory and Statistics*, reprint ed.; Smith: Gloucester, MA, 1978; OCLC: 187308462.
(35) Müller, R.; Kornblith, S.; Hinton, G. When Does Label Smoothing Help? *arXiv (Computer Science.Machine Learning)*, December 5, 2019, 1906.02629, ver. 2. https://arxiv.org/abs/1906.02629 (accessed 2020-02-17).
(36) Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. *arXiv (Computer Science.Computer Vision and Pattern Recognition)*, Decem-

ber 11, 2015, 1512.00567, ver. 3. https://arxiv.org/abs/1512.00567 (accessed 2020-02-17).
(37) He, K.; Zhang, X.; Ren, S.; Sun, J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. *arXiv (Computer Science.Computer Vision and Pattern Recognition)*, February 6, 2015, 1502.01852, ver. 1. https://arxiv.org/abs/1502.01852 (accessed 2020-02-17).
(38) Manning, C. D.; Raghavan, P.; Schütze, H. *Introduction to Information Retrieval*; Cambridge University Press: New York, 2008; OCLC: ocn190786122.
(39) Gal, Y.; Ghahramani, Z. A Theoretically Grounded Application of Dropout in Recurrent Neural Networks. *arXiv (Statistics.Machine Learning)*, October 5, 2016, 1512.05287, ver. 5. https://arxiv.org/abs/1512.05287 (accessed 2020-02-17).
(40) Pedregosa, F.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825−2830.
(41) O'Boyle, N. M. Towards a Universal SMILES representation - A standard method to generate canonical SMILES based on the InChI. *J. Cheminf.* **2012**, *4*, 22.
(42) Heller, S. R.; McNaught, A.; Pletnev, I.; Stein, S.; Tchekhovskoi, D. InChI, the IUPAC International Chemical Identifier. *J. Cheminf.* **2015**, *7*, 23.
(43) Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Computation* **1997**, *9*, 1735−1780.
(44) Cho, K.; van Merrienboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv (Computer Science.Computation and Language)*, September 3, 2014, 1406.1078, ver. 3. https://arxiv.org/abs/1406.1078 (accessed 2020-02-17).
(45) Arús-Pous, J.; Johansson, S. V.; Prykhodko, O.; Bjerrum, E. J.; Tyrchan, C.; Reymond, J.-L.; Chen, H.; Engkvist, O. Randomized SMILES strings improve the quality of molecular generative models. *J. Cheminf.* **2019**, *11*, 71.
(46) Wolpert, D.; Macready, W. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation* **1997**, *1*, 67−82.
(47) Wu, Z.; Ramsundar, B.; Feinberg, E. N.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: a benchmark for molecular machine learning. *Chemical Science* **2018**, *9*, 513−530.
(48) Huang, B.; von Lilienfeld, O. A. Communication: Understanding molecular representations in machine learning: The role of uniqueness and target similarity. *J. Chem. Phys.* **2016**, *145*, 161102.
(49) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In *NIPS '17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, CA, 2017; Curran Associates, Inc., 2017; p 11.
(50) Baldi, P.; Benz, R. W.; Hirschberg, D. S.; Swamidass, S. J. Lossless Compression of Chemical Fingerprints Using Integer Entropy Codes Improves Storage and Retrieval. *J. Chem. Inf. Model.* **2007**, *47*, 2098−2109.
(51) Winter, R.; Montanari, F.; Noé, F.; Clevert, D.-A. Learning continuous and data-driven molecular descriptors by translating equivalent chemical representations. *Chemical Science* **2019**, *10*, 1692−1701.
(52) Kuleshov, V.; Fenner, N.; Ermon, S. Accurate Uncertainties for Deep Learning Using Calibrated Regression. *arXiv (Computer Science.Machine Learning)*, July 1, 2018, 1807.00263, ver. 1. https://arxiv.org/abs/1807.00263 (accessed 2020-02-17).
(53) Hernández-Lobato, J. M.; Adams, R. P. Probabilistic Backpropagation for Scalable Learning of Bayesian Neural Networks. In *ICML '15: Proceedings of the 32nd International Conference on Machine Learning*, Lille, France, 2015; Curran Associates, Inc., 2015; pp 1861−1869.
(54) Blundell, C.; Cornebise, J.; Kavukcuoglu, K.; Wierstra, D. Weight Uncertainty in Neural Networks. *arXiv (Statistics.Machine Learning)*, May 21, 2015, 1505.05424, ver. 2. https://arxiv.org/abs/1505.05424 (accessed 2020-02-17).
(55) Rohekar, R. Y.; Gurwicz, Y.; Nisimov, S.; Koren, G.; Novik, G. Bayesian Structure Learning by Recursive Bootstrap. In *NIPS '18: Proceedings of the 32nd International Conference on Neural Information Processing Systems*, Montreal, Canada, 2018; Curran Associates, Inc., 2018; pp 10546−10556.
(56) Hinton, G. E.; van Camp, D. Keeping the neural networks simple by minimizing the description length of the weights. In *Proceedings of the Sixth Annual Conference on Computational Learning Theory*, Santa Cruz, CA, 1993; Association for Computing Machinery, 1993; pp 5−13.
(57) Graves, A. Practical Variational Inference for Neural Networks. In *Advances in Neural Information Processing Systems 24*; Shawe-Taylor, J., Zemel, R. S., Bartlett, P. L., Pereira, F., Weinberger, K. Q., Eds.; Curran Associates, Inc., 2011; pp 2348−2356.

https://dx.doi.org/10.1021/acs.jpca.0c01376
J. Phys. Chem. A 2020, 124, 3002−3017