Identification of Molecular Substances from NMR Spectroscopy

Riley Carlson and **Ishaan Singh**
CS 229 Winter 2024
Stanford University
{rileydc, ishaanks}@stanford.edu

1 Introduction

In drug discovery, identifying products of chemical reactions is often a complex and time-consuming task that slows down research and consumes resources. One common technique for identification, NMR spectroscopy, aligns magnetic spins in a compound with a constant magnetic field, perturbs this alignment with an oscillating field, and analyzes phase coherence loss in nuclei to detect unique signals from different functional groups, producing a molecular 'fingerprint'. However, complex chemical structures with multiple functional groups can complicate interpreting these signals, challenging the assignment of NMR data to specific structures (Jones and Fleming, 2014 - 2014). Below is a sample NMR spectra for ethanol, which is a compound that has two functional groups, an alcohol and an alkane:

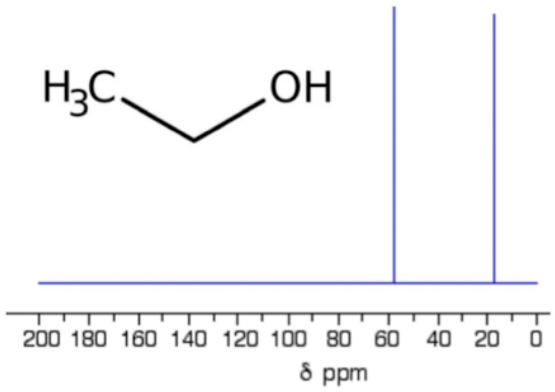

Figure 1: ^{13}C NMR of ethanol

As this example demonstrates, there are two major peaks: the leftmost one for the alcohol group and the rightmost one for the alkane group (Jones and Fleming, 2014 - 2014).

We aim to expedite chemical research by developing a model to classify chemical structures using NMR spectroscopy data. This model will enable rapid identification of compounds and comprehension of their environmental interactions, thereby streamlining drug design.

In this research, we will take an unknown molecule's spectral data and predict the functional groups present in the molecule. Specifically, the input to our algorithm is 5 features about the reaction environment and 20 features detailing numerical values in ppm (parts per million) about the chemical shifts. We then use multilabel logistic regression, a fully connected neural network, a mixed fully connected neural network and RNN-LSTM, and XGBoost to output the predicted binary classes for each of the 12 following labels:

Alkane	Alkene	Aldehyde/Ketone
Alcohol	Amide	Amine
Ether	Carboxyllilic Acid	Alkyne
Ester	Aromatic Ring	Nitrile

Table 1: The 12 functional groups molecules will be labeled with

2 Related Work

Historically, the task of assigning functional groups to chemical shifts has been done by hand, usually at the expense of unsuspecting students on an organic chemistry exam (Jones and Fleming, 2014 - 2014). The application of machine learning is a relatively new task, and earlier works seem to use differing versions of neural networks. Indeed, Ala-Korpela et al. (1996) used a neural network to quantify the percentages of lipoproteins and lipids in human plasma from ^{1}H NMR. Similarly, Basu et al. (2003) trained a neural network on ^{1}H NMR chemical shifts to predict the cetane number

for fuels, a metric depicting fuel quality. Gan et al. (2019) used a fully connected neural network with Adam optimization and Hamming Loss as the major evaluation metric, as they focused on classifying spectroscopic samples that contained mixtures of gases, quantifying the relative amounts of each gas. They found that the fully connected neural network was able to capture the complexities in the dependencies that each feature had on the overall chemical shifts observed, which is a key observation that is implicit in our work and supported by major chemistry publishers (Jones and Fleming, 2014 - 2014).

In recent years, more successful approaches have gravitated towards using regression and boosting as the main techniques in classification. For example, Abdul Jameel et al. (2016) worked on a similar cetane number classification problem to Basu et al. (2003) but used multiple linear regression for the prediction, by predicting the branching index of the given molecule from the ^{1}H NMR and then predicting the cetane number based on this information. Martinez-Trevino et al. (2020) took in NMR spectra and attempted to classify into 9 exclusive classes of natural products (much more specific than functional groups). They found that XGBoost achieved the lowest loss on the data, giving us inspiration to use this model in our predictions.

In our project, we aim to draw on previous work by integrating these methods, using regression analysis, neural networks, and XGBoost as learning paradigms. However, the scope of our project is far broader: rather than considering a specific class of compound (gas, natural products, fuel, etc.) we generalize to any molecule's NMR and predict what functional groups it possesses.

3 Methods

We will detail the dataset used along with preprocessing tasks required to use the data for training.

3.1 Data

Our data was found from an online NMR database (Spectral Database for Organic Compounds, SDBS) by AIST, accessible at this link: `https://sdbs.db.aist.go.jp/sdbs/cgi-bin/cre_index.cgi`. There are around 45,000 spectra in this dataset, most of which come with the names, the solvents, the type of nuclei observed, the current applied, the temperature used during the experiment, and the values in ppm of the peaks associated with the given compound. Each functional group in a molecule will produce a peak based on the magnetic resonance of the bonds, and the values at which these peaks occurred (known as the chemical shifts) were the output for each of these molecules. However, the labels of the molecule from Table 1 were missing from the data, resulting in the need to label the molecules based on a lookup from the molecule's name, to an NIH database that produced an iterable "SMILES" string format of the molecule. After labelling, the following distribution was obtained:

Distribution of Functional Groups

Compound Type	Percentage (%)
Alkane	86.57
Alkene	24.18
Alkyne	3.84
Ketone/Aldehyde	22.69
Ester	20.90
Amide	10.52
Amine	40.44
Aromatic Ring	67.40
Nitrile	5.39
Alcohol	17.96
Carboxylic Acid	5.46
Ether	31.31

Table 2: Percentage of Different Compound Types

Alkanes may seem overrepresented in our dataset, but this is due to the prevalence of carbon-carbon single bonds in most molecules.

3.2 Preprocessing

Data extraction from the XML file involved using regular expressions to scrape molecule names, NMR types, solvents, field strength, temperature, and chemical shift x-coordinates. Unnamed or irrelevant entries were discarded. We handled Greek letters by romanization for compatibility with the cheminformatics toolkit `RDKit`, which converted molecule names to SMILES format. This conversion allowed atom-by-atom iteration to identify functional groups, with assignment to the largest group given when overlaps occurred (e.g., labeling a group of atoms as a carboxylic acid instead of an alcohol and a carbonyl). Figure 2 illustrates this

convention.

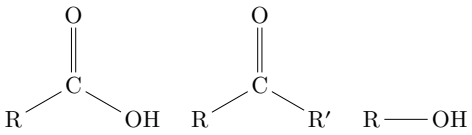

Figure 2: From left to right: a generic carboxylic acid, a generic ketone, and a generic alcohol.

To manage the dataset's complexity, we capped the number of peaks per molecule at 20, filling in missing values with "NA." After preprocessing, we retained data for 30,492 molecules, each labeled with up to 12 possible functional groups, marked as '1' for presence and '0' for absence. The data preprocessing and labeling pipeline was executed using `RDKit`, with subsequent machine learning tasks performed in PyTorch, NumPy, and SciKit. We used an 80/20 random split for training and testing.

3.3 Evaluation

To evaluate our models, we employed Hamming Loss, which is given by

$$\mathcal{L}_{\mathrm{H}}(y, \hat{y}) = \frac{1}{N \cdot L} \sum_{i=1}^{N} \sum_{j=1}^{L} \mathrm{Hamming}(y_i, \hat{y}_i)$$

where the Hamming function is an indicator function that takes as input y_i(the actual label) and $\hat{y}_i$ (the predicted label) and returns a 1 if $y_i \neq \hat{y}_i$ and a 0 otherwise (Wu and Zhou, 2017). Here, N is the number of testing examples, and L is the number of labels per example (which here is 12). A Hamming Loss of 0.5 implies that, on average, half of the labels are misclassified per compound, indicating a less accurate model. This loss is a standard metric for multi-labeling problems, considering that a molecule can have multiple labels simultaneously. Exact accuracy, which measures the percentage of predictions aligning perfectly with the ground truth, is misleading in multilabel classification. It fails to depict error distribution across misclassified classes, hence why we prefer the Hamming Loss metric.

In terms of qualitative assessment, we will consider the confusion matrices for each of the models and analyze the patterns in the matrices to identify common errors.

4 Models

In this section, we summarize the four models implemented including model specifications.

4.1 Baseline: Multi Label Logistic Regression

As our baseline, we implemented multilabel logistic regression to see how well a simple model could distinguish between the classes. After splitting, we used full-batch gradient descent over 170 epochs. Furthermore for optimization, we used Adam (Duchi et al., 2011). As for our loss function, we used binary cross entropy loss (the standard loss function for problems of this nature), which is given by

$$\mathcal{L}_{\mathrm{BCE}}(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^{N} \left(y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \right)$$

where $\hat{y}_i$ is the predicted probability for a given label, y is the actual value (either 0 or 1), and N is the number of labels (in our case 12) (Zhang and Zhou, 2014).

4.2 Fully Connected Neural Network

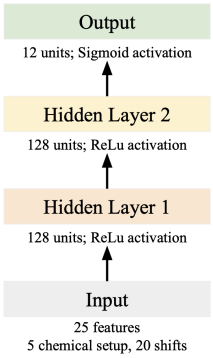

Figure 3: Architecture for fully connected NN

Neural networks generally are known for detecting complex relationships between given features, reducing bias in so doing. Because our dataset contains features such as temperature, solvent type, and field strength applied that directly impact the position of the signals, implementing a simple, fully connected neural network is useful in capturing the complex relationships inherent to this problem, similar to the approach taken by Gan et al. (2019). To do so, we implemented a fully connected neural network with the architecture described in Figure 3. The loss function used was again binary cross entropy detailed in Section 4.1.

We utilized the limited memory BFGS algorithm, a quasi-Newtonian method for approximating the inverse Hessian at each step, to update the weight and bias terms at each epoch (Liu and Nocedal, 1989). This algorithm is useful for full batch gradient descent where direct calculation of the inverse Hessian is computationally intractible.

4.3 Mixed RNN-LSTM with FCNN

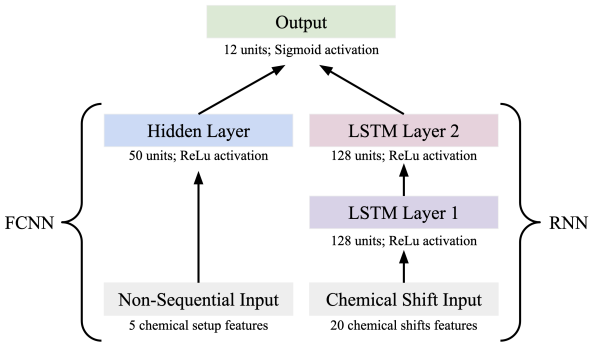

Figure 4: Architecture for Mixed RNN-LSTM with FCNN

Given the above discussion on why a neural network is useful in a setting such as this one, we believed that using a model that treated the chemical shifts as sequential data would be useful in classification. To do so, we split the data by the non-sequential features (temperature, field strength, etc.) and the sequential features (the chemical shifts). The mixed neural network architecture is specified in Figure 4.

Under this architecture, full batch gradient descent was run using Adam optimization, with a learning rate of .001.

4.4 XGBoost

The last model we implemented was XGBoost because of its reputation of performing well in classification (Chen and Guestrin, 2016). For our implementation, we use the wrapper `MultiOutputClassifier()` to use XGBoost in this multilabel classification setting. Further, we use binary cross entropy, detailed in Section 4.1, as the loss function.

4.5 Results

We report the Hamming Loss for each of the models in Table 3

For brevity, we will present the confusion matrices for all 12 classes of two of the models, namely that of the XGBoost and the mixed NN model in Figures 6 and 5, respectively, to illustrate the quality of prediction from the best and worst performing models based on the Hamming loss in Table 3.

Model	Test Hamming Loss
Logistic Regression	0.168
Mixed NN	0.190
Fully Connected NN	0.140
XGBoost	0.118

Table 3: Hamming Loss results on test data for all models trained

5 Discussion

Based solely on the Hamming Loss comparison for the test data outlined in Table 3, the XGBoost model emerges as the most effective. This outcome can be attributed to the model's architecture, which prioritizes the frequently misclassified classes, such as amines, ethers, and carboxylic acids. In contrast, other model architectures treat each data point equally, neglecting to emphasize training on the 'weaker' performing areas.

Under our Logistic Regression baseline, our Hamming Loss was 0.168 on the testing dataset, which indicates that, on average, 16.8 labels per 100 labels are incorrect, or in our case that 9.98 out of the 12 labels for a given molecule are correct. While this result is promising and suggests that the baseline has relatively low error, it also indicates that there certainly is room for improvement.

Interestingly, the baseline outperformed the Mixed NN. This is a surprising result due to some of the data (namely the chemical shifts) being sequential data and the RNN architecture being apt for sequential data. However, the poor performance of this model can be explained by the fact that chemical shifts of the NMR spectra vary greatly by solvent type and by the nuclei observed (^{1}H vs ^{13}C). In the mixed model, the chemical shifts of the NMR are passed into the RNN without the additional context provided in a classical neural network. Fitting the RNN using only the spectral data is insufficient as the model will struggle to converge due to misunderstanding the chemical shifts and the solvent effects, which are

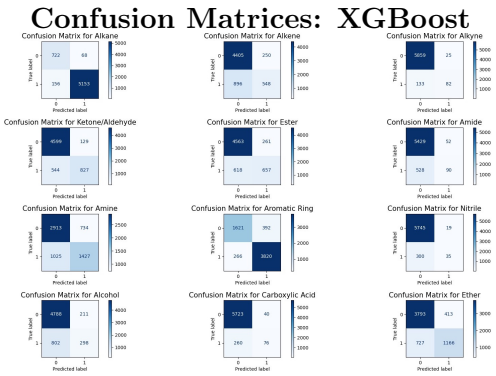

Figure 5: Confusion matrices of 12 functional groups for XGBoost

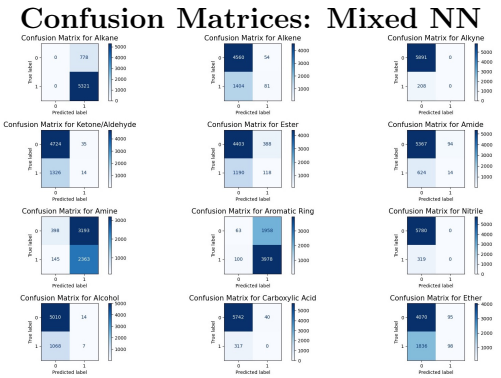

Figure 6: Confusion matrices of 12 functional groups for Mixed NN

both crucial in identifying the type of spectra observed–even though the mixed model combines these after learning the neural network, this proves to be insufficient.

Looking at the confusion matrices in Figures 6 and 5, we can see where the best model (XGBoost) outperforms the worst model (mixed NN). Specifically, the mixed NN performed poorly on many of the classes but especially with amines, aromatic rings, and ether. On the other hand, the XGBoost performs better in the aromatic rings and ether groups as seen with the brighter diagonal entries of the matrices. However, the XGBoost also struggles with amines although not to the extent to which the mixed NN does.

To address class performance imbalance, we attempted dataset rebalancing. However, rebalancing multilabel datasets, especially using techniques like SMOTE (Chawla et al., 2002) and bootstrap aggregation, is nontrivial due to the observed co-occurrences between frequently and infrequently occurring labels. Indeed, our rebalancing approach led to deteriorated model performance across all labels, as evidenced by increased hamming loss. These results imply that class imbalance is not the sole driver of negative performance.

Rather, the answer may lie in the typical chemical shifts produced by these groups (Jones and Fleming, 2014 - 2014). Amines and ethers, major sources of the loss, have significant overlap with alkanes and alkenes in the values of their typical chemical shifts. While finer variation exists in the shape of the peaks produced, this is not captured by the features passed into our model. Similarly, carboxylic acids (another major source of error) have major regions of overlap in ^{13}C spectra with other major carbonyl-containing compounds (aldehydes/ketones, esters, and amides) (Jones and Fleming, 2014 - 2014). Therefore, to increase performance of these models, some form of data that captures the variance inherent in the shape of the chemical shifts is required–no resampling technique can achieve this.

6 Conclusion

In this project, we aimed to predict the functional groups of an unknown molecule using NMR spectra. To do so, we implemented four machine learning architectures: logistic regression, fully connected neural network, mixed RNN-LSTM with fully connected neural network, and XGBoost with XGBoost performing the best in accordance with the Hamming Loss. The performance of XGBoost gives a unique, robust, and facile approach to analyzing NMR spectra, while also providing more interpretable results compared to neural nets.

Future work would expand the number of possible labels (there are far more than 12 functional groups) and include additional features to capture the finer variation in the types of peaks observed, improving accuracy as well. Such work would allow for prediction of molecules' SMILES string formats from NMR data, which is a far more difficult task but one that could revolutionize the field.

7 Acknowledgements

The authors would like to thank Audrey Kim ('25) for her collaboration in background knowledge in NMR data.

8 Contributions

Riley worked on creating the regular expressions, while Ishaan implemented code to label every molecule. They worked together on the logistic regression baseline, the fully connected neural network, the mixed neural network, and the XGBoost model, using Google Colab and bouncing ideas off of each other. Riley worked on the diagrams in the final report, while Ishaan worked on the related works. They wrote the text of the report together.

References

Abdul Gani Abdul Jameel, Nimal Naser, Abdul-Hamid Emwas, Stephen Dooley, and S Mani Sarathy. 2016. Predicting fuel ignition quality using 1h nmr spectroscopy and multiple linear regression. *Energy & Fuels*, 30(11):9819–9835.

Mika Ala-Korpela, Y Hiltunen, and Jimmy D Bell. 1996. Artificial neural network analysis of 1h nmr spectroscopic data from human plasma. *Anticancer research*, 16(3B):1473–1478.

B Basu, GS Kapur, AS Sarpal, and R Meusinger. 2003. A neural network approach to the prediction of cetane number of diesel fuels using nuclear magnetic resonance (nmr) spectroscopy. *Energy & fuels*, 17(6):1570–1575.

Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7).

Luyun Gan, Brosnan Yuen, and Tao Lu. 2019. Multi-label classification with optimal thresholding for multi-composition spectroscopic analysis. *Machine Learning and Knowledge Extraction*, 1(4):1084–1099.

Maitland Jones and Steven A. (Steven Alan) Fleming. 2014 - 2014. *Organic chemistry / Maitland Jones, Jr., New York University, Steven A. Fleming, Temple University.*, fifth edition. edition. W.W. Norton Company, New York.

Dong C Liu and Jorge Nocedal. 1989. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1):503–528.

Saul H Martinez-Trevino, Víctor Uc-Cetina, María A Fernández-Herrera, and Gabriel Merino. 2020. Prediction of natural product classes using machine learning and 13c nmr spectroscopic data. *Journal of Chemical Information and Modeling*, 60(7):3376–3386.

Xi-Zhu Wu and Zhi-Hua Zhou. 2017. A unified view of multi-label performance measures. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3780–3788. PMLR.

Min-Ling Zhang and Zhi-Hua Zhou. 2014. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1819–1837.