A FURTHER DESCRIPTIVE ANALYSES OF EACH DATASET

In this section, we provide a detailed data analysis of GlycoNMR, focusing on both the quantity and variety of monosaccharides within our dataset.

A.1 HISTOGRAM DISTRIBUTION OF CARBOHYDRATE LENGTHS IN BOTH DATASETS

We further analyze the data volume of GlycoNMR. We plot the distributions of the number of monosaccharides that every carbohydrate contains in both GlycoNMR.Exp and GlycoNMR.Sim. In Figure 5, we use 'length of glycan' to denote the number of monosaccharides that the carbohydrate contains. We observe: both histograms exhibit a right-skewed distribution in the length of the glycan. This indicates that GlycoNMR.Exp contains a greater proportion of small and middle-sized carbohydrates than large-sized carbohydrates. Therefore, existing MRL methods may be biased towards smaller carbohydrates.

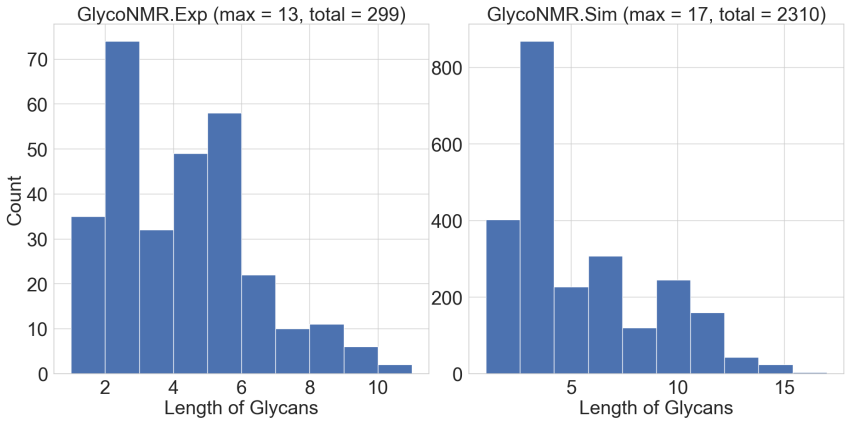

Figure 5: Distribution of glycan length in both datasets. The horizontal axis indicates the number of monosaccharides in the carbohydrate, the vertical axis indicates the corresponding number of carbohydrates presented in the dataset.

A.2 PERCENTAGE OF MONOSACCHARIDE TYPES IN BOTH DATASETS

We investigate the diversity of monosaccharide types in GlycoNMR. For each dataset, we count the occurrence of all monosaccharides and present the percentage of the top eight most frequently appearing monosaccharides in Table 5. The entry "Others" represents the category of relatively infrequently appeared monosaccharides, including stem type: ManA, Neu, GalN, Ara, etc. We demonstrate that GlycoNMR covers the most commonly occurring stems of monosaccharides as introduced in (Chaplin & Kennedy, 1986) for example.

Table 5: Percentage of the most common monosaccharide unit types in the two datasets

GlycoNMR.Sim		GlycoNMR.Exp	
Monosaccharide	Percentage	Monosaccharide	Percentage
Glc	18.86%	Gal	19.73%
Gal	17.5%	Glc	17.7%
GlcNAc	12.18%	GlcNAc	12.21%
Fuc	12.1%	Rha	11.06%
Xyl	8.51%	Man	6.81%
Man	6.23%	Fuc	4.87%
GlcA	6.19%	Kdo	4.78%
GalA	5.49%	GlcA	4.42%
Others	12.94%	Others	18.42%

B Details on Features Tables

In this section, we present a comprehensive description of the processed PDB file, including the curated features mentioned in Section 2 and Section 3.2. For each feature, we provide its data type along with a detailed explanation. Lines 1-8 in Table 6 record attributes presented in the original PDB file. We incorporate the Atom_name and Atom_type as components of the node features. Coordinate x, y, and z is used as spatial information to construct the MRL models. Lines 9-15 record the processed node features as introduced Table 2. Lines 15-25 describe the feature: Modifications, that are used in GlycoNMR.Sim. On curating the feature Modification, we first identify the modification group using Lineage, Atom_num, Residue_name, and atom connectivity. Then, we calculate each atom's distance(atom path) to the identified modification group, set up several distance thresholds to convert them into categorical values and incorporate them as node features. Notice that the atom connectivity information is generally missing in GlycoNMR.Exp, thus it can be ambiguous to match the atoms to their corresponding modification groups, and we omitted this feature for now in the smaller Glycosciences.DB-sourced dataset only (in contrast, Modification was included in the GODESS-sourced dataset). Future databases of new experimental results in carbohydrate NMR spectra should seek to improve the clarity in this area, such as with more uniform standards in data annotation by the original uploaders.

Last, we use the labeled in-ring atoms' NMR shift as ground truth values.

Table 6: Detailed feature description

Value	Datatype	Descriptions
Atom_num	Numerical	Atom index number in the carbohydrate
Atom_name	Categorical	Atom name that also indicates its within-monosaccharide position index
Residual_name	Categorical	Three letters abbreviation of monosaccharide name
Residual_num	Numerical	Monosaccharide order number assigned
x	Numerical	X coordinate of the atom
y	Numerical	Y coordinate of the atom
z	Numerical	Z coordinate of the atom
Atom_type	Categorical	Chemical element type of the atom
Residual_accurate_name	Categorical	Full name of monosaccharide or modification group that atom belongs to
Lineage	String	Lineage (linkage) information of the current residue
Ac_component	Categorical	Whether atom is in an Ac modification
bound_AB	Categorical	Anomeric orientation of hydroxyl group
fischer_projection_DL	Categorical	Fischer convention
reformulated_standard_mono	Categorical	Monosaccharide stem name
carbon_number_PF	Categorical	Number of ring carbons (ring size)
Me_min_atom_distance	Numerical	Distance of the shortest atom path to Me modification group
Me_min_atom_path	Categorical list	The shortest atom path to Me modification
Ser_atom_distance	Numerical	Distance of the shortest atom path to Ser modification group
Ser_atom_path	Categorical list	The shortest atom path to Ser modification
Ac_min_atom_distance	Numerical	Distance of the shortest atom path to Ac modification group
Ac_min_atom_path	Categorical list	The shortest atom path to Ac modification
S_min_atom_distance	Numerical	Distance of the shortest atom path to S-related modification group
S_min_atom_path	Categorical list	The shortest atom path to S-related modification
Gc_min_atom_distance	Numerical	Distance of the shortest atom path to Gc modification group
Gc_min_atom_path	Categorical list	The shortest atom path to Gc modification
main_ring_shift	Numerical	Chemical shift values of all labeled main ring atoms
shift	Numerical	Chemical shift values of all labeled atoms

C Possible Future Research Topics

In this section, we provide several unexplored glycoscience-related related research topics that `GlycoNMR` can be used for. We believe these topics can potentially benefit the overall ML and glycoscience community.

Customized models for carbohydrate data: Models specifically designed to accommodate the unique characteristics and structure of the carbohydrate data are important to develop. As introduced in Section 2, carbohydrates are a special type of biomolecule that is formed via the condensation reactions of monosaccharides. We conduct heavy feature engineering to extract the monosaccharide-related features, and our experimental results in Table 3 have already demonstrated the usefulness of monosaccharide information (stem type) in NMR shift prediction. However, we incorporate them as atom-level features in our baseline and the 3D-based MRL models. In this case, the existing models

may fail to capture the spatial information between monosaccharides, and more neural network layers corresponding to the structural hierarchies inherent to carbohydrates could improve prediction quality in future work. On the other hand, a carbohydrate's unique atoms-to-monosaccharides-to-carbohydrate characteristic inherently satisfies a hierarchical graph structure so the information is partly captured in the current implementation. We believe that developing a customized MRL model (e.g. learning representations for both atoms and monosaccharides) can help learn a better node representation for accurate NMR shift predictions in future work.

Predicting NMR spectra: As presented in Section 3.1, extensive data annotation is required for preparing the atom-level carbohydrate NMR chemical shift data. Notably, for annotating each carbohydrate, the key step is to match the monosaccharides present in the PDB (Protein Data Bank) structure file to the monosaccharides present in the NMR (Nuclear Magnetic Resonance) chemical shift file. This step not only demands significant effort but also necessitates domain expertise, but will continue to do so at least until the experimental glycoscience field adopts more uniform standards in data files.

In the field of glycosciences, the ideal scenario is to predict the full continuous spectrum (peak widths and noise included) depicted in Figure 2 (b) and (c) directly from the carbohydrate structure. In our case, the NMR chemical shift prediction problem of just peaks is reformulated as graph-regression tasks with promising initial performance. The biggest improvements in this direction will necessitate both increasingly larger and more diverse experimental datasets, as well as model innovations.

D Model Setup and Computation Resources

To ensure a fair comparison, the hidden embedding size for all 3D GNN models is set to 128, and the number of hidden layers is set to 4 in the GlycoNMR.Sim dataset. In the GlycoNMR.Exp dataset, due to the limitations in data size and to prevent over-fitting, the number of hidden layers is set to 2. It takes around 5-34 seconds to train a single epoch with a batch size of 4, depending on different models. All data processing and model training is performed on a Linux workstation with an Intel Core i7 CPU, 32GB memory, and two GeForce RTX 3090 GPUs. Our entire training time for all models in aggregate was on the scale of several hours. Loading codes for the dataset will also be provided in the linked anonymous GitHub after the completion of the peer review. We also provided more detailed run-time information and epoch numbers in the anonymous Github repository.

E RMSE formula for Benchmarks

The RMSE was calculated according to the usual equation in all results presented throughout the manuscript:

$$RMSE = \sqrt{\sum_{i=1}^{N} \frac{(y_i - \hat{y}_i)^2}{N}}$$

Where y_i is the recorded NMR chemical shift, $\hat{y}_i$ is the prediction from our GNN model on the i^{th} atom from the test set, and N is the number of the test data points.

F Random Forest Baseline

Table 7: NMR chemical shift prediction benchmark using a random forest model (in RMSE). The code is provided on the anonymous Github repository.

	GlycoNMR.Sim		**GlycoNMR.Exp**	
	^{13}C	^{1}H	^{13}C	^{1}H
Random Forest	2.446	0.132	4.117	0.178

We conducted a traditional ML baseline experiment using random forest to predict atomic NMR shifts. The features of each atom (represented as a node in its carbohydrate graph) follow the same

initializing method as used for training the 2D GNN model. In addition, we follow the same splitting method as we did in Section 3.4. In general, the baseline model slightly underperforms relative to the 2D GNN model. This demonstrates the effectivenss of our feature engineering step in Section 3.2.

G BENCHMARK FOR MULTI-TASK NMR SHIFT PREDICTION

We trained 3D GNN models to perform multi-task learning on both GlycoNMR.Sim and GlycoNMR.Exp. Each 3D-based model is trained to predict the carbon NMR shift and the hydrogen shift jointly. The results are summarized in Table 8. We notice that there is an overall significant drop in performance across all 3D GNN models.

Table 8: NMR chemical shift prediction benchmark using 3D MRL methods (in RMSE).

	GlycoNMR.Sim		**GlycoNMR.Exp**	
	^{13}C	^{1}H	^{13}C	^{1}H
ComENet (Wang et al., 2022)	1.987	**0.157**	**3.006**	0.411
DimeNet++ (Gasteiger et al., 2020a)	1.954	0.199	3.696	**0.185**
SchNet (Schütt et al., 2017)	**1.523**	0.590	3.187	0.946
SphereNet (Liu et al., 2022)	2.258	0.169	3.364	0.638

H RUNNING TIME COMPARISON

Table 9: Running time(s) comparisons for 3D GNNs

Dataset	**ComeNet**	**DimeNet++**	**SchNet**	**SphereNet**
GlycoNMR.Sim	7.564	20.581	3.615	31.831
GlycoNMR.Exp	1.257	2.312	0.754	2.032

Running time comparison of 3D GNN models, the duration in seconds for each training epoch is reported. For a fair comparison across the 3D-based GNN models, in GlycoNMR.Sim dataset, we set the batch size to 4, the number of hidden channels to 128, and the number of layers to 4, in GlycoNMR.Exp, we set the batch size to 2, the number of hidden channels to 64, and the number of layers to 2.

I HYPERPARAMETER SELECTION ON GLYCONMR.EXP

We fine-tune the 3D-based GNN models on GlycoNMR.Exp to prevent overfitting, The hyperparameter is selected from the following ranges: learning rate [0.001, 0.01], batch size: [2, 4, 8], number of layers: [2, 3, 4], hidden channel size: [32, 64, 128, 256], and the cut-off distance for deciding the interactions between atoms: [4.0, 5.0]. We unfortunately did not have time to do more substantial hyperparameter tuning. We believe users of our dataset will be in better positions to provide better results than us with the innovative design of the 3D-based MRL model and substantial hyperparameter selection.

J DATA ANNOTATION SUPPLEMENTS

In this section, we provide two supplemental repositories to help illustrate our data preprocessing pipeline. One of our major contributions is to extensively curate the raw files from the Glycosciences extensively.DB and GODESS dataset and make the GlycoNMR dataset friendly to machine learning researchers. To achieve this, we have made significant efforts in data preprocessing.

We summarize the data preprocessing pipelines on Glycoscience.DB in the following five steps. 1, We manually check the carbohydrate data scrapped from Glycosciences.DB, and we filtered

those carbohydrates with complete or nearly complete NMR shifts. 2, We reformulate all the PDB files(as well as the label files) into an interpretable and consistent format, as they are uploaded from various labs. 3, We examined the carbohydrates with branched monosaccharide chains, and manually matched the monosaccharide IDs from the PDB file and the label file. 4, We trained a simple 2D GNN model and got the NMR chemical shifts for each annotated atom. 5, We examine those carbohydrates with significant high errors and apply an outlier check. If the error comes from the mismatches in monosaccharide IDs in Step 4, we then go back to the previous steps 2, 3 and 4. The data preprocessing pipeline in GODESS is relatively similar to the Glycoscience.DB. We construct a semi-automatic pipeline to annotate the GODESS dataset since the dataset is generated from the simulated software with consistent formatting. We introduce this pipeline in our released repository provided below.

To further demonstrate our efforts, we released two repositories for reference on data cleaning, processing, and annotating:

Creating GlycoNMR.Sim from the GODESS (https://anonymous.4open.science/r/GODESS_preprocess-F9CD/README.md)

Creating GlycoNMR.Exp from the Glycosciences.DB (https://anonymous.4open.science/r/GlycoscienceDB_preprocess-B678/README.md).

The data preprocessing steps are introduced in detail in the README.md file.

K EXAMPLE CODES AND DEMOS

We provide four Jupyter Notebook demos in the anonymous GitHub repo for detailed instructions. They introduce step by step on how to utilize the GlycoNMR.Sim and GlycoNMR.Exp datasets to train a 3D or 2D GNN model.

Train a 2D-based GNN model on GlycoNMR.Sim: https://anonymous.4open.science/r/GlycoNMR-D381/2D_example_Sim_GlycoNMR.ipynb.

Train a 2D-based GNN model on GlycoNMR.Exp: https://anonymous.4open.science/r/GlycoNMR-D381/2D_example_Exp_GlycoNMR.ipynb.

Train a 3D-based GNN model on GlycoNMR.Sim: https://anonymous.4open.science/r/GlycoNMR-D381/3D_example_Exp_GlycoNMR.ipynb.

Train a 3D-based GNN model on GlycoNMR.Exp: https://anonymous.4open.science/r/GlycoNMR-D381/3D_example_Sim_GlycoNMR.ipynb.