Glyde: A Domain-Aware, Topology-Biased Glycan Language Model for Viral Receptor Binding

Ganesh Talluri[1]
[1]BASIS Peoria

Abstract

Viruses recognize host cells by attaching to sugar molecules called glycans on the cell surface. These interactions depend on specific chemical features such as terminal structures and how sugars link together, rather than the entire glycan shape. Understanding these recognition patterns is of key importance for predicting which viruses might jump to humans and for personalized medicine treatments. Yet, most glycan machine learning models still treat these molecules as generic sequences expecting them to pick up on biological structures through pattern recognition across data. So, I built GLYDE, a language model that integrates these biological rules directly into its architecture. Instead of learning patterns purely from data, GLYDE prioritizes the glycan features that matter most for viral recognition using mathematical representations of terminal residues, branching patterns, and linkage chemistry. Using glycan binding data from GlyGen, I designed a lightweight neural network (380k parameters) that processes glycans as topology-aware structures rather than just flat sequences. To validate my model, I tested GLYDE on viral binding prediction tasks. It achieved 0.96 AUROC in distinguishing binding from non-binding glycans and showed strong ranking performance, significantly outperforming random selection. These metrics clearly show that researchers can accurately identify which glycans viruses will target, and that is very important for finding pandemic threats. My work proves that incorporating prior knowledge into language models improves both accuracy and generalization. Beyond the results, GLYDE helps us study how certain viruses recognize cells, with potential applications in vaccine development, antiviral design, and pandemic risk assessment.

Code and implementation available at: `https://github.com/g4nesh/glyde`

1 Introduction

Glycans are of key importance to host–pathogen recognition. Unlike proteins or nucleic acids, glycans are branched, chemically heterogeneous, and highly sensitive to linkage configuration. Viral attachment often depends less on the entire glycan structure than on exposed terminal motifs and the chemistry of the linkages that present them. Influenza receptor recognition is a standard example: preference for α2-3 versus α2-6 sialylated termini can substantially alter host specificity and tissue tropism (Varki, 2017; Coff et al., 2020). This creates a representation problem. A model that treats glycans as flat strings may blur the key difference between core and terminal residues, while a purely data-driven graph encoder may need a lot of data before it rediscovers well-known structural priors.

Recent glycobiology models have shown that learned representations are useful for motif discovery and glycan interaction prediction. Motif-mining methods identify recurrent substructures associated with binding (Coff et al., 2020). Graph neural networks learn glycan embeddings from connectivity

Preprint. Under review.

patterns (Burkholz et al., 2021). GlyNet frames protein–glycan binding as a multi-output prediction problem (Carpenter et al., 2022), and LectinOracle demonstrates that learned glycan representations can generalize when paired with strong protein encoders (Lundstrom et al., 2022). These systems are important baselines, but they do not directly encode the specific topological cues that glycobiologists routinely inspect such as branch depth, terminality, linkage type, and distance to a non-reducing end.

This paper reconstructs and formalizes GLYDE, a small glycan language model built around that observation. The model takes canonical IUPAC-condensed glycans, extracts token-aligned topology features, injects those features into transformer attention and pooling, and couples the resulting encoder to a binding head and a lightweight decoder.

The main contribution is not size. GLYDE is intentionally small, with 380,659 parameters, a single encoder layer, and two decoder layers. The contribution is that explicit structural bias appears sufficient to close a large fraction of the gap between naive token models and biologically informed glycan reasoning under limited data. Across structural holdout experiments, topology features and topology-biased attention improve ranking, classification, and calibration. The same shared representation also supports target-conditioned glycan generation.

2 Background and related work

Machine learning on glycans is difficult because the input object is neither a simple sequence nor an unconstrained graph. Glycans contain repeated monosaccharides, branching, multiple linkage chemistries, and symbolic conventions such as SNFG and IUPAC-condensed notation (Neelamegham et al., 2019; Fujita et al., 2021). These properties make direct transfer from protein language modeling or genomic language modeling much harder. In proteins and DNA, local sequence neighborhoods help preserve the dominant information. In glycans, residues that are distant in a serialization can be functionally adjacent because they occupy different branches with shared terminal exposure.

Earlier computational work approached this by mining motifs or by learning graph embeddings. Coff et al. developed a subtree-mining method for identifying glycan motifs associated with downstream tasks (Coff et al., 2020). Burkholz et al. showed that graph convolutional neural networks can learn useful glycan representations directly from graph structure (Burkholz et al., 2021). GlyNet advanced binding prediction with a neural model trained on protein–glycan interactions (Carpenter et al., 2022). LectinOracle paired glycan and protein encoders to improve lectin–glycan binding prediction (Lundstrom et al., 2022). These studies establish that glycans are learnable objects and that deep models can recover meaningful biochemical patterns.

GLYDE occupies a different point in the design space. Instead of expecting the model to infer branch salience implicitly, it inserts prior biological knowledge directly into the representation pathway. The resulting model is closer in spirit to inductive-bias design than to brute-force scale. This is appropriate for viral glycan binding, where experimental datasets are relatively small, classification can be poor, and generalization is more important than raw interpolation on already seen structures.

3 Problem formulation and data representation

We consider supervised viral glycan binding prediction. Let g denote a glycan structure and v denote a virus identity. The task is to predict a binding score $\hat{y}(g, v) \in [0, 1]$ that ranks and classifies likely binding interactions between a given virus and a glycan.

3.1 Data sources and canonicalization

The project aggregates publicly available glycan-array measurements from the GlyGen glycan array repository and the Consortium for Functional Glycomics (CFG) glycan array experiments, then maps those measurements to canonical glycan structures using GlyTouCan identifiers and IUPAC-condensed strings (GlyGen Consortium, 2025; Consortium for Functional Glycomics, 2025; Fujita et al., 2021). This canonicalization step is essential. Glycan arrays often expose the same or closely related motifs under different naming conventions, and a canonical representation reduces accidental duplication while making topology extraction reproducible.

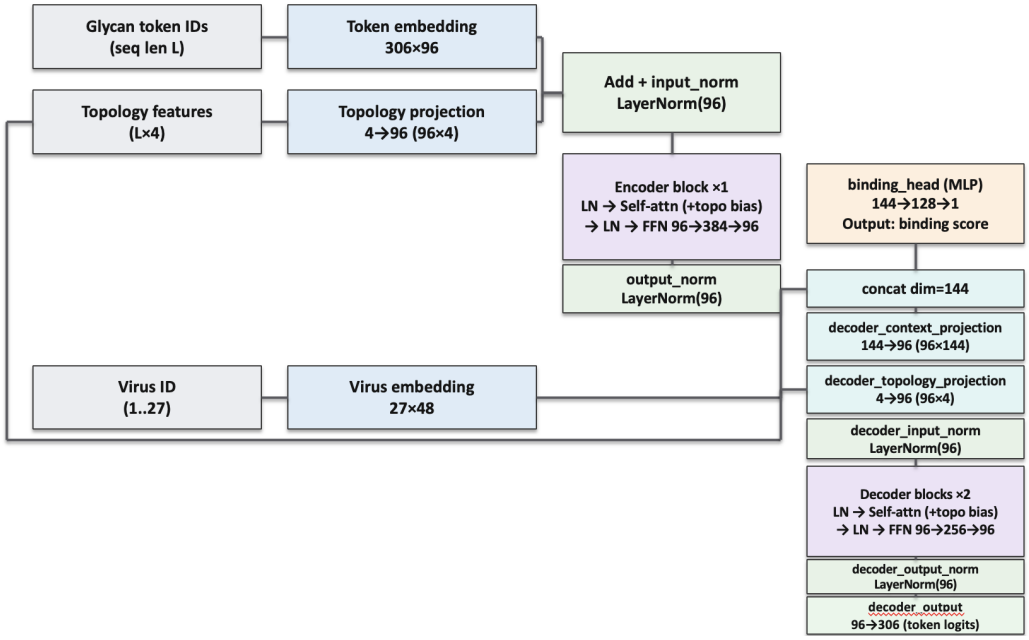

Figure 1: Topology-biased encoder–decoder architecture used in GLYDE. Token embeddings are fused with four topology features, passed through a one-layer encoder, pooled for binding prediction, and reused to condition a two-layer decoder for glycan generation. The project materials report 380,659 parameters in total.

3.2 Tokenization and topology features

Each glycan is tokenized from its IUPAC-condensed form into a residue/linkage sequence aligned to four topology features per position. For token i, the feature vector is

$$\mathbf{f}_i = [d_i \quad t_i \quad l_i \quad r_i]^\top , \tag{1}$$

where d_i is branch depth, t_i indicates whether the token lies at or near a terminal residue, l_i summarizes linkage chemistry, and r_i measures distance to a terminal motif. These are the four biologically motivated cues emphasized within the architecture of my glycan language model.

The structural holdout split is important because it better tests whether the model generalizes beyond simple motif memorization. In this setting, held-out glycans differ structurally from training glycans, so strong performance requires the model to learn patterns and regularities rather than memorize recurring sequences.

4 Model

4.1 Topology-aware token encoding

Figure 1 summarizes the core architecture. Token identities are embedded in a 96-dimensional space and fused with a learned linear projection of the four topology features. The resulting token representation is

$$\mathbf{x}_i = \mathrm{LayerNorm}(\mathrm{Emb}(t_i) + W_{\mathrm{topo}}\mathbf{f}_i) . \tag{2}$$

The project materials report a 306-token vocabulary for glycans and a 96-dimensional topology projection, yielding a compact but expressive embedding stage.

Component	Mathematical Formulation
Topology Feature Vector	$f_i = \begin{bmatrix} d_i \\ t_i \\ l_i \\ r_i \end{bmatrix}$
Topology-Aware Token Representation	$x_i = \mathrm{LayerNorm}(\mathrm{Emb}(t_i) + W_{\mathrm{topo}} f_i)$
Pairwise Topology Interaction	$g_{ij} = \begin{bmatrix} 1 - \lvert d_i - d_j \rvert \\ t_i t_j \\ l_i l_j \\ 1 - \lvert r_i - r_j \rvert \end{bmatrix}$
Topology-Biased Attention Score	$\mathrm{score}_{ij}^{(h)} = \frac{q_i^{(h)} k_j^{(h)}}{\sqrt{d_h}} + \lambda \left(w_h^\top g_{ij} \right)$
Topology-Biased Pooling Weight	$\alpha_i \propto 1 + \frac{w_d}{1 + d_i} + w_t t_i + w_l l_i + \frac{w_r}{1 + r_i}$

Figure 2: Topology-aware inductive bias in GLYDE. Panel A shows the mathematical formulation used for token features, pairwise topology interactions, attention bias, and structure-aware pooling. Panel B grounds those abstractions in a branched glycan with core and terminal sialic-acid-containing motifs highlighted.

4.2 Topology-biased attention and pooling

The encoder differs from a standard transformer in two places. First, pairwise topology interactions are computed explicitly:

$$\mathbf{g}_{ij} = \begin{bmatrix} 1 - |d_i - d_j| \\ t_i t_j \\ l_i l_j \\ 1 - |r_i - r_j| \end{bmatrix}. \tag{3}$$

These interactions are then added as a learned bias term to the self-attention score for each head,

$$\mathrm{score}_{ij}^{(h)} = \frac{\mathbf{q}_i^{(h)} \cdot \mathbf{k}_j^{(h)}}{\sqrt{d_h}} + \lambda \, \mathbf{w}_h^\top \mathbf{g}_{ij}. \tag{4}$$

This biases the attention map toward residues that are similar in depth, jointly terminal, linkage-compatible, or similarly close to branch termini. In biological terms, the model is encouraged to emphasize residues that are plausibly co-involved in receptor recognition.

Second, the glycan representation is pooled with topology-aware weights rather than plain mean pooling. The supplied formulation is

$$\alpha_i \propto 1 + \frac{w_d}{1 + d_i} + w_t t_i + w_l l_i + \frac{w_r}{1 + r_i}. \tag{5}$$

This favors residues near terminals and de-emphasizes deep core residues when appropriate. The pooled glycan embedding is concatenated with a learned virus embedding before a small multilayer perceptron predicts the final binding score.

4.3 Encoder, binding head, and decoder

The encoder uses a single topology-biased transformer block with feed-forward size $96 \to 384 \to 96$. The binding head consumes a 144-dimensional concatenation of glycan and virus representations and

Table 1: Reported hyperparameters and topology weights for GLYDE.

Component	Parameter	Value
Model	Glycan embedding dimension	96
Model	Virus embedding dimension	48
Model	Hidden width	128
Model	Dropout	0.1
Training	Batch size	128
Training	Learning rate	0.002
Training	Weight decay	0.0001
Training	Epochs	40
Training	Seed	13
Split	Protocol	structural_holdout
Topology	Depth weight	0.35
Topology	Terminal weight	0.50
Topology	Linkage weight	0.20
Topology	Distance weight	0.25
Tokenization	Max tokens	32
Tokenization	Unknown token	[UNK]

maps it through an MLP of size $144 \to 128 \to 1$. The decoder conditions on the encoder context and virus embedding through learned projections, then applies two transformer decoder blocks with feed-forward size $96 \to 256 \to 96$, finally producing logits over the 306-token glycan vocabulary.

5 Experimental setup

Table 1 summarizes the hyperparameters reported in the project materials. The training configuration is deliberately modest: 96-dimensional token embeddings, 48-dimensional virus embeddings, hidden width 128 in the binding head, dropout 0.1, batch size 128, learning rate 0.002, weight decay 10^{-4}, and 40 epochs. The structural split protocol is explicitly labeled `structural_holdout`. Topology weights emphasize terminality most strongly, followed by branch depth and terminal distance.

The model was trained on an NVIDIA DGX-1 system with V100 accelerators and did interactive generation and inference tasks on an Apple M3 Pro machine. The reproducible training time is roughly 50 minutes for 40 epochs. Evaluation uses standard ranking and calibration metrics, including AUROC, average precision (AP), F1, log loss, Brier score, expected calibration error (ECE), top-k precision, lift, and decoder perplexity.

6 Results

6.1 Ablation study

The ablation results show a consistent pattern that topology matters in these models. Starting from a baseline that uses glycan token embeddings, a virus embedding, and a simple MLP, test AUROC is 0.88, test AP is 0.62, and test F1 is 0.64. Adding explicit topology features increases these to 0.91, 0.72, and 0.70. Replacing plain attention with topology-biased attention lifts performance again to 0.94 AUROC, 0.83 AP, and 0.77 F1. The best joint model reaches 0.957 test AUROC, 0.905 test AP, and 0.837 test F1.

These gains are too structured to dismiss as noise. The move from 0.62 to 0.905 AP is especially important for experimental prioritization because average precision is sensitive to early ranking quality under class imbalance. In practical glycobiology workflows, the model is usually most valuable when it brings strong candidates to the top of the list rather than when it merely separates positives and negatives in aggregate.

6.2 Joint training versus decoder-only fine-tuning

The project materials also compare the best jointly trained model against a decoder-only fine-tuned variant. The two are nearly tied on AUROC: 0.957 for the baseline joint model and 0.958 for

Table 2: Performance of progressively stronger GLYDE variants on the reported structural-holdout evaluation.

Model variant	Val AUROC	Test AUROC	Test AP	Test F1
Baseline (token + virus embedding + MLP)	0.90	0.88	0.62	0.64
+ Topology features	0.93	0.91	0.72	0.70
+ Topology-biased attention	0.96	0.94	0.83	0.77
+ Joint training (best binding checkpoint)	**0.975**	**0.957**	**0.905**	**0.837**

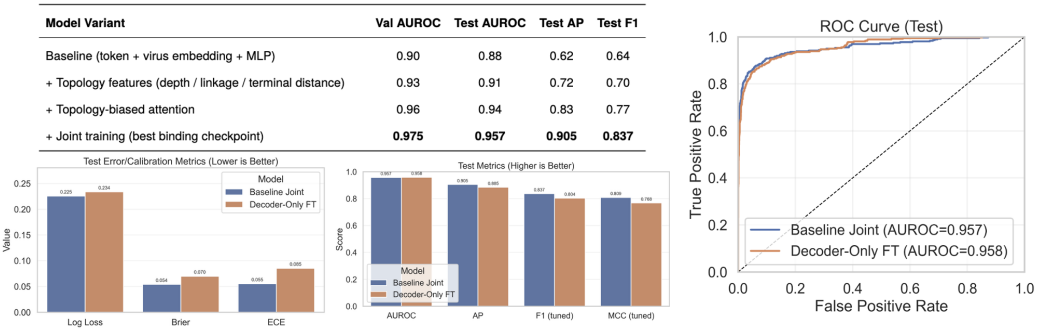

Figure 3: Reported performance summary. The left side shows the main ablation table and calibration bars; the right side shows ROC behavior and the comparison between joint training and decoder-only fine-tuning.

decoder-only fine-tuning. But AUROC alone hides the more relevant differences. The joint model attains better AP (0.905 vs. 0.885), better tuned F1 (0.837 vs. 0.804), better tuned MCC (0.809 vs. 0.768), lower log loss (0.225 vs. 0.234), lower Brier score (0.054 vs. 0.070), and lower ECE (0.055 vs. 0.085). In other words, decoder-only fine-tuning preserves ranking but degrades calibration and decision quality.

This distinction is very meaningful. A model that is slightly better at pairwise ranking but substantially worse calibrated is less useful for experimental triage, especially when follow-up assays are expensive. The calibration metrics suggest that a shared topology-aware representation constrains the model in a way that improves robustness outside the immediate decoder objective.

6.3 Ranking and generation diagnostics

The project shows that top-k precision remains close to 1.0 at the highest-ranked fraction of predictions and that the lift curve reaches approximately $6\times$ enrichment at the top of the ranking. Both observations support the same conclusion: the model is not merely well separated on average, but especially effective at surfacing strong candidates early. On the generation side, final decoder perplexity is low and similar across train and validation splits (1.066 train, 1.031 validation), which is consistent with learning stable glycan token regularities without obvious overfitting.

7 Target-conditioned glycan generation

The decoder extends the utility of the model beyond binary or scalar prediction. Figure 4 shows a prototype interface for target-conditioned glycan search. In the displayed example, the selected viral target is a coronavirus label, the input glycan begins with `Mana1-3(Mana1-6)Manb1-4GlcNAcb1-4GlcNAc`, and the current predicted affinity is 35.15%. The best generated candidate reaches 50.17%, essentially matching the requested target affinity of 50.21% with an absolute delta of 0.04%.

This does not constitute experimental validation of a novel binder. It does, however, demonstrate that the shared encoder–decoder representation can be used to navigate local glycan neighborhoods under a task-conditioned objective. For computational glycobiology, that is valuable.

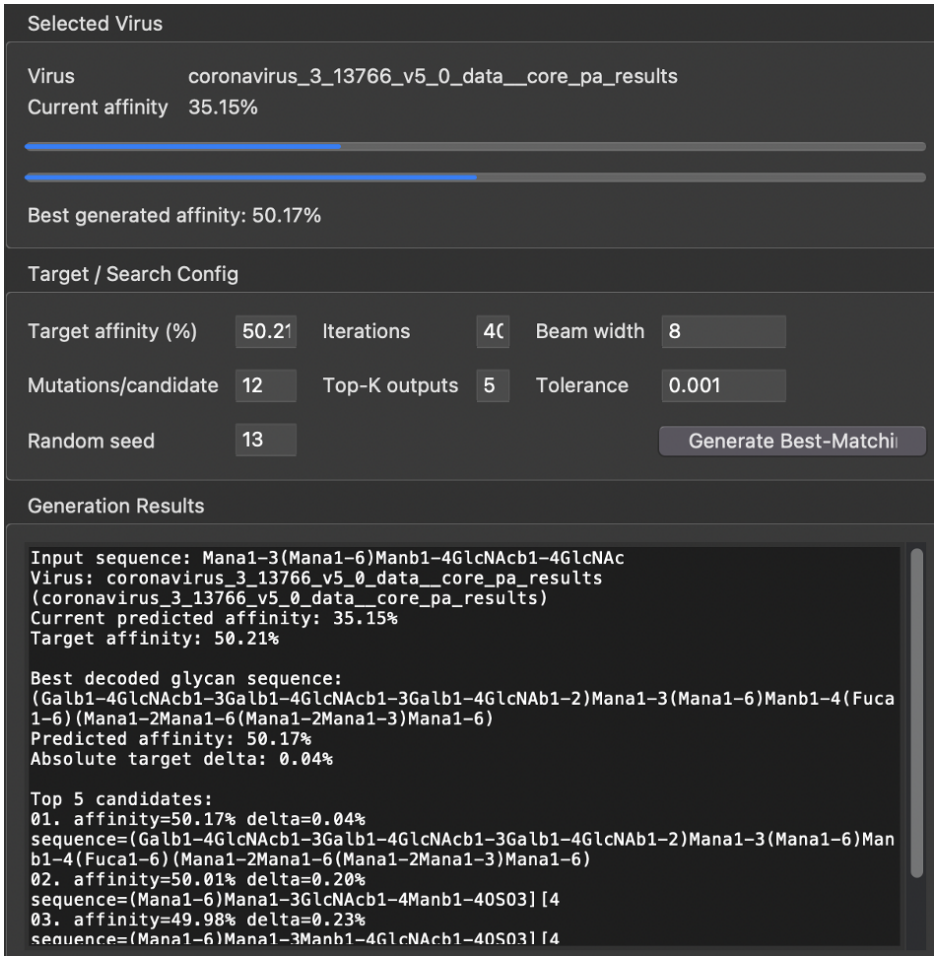

Figure 4: Prototype target-conditioned glycan generation interface. In the illustrated case, predicted affinity for a coronavirus target improves from 35.15% for the input glycan to 50.17% for the best generated candidate, closely matching a 50.21% target.

8 Discussion and limitations

The story is consistent throughout the project with the fact being that encoding biological structure directly into the model helps. Branch depth separates core from terminal branches, terminal indicators emphasize exposed binding sites, linkage chemistry preserves biologically meaningful distinctions such as α2-3 versus α2-6 presentation, and terminal distance shifts attention toward motifs that are likely to matter for viral recognition. The gains are strongest under structural holdout evaluation, which is the right place to expect inductive bias to matter.

At the same time, the current system has clear limitations. First, it is a small proof-of-concept model built from array-derived data. That is a strength from an efficiency standpoint but a limitation relative to large-scale multimodal receptor models. Second, viral context is encoded categorically rather than through protein sequence or structure, which constrains transfer to unseen viral families. Third, the generation results are computational only; no wet-lab validation is presented in the supplied materials. Fourth, some low-level implementation details, such as the exact multitask loss weighting, are not recoverable from the figures alone. This manuscript therefore reports the architecture and the measured outcomes conservatively rather than overstating undocumented choices.

Several extensions follow naturally. A stronger successor could condition on viral glycoprotein sequence or structure instead of a discrete virus ID, integrate explicit graph edges or three-dimensional glycan conformations, add uncertainty-aware calibration for ranking under limited labels, and evaluate virus-holdout rather than only structure-holdout generalization. It would also be useful to compare

against recent glycan graph transformers and against sequence-only language models trained on canonicalized glycans.

There is also a practical broader-impact angle. A system like GLYDE could help laboratories prioritize glycan arrays, generate receptor-binding hypotheses for emerging strains, and focus synthesis effort on candidate motifs. That is useful, but it should be used as a triage tool rather than as a substitute for experimental confirmation.

9 Conclusion

GLYDE shows that small, domain-aware glycan models can capture biologically meaningful receptor-binding structure without relying on large-scale brute force. By combining explicit topology features with transformer attention bias, structure-aware pooling, and a lightweight decoder, the model improves viral glycan binding prediction across ranking, calibration, and classification metrics. The strongest reported checkpoint reaches 0.957 test AUROC, 0.905 test AP, and 0.837 test F1 under structural holdout evaluation. The same shared representation also supports target-conditioned glycan generation. The central lesson is that careful inductive bias remains highly effective in glycobiology, especially when data are scarce and the biological object itself is richly structured.

References

Ajit Varki. Biological roles of glycans. *Glycobiology*, 27(1):3–49, 2017.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Lachlan Coff, Jeffrey Chan, Paul A. Ramsland, and Andrew J. Guy. Identifying glycan motifs using a novel subtree mining approach. *BMC Bioinformatics*, 21(1):42, 2020.

Rebekka Burkholz, John Quackenbush, and Daniel Bojar. Using graph convolutional neural networks to learn a representation for glycans. *Cell Reports*, 35(11):109251, 2021.

Eric J. Carpenter, Shaurya Seth, Noel Yue, Russell Greiner, and Ratmir Derda. GlyNet: a multi-task neural network for predicting protein–glycan interactions. *Chemical Science*, 13:6669–6686, 2022.

Jon Lundstrom, Emma Korhonen, Frederique Lisacek, and Daniel Bojar. LectinOracle: A generalizable deep learning model for lectin–glycan binding prediction. *Advanced Science*, 9(1):e2103807, 2022.

Sriram Neelamegham, Kiyoko F. Aoki-Kinoshita, Evan Bolton, Martin Frank, Frederique Lisacek, Thomas Lutteke, Noel O'Boyle, Nicolle H. Packer, Pamela Stanley, Philippe Toukach, et al. Updates to the symbol nomenclature for glycans guidelines. *Glycobiology*, 29(9):620–624, 2019.

Akihiro Fujita, Nobuyuki P. Aoki, Daisuke Shinmachi, Masaaki Matsubara, Shogo Tsuchiya, Masakazu Shiota, Kentaro Tanaka, Ichiro Yamada, Kiyoko F. Aoki-Kinoshita, and Hisashi Narimatsu. The international glycan repository GlyTouCan version 3.0. *Nucleic Acids Research*, 49(D1):D1529–D1533, 2021.

GlyGen Consortium. GlyGen glycan array data repository. `https://glygen.ccrc.uga.edu/ggarray/`.

Consortium for Functional Glycomics. CFG glycan array experiments. `https://www.functionalglycomics.org/glycan-array`.