SWEETBERT: EXPLORING BERT-BASED MODELS FOR IUPAC GLYCAN NOMENCLATURE MODELING

Irene Rubia-Rodríguez
Department of Applied Mathematics and Computer Science
Technical University of Denmark
Kongens Lyngby, Denmark
iruro@dtu.dk

Henrik Nielsen
Department of Health Technology
Technical University of Denmark
Kongens Lyngby, Denmark
henni@dtu.dk

Garry P. Gippert & Kristian Barrett
Department of Biotechnology and Biomedicine
Technical University of Denmark
Kongens Lyngby, Denmark
{garryg,kbaka}@dtu.dk

Bernard Henrissat
Department of Biotechnology and Biomedicine
Technical University of Denmark
Kongens Lyngby, Denmark
Architecture et Fonction des Macromolécules Biologiques
CNRS, Aix-Marseille University
Marseille, France
behen@dtu.dk

Ole Winther
Department of Applied Mathematics and Computer Science
Technical University of Denmark
Kongens Lyngby, Denmark
Department of Biology
Bioinformatics Centre, University of Copenhagen
Copenhagen, Denmark
olwi@dtu.dk

ABSTRACT

Glycans are the most abundant biomolecules on Earth, and participate in key processes in all living organisms. The chemical variability and topological complexity of their natural branched structures has been a challenge in computational glycobiology. As a tool for improving predictive models associated with glycobiology, we propose SweetBERT, a BERT-based language model for encoding glycan sequences which includes explicit information about the branching structure of the sequence. This is achieved by including a pseudo-graph representation in the input embeddings. Performance on downstream tasks by our model underscore promising results of Transformer architectures in addressing the complexities of glycan representation.

1 INTRODUCTION

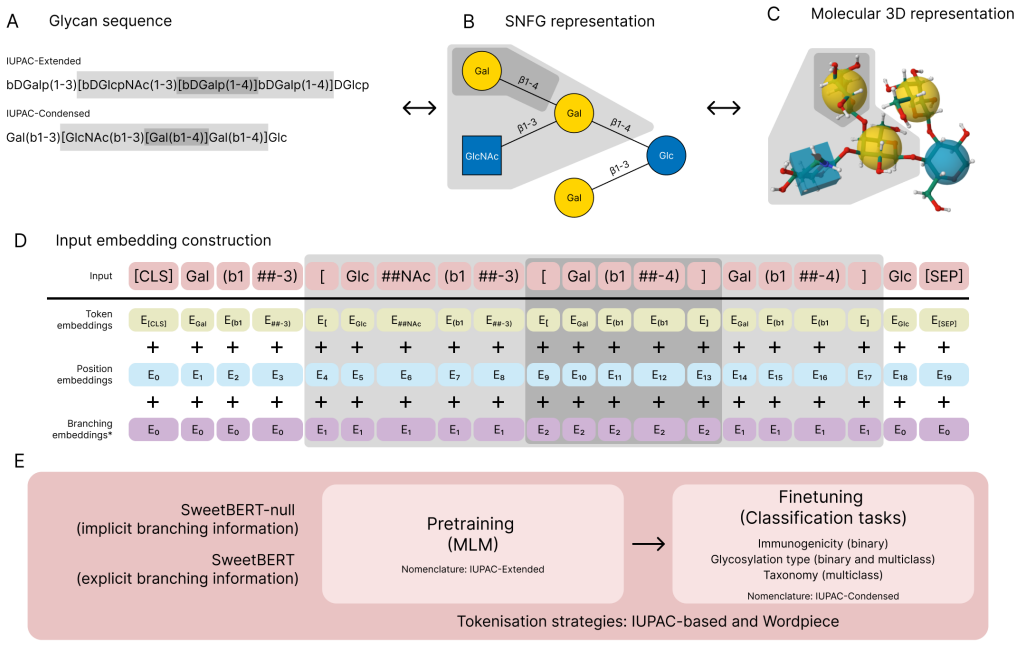


Figure 1: Summary of the study. A, B and C: example of a glycan represented as IUPAC-Extended and IUPAC-Condensed sequences (A), its equivalent in SNFG nomenclature (B) and its molecular 3D representation (C). D: schematic example of the tokenisation and construction of the input embedding from the sequence in (A). Depth in the branched structure is shown with different shades of gray; the darker the shade, the further the subsequence is in the tree of ramifications. Here, the branching embedding values E_0 represents the main sequence, E_1 the first level of ramification, and E_2 the second level of ramification. E: model performance evaluation. SweetBERT was pretrained as a masked language model (MLM), with generated and null branching embeddings, on glycan sequences in IUPAC-Extended nomenclature using 2 tokenisation strategies. It was afterwards fine-tuned on classification tasks (immunogenicity, glycosylation linkage type and taxonomy prediction) on glycan sequences in IUPAC-condensed nomenclature for the same 2 tokenisation strategies.

Glycans are one of the fundamental biomolecules in biology, along with proteins, lipids, and nucleic acids. They are complex molecules based on carbohydrates that can be found either isolated or covalently bound to other biomolecules, such as proteins and lipids, forming glycoproteins or glycolipids, or other chemical compounds like sulfates (Varki & Kornfeld, 2022; Varki & Gagneux, 2015). Glycans participate in most metabolic processes in living organisms, both normal and pathological, as well as in the regulation of protein stability and function, cell-to-cell recognition, and cell structure among others. Due to their importance in metabolism, glycans also play a role in disease development and inflammation processes, making carbohydrate-based molecules attractive candidates as alternatives to traditional drug targets (Hudak & Bertozzi, 2014).

Contrary to proteins or nucleic acids, whose sequence can be deduced from the gene encoding their structure, the expression of a specific glycan can lead to a variety of different functions, even within the same organism and across different tissues (Varki & Gagneux, 2017). Furthermore, glycans are not necessarily simply linear chains of their most basic unit, the monosaccharide, due to the possibility of linkage at almost any carbon in the glycan. This results in multiple branches and a huge diversity of glycans found in organisms. These peculiarities make carbohydrate sequencing a challenge that has slowed the development of research in glycobiology.

The continuous development of experimental and computational techniques within glycobiology has nonetheless led to the accumulation of significant amounts of data and the development of structured databases that bioinformatics approaches can leverage for further research. AI tools have

significantly enhanced bioinformatics by providing powerful methods for analyzing complex biological data. Machine learning algorithms can detect patterns and relationships in large datasets that are difficult for humans to discern, facilitating predictions about gene function, protein interactions, and disease associations (Jumper et al., 2021; Jia et al., 2024; Rehana et al., 2024). Most of these approaches have been focused on working with linear chains of amino acids and nucleotides.

SweetTalk (Bojar et al., 2020a), a glycan language model based on recurrent neural networks (RNN), has been successful in studying and classifying connectivity of glycans to proteins (glycosilation) and predicting their immunogenicity (i.e. the ability of glycan structures to trigger an immune response) from their representation in IUPAC-Condensed nomenclature (McNaught, 1996). This nomenclature is highly specific for glycans, and is far more structured and inflexible than the general IUPAC conventions, as it precisely encodes intricate branching patterns, stereochemistry, and glycosidic linkages to eliminate ambiguity. Based on SweetTalk, SweetOrigins (Bojar et al., 2020b) extends these capabilities by predicting the taxonomy (i.e. the systematic classification of glycan structures based on shared biosynthetic pathways and evolutionary relationships) of glycans. However, due to the complexity of polysaccharide linkage, these models present issues dealing with a high amount of branches (Bojar et al., 2020a). To overcome this problem, SweetNet (Burkholz et al., 2021) was developed, a graph convolutional neural network (GCNN) that accounts for the tree-like structure of glycans. It serves as a representation for the SNFG nomenclature system (Varki et al., 2015) obtained from the IUPAC-Condensed sequences, and outperforms previous machine learning tools.

BERT architectures (Devlin et al., 2018), initially developed for natural language processing (NLP) tasks, have successfully proven their capacity to understand chemical properties from linear strings (SMILES, SELFIES) (Chithrananda et al., 2020; Ahmad et al., 2022) by using Transformer blocks (Vaswani et al., 2017). Typically, RNNs have a limited ability to capture contextuality, whereas BERT allows for parallel processing of the sequence, resulting in a better understanding of the full context, and in faster training and inference times. BERT-based models also efficiently manage variable-length inputs with better interpretability through attention mechanisms. Although GNNs are computationally efficient and well-suited for graph-like structures, such as glycans, they tend to be task-specific and less flexible than transformer-based models. GNNs are often limited in their ability to generalize across tasks, constraining their adaptability across diverse biological tasks. In contrast, BERT-based models have the potential to manage the branched structure of glycans more dynamically while preserving the flexibility needed for various biological tasks. Considering the importance of glycans in biological processes a more versatile and generalizable model is crucial. In this work, we explore the capabilities of the transformer architecture for representing glycan sequences with SweetBERT —a BERT-based model that includes a pseudo-graph representation that can be extracted from linear text sequences in IUPAC glycan nomenclatures (McNaught, 1996) to account for the branched structure of glycans.

2 METHODS

The glycan sequences used in this study follow the IUPAC rules for glycans (McNaught, 1996), which is immediately recognisable by glycobiologists (Figure 1A). These sequences were obtained from the dataset based in Sugarbase (Bojar et al., 2021) provided in the SweetTalk repository. It contains 21296 glycans from several organisms in IUPAC-Condensed nomenclature. IUPAC nomenclatures represent the molecule linearly by connecting the monomers with their bonds between parenthesis ("(" and ")" characters) and the sequence of each ramification between brackets ("[" and "]" characters). Further ramifications can also occur within a branch, having then nested brackets as in the example shown in Figure 1. These 4 symbols are essential for the representation of the tree-like structure of the glycan in other glycan encodings, such as SNFG (Varki et al., 2015) (Figure 1B). In the original dataset, the sequences are presented in the IUPAC-Condensed format, which makes some abbreviation to the IUPAC-Extended format based in some assumptions (McNaught, 1996). For pretraining the models, the dataset was converted to the extended nomenclature, as it is the most explicit way of representing monosaccharides. Thus, it will provide more information about underlying grammar implied by the nomenclature system, avoiding contextual ambiguities.

For encoding the sequences into tokens, two different tokenisation strategies were considered (Figure 1E). The first one, similarly to the "glycowords" proposed by Bojar et al., follows the IUPAC

rules for naming glycans (Bojar et al., 2020a; McNaught, 1996). It considers each monosaccharide and bond as words, and splits them into elements that correspond to principal characteristics of the monomer (a and b, for alpha and beta configurations; p and f, for the ring type; etc.). The second tokenisation strategy we explored is based on a wordpiece algorithm that is widely used in BERT-based models (Song et al., 2020). It iterates over the data corpus splitting the words into subwords that individually correspond to a token. With this approach the number of identical occurrences is maximised while the overlaps between them are minimised. This results in two different corpus vocabulary sizes with a difference of almost one order of magnitude (1156 vs 210).

SweetBERT is based on $\text{BERT}_{\text{BASE}}$, which has 12 Transformer blocks (Vaswani et al., 2017) resulting in 110M parameters. It uses the embeddings to numerically represent the input sequence. The sum of three different types of embeddings gives this initial input embedding: token embedding, which encodes the values of each token (tokens are numbers assigned to the slices obtained after splitting the sequence. These slices can represent words, characters, or subwords depending on the tokenisation strategy); segment embedding, which identifies the tokens that belong to a type of sequence (in our case there is only one type); and positional embedding, which indicates the position of each token within the sequence.

In order to include information about the depth of ramification of the motifs composing a glycan, we added a fourth embedding to the previous ones (Figure 1D), the branching embedding. This has been inspired by the work of Wang et al. (2021), where the positional embedding is a depiction of the priority of mathematical operations. This can be understood as a pseudo-graph that represents the nested nature of mathematical equations, similar to glycan branched structures. In our case, we combine both the positional embedding and the branching embedding as the representation of the location of the tokens in the tree-like structure of the glycan. The branching embeddings were generated by searching for the tokens that encode the characters "[" and "]", and encoding the level of ramification of the tokens contained between them with scalars, being 0 those that are present on the main branch, 1 for the first level of branching, 2 for the second, etc. See Figure 1 and section A.1 for more details.

3 RESULTS

3.1 PRE-TRAINING

Table A2 presents the performance metrics of SweetBERT models based on perplexity and loss. Perplexity measures the ability of the model to predict token sequences, with lower values indicating higher confidence in predictions. Loss, on the other hand, quantifies the difference between the tokens predicted by the model for the masked positions and the actual tokens, where lower loss reflects a better fit to the training data. Including explicit branching information shows the best overall performance, with the lowest loss values (0.1583 and 0.1639 for training and validation respectively when tokenising with the wordpiece algorithm and 0.0842 and 0.0784 for the IUPAC-based tokenisation). This suggests that the pseudo-graph embedding helps the model to capture the underlying data structure more efficiently, even with very similar perplexity values during training.

Although, at this stage of the analysis it seems that providing explicit branching level information in the input embeddings may improve the performance, this will be corroborated by downstream classification tasks. SweetBERT performance will be compared to SweetTalk using the same training, validation and test dataset splits explained in A.2.

3.2 FINE-TUNING

We fine-tuned SweetBERT using three specific datasets representing immunogenicity, protein O- and N-linkage and taxonomy data for glycans which were previously used to test SweetTalk by Bojar et al. These contain IUPAC-condensed sequences, with binary labels for immunogenicity (1370 sequences), "N", "O" and "free" labels for glycosilation (1686 sequences), and the taxonomy information at 8 different levels (domain, kingdom, phylum, class, order, family, genus and species) (12674 sequences). With regards of glycosilation type labels, "N" refers to glycans liked to the N present in asparagine residues in proteins, "O" to those attached to serine or threonine residues, and "free" are glycans that exist independently without covalent links to proteins or lipids. These datasets were split following the proportions mentioned in A.2 with an equivalent proportion of labels in all

splits. It is important to note that SweetBERT was fine-tuned with datasets that included sequences in a nomenclature system that differs slightly from the one used for pretraining (IUPAC-extended for pretraning and IUPAC-condensed for fine-tuning), showcasing also the capacity of BERT-based models to handle tokens outside of the vocabulary (OOV). This cannot be done with the current implementation of SweetTalk, as it cannot handle OOVs. For the taxonomy dataset splits, due also to the implementation of SweetTalk in SweetOrigins, the test set was previously obtained from the complete dataset, and the splitting into the training and validation sets is performed by SweetOrigins after the tokenisation of the remaining sequences for every taxonomy level independently, dropping those that appear less than 5 times. This means that sequences that are used as training examples in one taxonomy level might be in the validation set for another one.

For glycosylation type classification, to compare our models with the reported performance of SweetTalk, the dataset was preprocessed in such a way that glycans tagged as "free" were dropped from the final input, reducing the problem into that of binary classification as only O- and N-linkages are taking into account. In this case, both SweetBERT and SweetTalk reached perfect scores in both accuracy and MCC (Table 1), showing no differences in the performance when using the wordpiece tokenisation. We also explored the capacity of the models on a multi-class classification task on the same dataset, by recovering the glycans tagged as "free" (less than 10% of the data). In this task, also using the wordpiece tokenisation, SweetBERT outperforms the other models in accuracy and MCC scores, with a very similar performance of SweetBERT-null compared to with SweetTalk. This improvement becomes clearer when calculating the balanced accuracy due to class imbalance. Also, the confusion matrices (Figure A2) show that, even though the accuracy and MCC are very close for these models, SweetBERT provides a more balanced precision for all three classes.

SweetTalk is also outperformed by SweetBERT and has very similar metrics as SweetBERT-null in the immunogenicity classifier for the wordpiece tokenisation. The confusion matrices (Figure A2) show that both BERT-based models achieve a more balanced classification than SweetTalk, although the three models seem to predict better the immunogenic-positive glycan sequences. This also suggests that the BERT-based models find clearer features within these sequences used to make predictions.

In the taxonomy prediction of glycans SweetBERT-null outperforms SweetTalk, with SweetBERT showing a performance very similar and often superior to SweetTalk. When comparing SweetBERT and SweetBERT-null, even though the performance is worse for all tasks using the IUPAC-based tokenisation, the ranks of the metrics are consistent within the tokenisation strategies, giving some insights on whether the explicit information of branching is actually needed or not.

Table 1: Table with the average accuracy and Matthews Correlation Coefficient (MCC) after 5 runs. MCC was calculated for multi-class classifiers according to Gorodkin (2004). *Due to the unbalanced dataset for "free"-labeled sequences, balanced accuracy (in parentheses) was also obtained.

Architecture	SweetBERT-null (Wordpiece)		SweetBERT-null(IUPAC-based)		SweetBERT(Wordpiece)		SweetBERT(IUPAC-based)		SweetTalk	
Task	Acc.	MCC	Acc.	MCC	Acc.	MCC	Acc.	MCC	Acc.	MCC
Glycosylation (binary)	0.9901	0.9800	0.9640	0.9276	**1.0000**	**1.0000**	0.9640	0.9276	**1.0000**	**1.0000**
Glycosylation (3 classes)*	0.9254 (0.8713)	0.8736	0.8982 (0.8727)	0.8378	**0.9740 (0.9356)**	**0.9529**	0.9207 (0.9072)	0.8628	0.9728 (0.9014)	0.9501
Immunogenicity	0.8864	0.7801	0.8106	0.6557	**0.8985**	**0.7986**	0.8364	0.6714	0.8894	0.7832
Domain	**0.9107**	**0.8248**	0.8470	0.7006	0.8915	0.7847	0.7522	0.5333	0.7587	0.8035
Kingdom	**0.8517**	**0.7917**	0.7174	0.5967	0.8295	0.7567	0.6789	0.5308	0.8207	0.7445
Phylum	0.7491	**0.6849**	0.6422	0.5386	**0.7524**	0.6805	0.5603	0.4162	0.7521	0.6809
Class	**0.6533**	**0.6008**	0.5516	0.4855	0.5863	0.5278	0.4426	0.3507	0.5814	0.5185
Order	**0.4966**	**0.4695**	0.3950	0.3605	0.4079	0.3789	0.2913	0.2436	0.4354	0.4056
Family	**0.4208**	**0.4004**	0.3278	0.2994	0.3674	0.3463	0.2579	0.2205	0.4090	0.3868
Genus	**0.3657**	**0.3534**	0.2715	0.2554	0.3125	0.3023	0.1952	0.1772	0.3085	0.2925
Species	**0.2872**	**0.2759**	0.1802	0.1636	0.2221	0.2109	0.1530	0.1374	0.2702	0.2548

4 Discussion

SweetTalk tokenisation strategy considers glycowords as overlapping triplets composed by three monosaccharides and two bonds. This process removes "[" and "]", which are the characters that inform about a branch occurring in the sequence. In other words, branching is not taken into account in the tokenisation. This implies that two different sequences, one with branches and one completely linear, can be interpreted by this model to be the same if their composing monosaccharides follow

the same order in the linear sequence. This can be supported by the analysis made to understand how including branching information plays a role in model performance in Section A.4.

In the case of linkage prediction including the three classes, SweetBERT-null and SweetTalk appear to be more sensitive to unbalanced datasets, probably because the third label can be considered as noise for these two models (Figure A1). However, including explicit information about the branching level in SweetBERT helps the model to focus the attention towards the main skeleton of the sequence, explaining its overall better performance despite of branching level. This also relates to what is seen with the immunogenicity confusion matrices (Figure A2).

For immunogenicity prediction, SweetTalk outperformed SweetBERT models on the sequences without any branches. This is consistent with the idea that effectively de-branching the training data causes it to appear more highly redundant. Though SweetBERT-null is not including explicit information about branches, implicit information about branching may be derived from tokenisation of the characters "[" and "]". Additionally, this model shows similar performances to SweetBERT for this classification task, although the values of its metrics are always lower.

When evaluating taxonomy prediction, we encountered additional challenges due to class reduction and test set generation. SweetTalk removes classes that appear fewer than five times during its train/validation split. Since the original implementation only creates training and validation sets, we added a previous step to extract a test set. Consequently, some classes appear exclusively in the test set but not in the training or validation sets—and vice versa—leading to discrepancies in performance. Addressing these issues will be a focus of future work to ensure more consistent evaluation across taxonomy levels.

5 Conclusions

In this work, we introduced SweetBERT, a novel BERT-based model that explicitly incorporates a pseudo-graph representation of glycan sequences by embedding their branching information from IUPAC nomenclature. By leveraging Transformer architectures to capture long-range contextual dependencies and structural intricacies, SweetBERT outperforms traditional RNN-based approaches in language modeling and downstream classification tasks—including immunogenicity, glycosylation type, and taxonomy prediction—while also demonstrating robust handling of out-of-vocabulary tokens across different nomenclature variants. We explored how the branching information affects the performance of the models, demonstrating that branching information can guide the attention of the model producing more balanced classifiers in some tasks. These findings underscore the potential of Transformer-based models in advancing computational glycobiology and pave the way for developing more sophisticated biomolecular analysis tools, such as multitask models or multi-modal approaches.

6 Acknowledgements

This work was supported by Novo Nordisk Foundation (Data Science Collaborative Research Programme 2022 0077058) and the Pioneer Centre for AI (DNRF grant number P1).

References

Walid Ahmad, Elana Simon, Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. Chemberta-2: Towards chemical foundation models. *arXiv preprint arXiv:2209.01712*, 2022.

Daniel Bojar, Diogo M Camacho, and James J Collins. Using natural language processing to learn the grammar of glycans. *bioRxiv preprint*, pp. 2020.01.10.902114, 2020a.

Daniel Bojar, Rani K Powers, Diogo M Camacho, and James J Collins. SweetOrigins: Extracting evolutionary information from glycans. *bioRxiv preprint*, pp. 2020.04.08.031948, 2020b.

Daniel Bojar, Rani K Powers, Diogo M Camacho, and James J Collins. Deep-learning resources for studying glycan-mediated host-microbe interactions. *Cell Host Microbe*, 29(1):132–144.e3, 2021.

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

Tadeusz Caliński and Jerzy Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974.

Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. Chemberta: large-scale self-supervised pretraining for molecular property prediction. *arXiv preprint arXiv:2010.09885*, 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint*, pp. arXiv:1810.04805, 2018.

J. Gorodkin. Comparing two k-category assignments by a k-category correlation coefficient. *Computational Biology and Chemistry*, 28(5):367–374, 2004. ISSN 1476-9271. doi: https://doi.org/10.1016/j.compbiolchem.2004.09.006. URL https://www.sciencedirect.com/science/article/pii/S1476927104000799.

Jason E Hudak and Carolyn R Bertozzi. Glycotherapy: New advances inspire a reemergence of glycans in medicine. *Chem. Biol.*, 21(1):16–37, 2014.

Xianghu Jia, Weiwen Luo, Jiaqi Li, Jieqi Xing, Hongjie Sun, Shunyao Wu, and Xiaoquan Su. A deep learning framework for predicting disease-gene associations with functional modules and graph augmentation. *BMC bioinformatics*, 25(1):214, 2024.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Hong Liu, Sang Michael Xie, Zhiyuan Li, and Tengyu Ma. Same pre-training loss, better downstream: Implicit bias matters for language models. In *International Conference on Machine Learning*, pp. 22188–22214. PMLR, 2023.

Alan D McNaught. Nomenclature of carbohydrates (iupac recommendations 1996). *Pure and Applied Chemistry*, 68(10):1919–2008, 1996.

Hasin Rehana, Nur Bengisu Çam, Mert Basmaci, Jie Zheng, Christianah Jemiyo, Yongqun He, Arzucan Özgür, and Junguk Hur. Evaluating GPT and BERT models for protein–protein interaction identification in biomedical text. *Bioinformatics Advances*, 4(1):vbae133, 2024.

Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987. ISSN 0377-0427. doi: https://doi.org/10.1016/0377-0427(87)90125-7. URL https://www.sciencedirect.com/science/article/pii/0377042787901257.

Xinying Song, Alex Salcianu, Yang Song, Dave Dopson, and Denny Zhou. Fast wordpiece tokenization. *arXiv preprint arXiv:2012.15524*, 2020.

A Varki and P Gagneux. Biological functions of glycans. In *Essentials of Glycobiology*. Cold Spring Harbor Laboratory Press, Cold Spring Harbor, NY, 2015.

Ajit Varki and Pascal Gagneux. Biological functions of glycans. In *Essentials of Glycobiology [Internet]. 3rd edition.* Cold Spring Harbor Laboratory Press, Nueva York, NY, Estados Unidos de América, 2017.

Ajit Varki and Stuart Kornfeld. Historical background and overview. In Ajit Varki, Richard D Cummings, Jeffrey D Esko, Pamela Stanley, Gerald W Hart, Markus Aebi, Debra Mohnen, Taroh Kinoshita, Nicolle H Packer, James H Prestegard, Ronald L Schnaar, and Peter H Seeberger (eds.), *Essentials of Glycobiology [Internet]. 4th edition.* Cold Spring Harbor Laboratory Press, Cold Spring Harbor, NY, 2022.

Ajit Varki, Richard D Cummings, Markus Aebi, Nicole H Packer, Peter H Seeberger, Jeffrey D Esko, Pamela Stanley, Gerald Hart, Alan Darvill, Taroh Kinoshita, et al. Symbol nomenclature for graphical representations of glycans. *Glycobiology*, 25(12):1323–1324, 2015.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint*, pp. 1706.03762, 2017.

Z Wang, A S Lan, and R G Baraniuk. Mathematical formula representation via tree embeddings. In *iTextbooks@ AIED*, pp. 121–133. 2021.

A Appendix

A.1 Branching embedding

To construct the branching embedding, we initialize it by considering all tokens in the sequence as branch level 0. Afterwards, we iteratively scan the IUPAC sequence identifying the position of the tokens that encode the characters "[" and "]" and the tokens located between these two. As shown before, these are the characters that represent which part of the sequence is located within a branch. We will call them "opening and closing branching tokens" for the sake of explanation. Every iteration finds the sequence of tokens with the deepest level of branching that starts with an opening branching token that is assigned to a branch level 0 in the branching embedding, and adds 1 at their positions in the branching embedding for representing a level of nested branching. Therefore, the higher the subramifications in a branch, the higher the branching level. This is performed in such way that the tokens located in the deepest nested branches are firstly identified by looking for any opening branching token in the branch subsequence and whether the next branching token found in the sequence is an opening or a closing one. The search for further subbranching is continued until no more opening branching tokens are found in the nested subsequence. Once the deepest level of subbranching has been found, 1 is added to the branching embedding on the positions that correspond to the tokens in this nested sequence. Then the process is reversed to find the parent subsequences and 1 is added to the branching embedding in the positions of the tokens contained within the opening branching token followed by a closing branching token whose both branching level codifications are 0.

Schematically, the process to produce the branching embedding is as follows:

1. Constructing a template null vector for the embedding with the same length as the token embedding.
2. Looking for the positions of all "[" tokens and their corresponding "]" token in the token embedding.
3. Adding +1 to the template in the positions that are contained between the positions of the pairs "["-"]" found in the previous step.

The final branching embedding will have a value equal or higher than 2 for the sub-branches, 1 for the primary branches and 0 for the skeleton of the sequence.

A.2 Experiments setup

We trained four models: SweetBERT including explicit branching information by generating the branching embedding and SweetBERT with a null branching embedding (considering that the final input embedding is constructed by adding all the embeddings, this will be equivalent to not including the branching embedding) using two tokenisation strategies: wordpiece and IUPAC-based. Each model was pre-trained in 6 parallelised Nvidia 11 GB GPUs for 100 epochs, with a batch size of 12 and an initial learning rate of 5×10^{-5} using the Adam optimiser (Kingma, 2014) with a weight decay of 0, beta parameters 0.9 and 0.999 and epsilon 1×10^{-8}.

The data was split in two, with 80% of the sequences for training and 20% for validation. The models were trained as masked language models with 20% of the tokens randomly masked.

To compare the performance of these two models and the tokenisation strategies adopted, further fine-tuning for classification tasks was done. It has been showed that cross-entropy loss and perplexity scores in pre-training models are not sufficient for comparing performance (Liu et al., 2023). This is where downstream tasks play a decisive role in performance analysis. Following Bojar's work (Bojar et al., 2020a;b), SweetBERT was fine-tuned for glycan immunogenicity, glycosylation type (O-/N-linkage to proteins) and taxonomy prediction. We used again the datasets provided by Bojar et al. (2020a;b) which are also based on the Sugarbase dataset (Bojar et al., 2021). The splits for training, validation and testing follow proportions of 80%, 10% and 10%, respectively. The sequences in these datasets are presented in the IUPAC-condensed nomenclature format.

We provide in table A1 the hyperparameters used for finetuning SweetBERT-null and SweetBERT. For finetuning SweetTalk, the same hyperparameters as the ones provided by Bojar et al. (2020a;b) were used.

Table A1: Summary of the hyperparameters used for fine-tuning for SweetBERT-null and SweetBERT.

TASK	EPOCHS	LEARNING RATE	WEIGHT DECAY	BATCH SIZE
IMMUNOGENICITY	20	2E-5	0.001	14
GLYCOSYLATION (BINARY)	50	1E-4	0.01	14
GLYCOSYLATION (3 CLASSES)	50	1E-4	0.01	14
TAXONOMY	30	1E-4	0.001	14
TAXONOMY (SPECIES)	50	1E-4	0.001	14

A.3 SWEETBERT PRETRAINING METRICS AND ANALYSIS

On top of comparing the perplexity and loss metrics, we also performed a t-SNE clustering analysis to explore the representation of branched sequences across the different models trained (Table A3). We compared them using the Calinski-Harabasz (Caliński & Harabasz, 1974) and the silhouette (Rousseeuw, 1987) scores. The Calinski-Harabasz score measures the quality of the clusters, considering the ratio of the sum of between-cluster dispersion to within-cluster dispersion. The higher the score, the more compact and separated the clusters are. On the other hand, Silhouette score measures how well-separated and compact clusters are with values from -1 to 1, where negative values indicate that samples have been assigned to the wrong cluster and 0 indicates overlapping clusters. In these terms, SweetBERT trained with the generated branching embeddings performs the best (A3), achieving the highest Calinski-Harabasz and Silhouette scores, although the latter are negative values relatively close to 0, which indicates well-defined and compact clusters but overlapping.

PCA shows that the IUPAC-based token models require fewer components to explain 90% of the variance (12-13 components) compared to the wordpiece models (21 components) (Table A3). This suggests that IUPAC-based tokenisation creates a more efficient data representation, capturing more meaningful variance with fewer dimensions, which may be due to the vocabulary size generated by this tokenisation strategy.

Table A2: Performance metrics of SweetBERT for both tokenisation strategies in pretraining.

		PERPLEXITY (TRAINING)	PERPLEXITY (VALIDATION)	LOSS (TRAINING)	LOSS (VALIDATION)
WORDPIECE	SWEETBERT	1.5967	1.5997	**0.1583**	**0.1639**
	SWEETBERT-NULL	**1.5163**	**1.5062**	0.167	0.1732
IUPAC-BASED	SWEETBERT	**1.9782**	1.9564	**0.0858**	**0.0784**
	SWEEETBERT-NULL	1.9825	**1.9487**	0.0964	0.0877

Table A3: Metrics on the t-SNE and PCA analyses of SweetBERT-null and SweetBERT learned embeddings for both tokenisation strategies after pretraining.

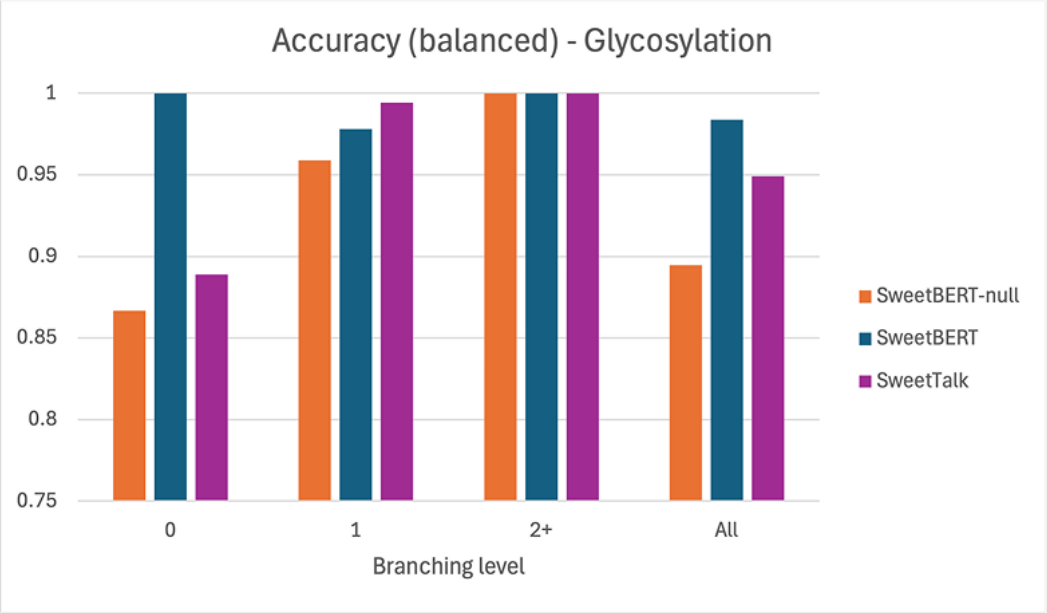
		Silhouette Score (t-SNE)	Calinski-Harabasz Score (t-SNE)	PC var>90%
Wordpiece	SweetBERT	**-0.0970**	**120.1990**	**21**
	SweetBERT-null	-0.1039	85.3525	**21**
IUPAC-based	SweetBERT	**-0.0987**	**102.8422**	13
	SweetBERT-null	-0.1081	86.0362	**12**

A.4 Branching analysis

Figure A1 shows the average of the balanced accuracy for the tasks of glycosylation and immunogenicity classification for SweetBERT, SweetBERT-null and SweetTalk, using for the first two the wordpiece tokenisation. The test dataset was filtered by the maximum branching level on the sequences. 0 indicates that there are no branches, 1 that only primary branches are present, 2+ that there are at least second-level branches, and All includes all sequences without filtering.

Figure A2 shows the accuracy values of the best runs for SweetBERT, SweetBERT-null and SweetTalk for the same classification tasks.

A



B

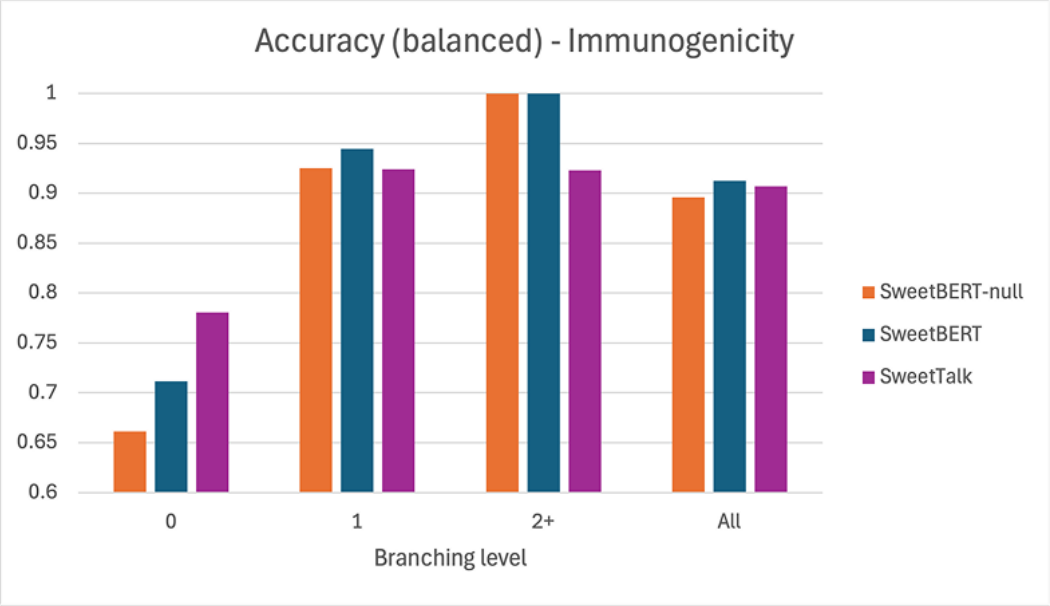


Figure A1: Branching analysis of the balanced accuracy of the three models for A) glycosylation classification (3 classes); and B) immunogenicity classification.

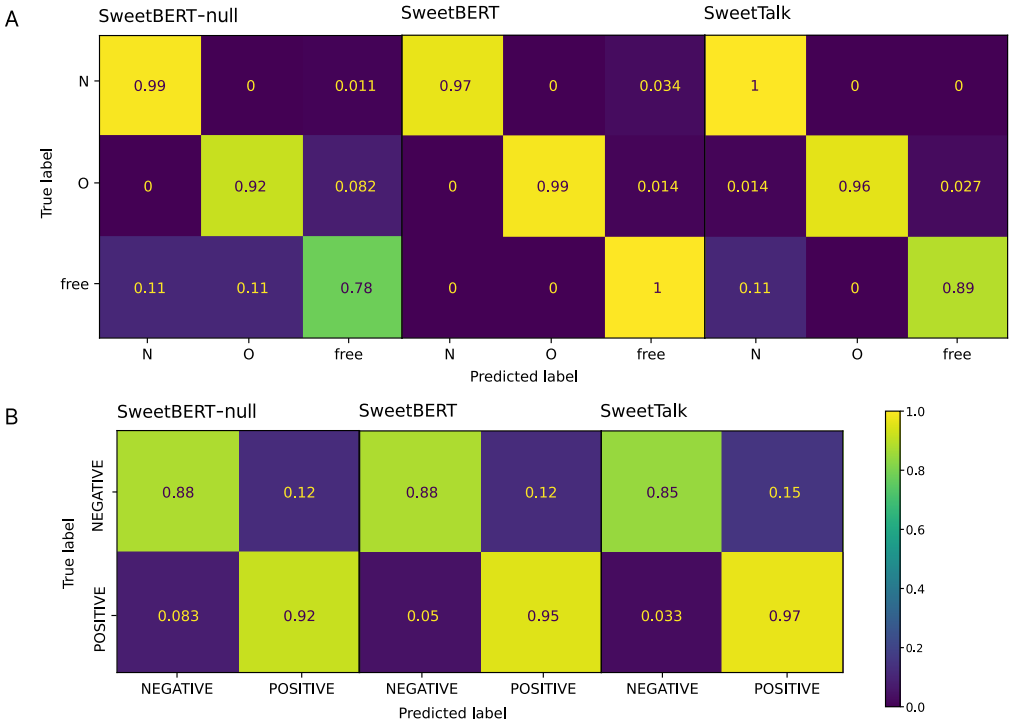


Figure A2: Confusion matrices of the best runs for SweetBERT-null, SweetBERT using wordpiece tokenisation and SweetTalk for A) glycosylation classification (3 classes); and B) immunogenicity classification. The values represent the proportion of data with each predicted label and its corresponding ground truth label. Values closer to 1 are represented in yellow while the lowest values are shown in deep purple.