HyperHELM: Hyperbolic Hierarchy Encoding for mRNA Language Modeling

Max van Spengler[*,1,2], **Artem Moskalev**[1], **Tommaso Mansi**[1], **Mangal Prakash**[†,1], **Rui Liao**[†,1]
[1]Johnson & Johnson Innovative Medicine
[2]University of Amsterdam
m.w.f.vanspengler@uva.nl

Abstract

Language models are increasingly applied to biological sequences like proteins and mRNA, yet their default Euclidean geometry may mismatch the hierarchical structures inherent to biological data. While hyperbolic geometry provides a better alternative for accommodating hierarchical data, it has yet to find a way into language modeling for mRNA sequences. In this work, we introduce HyperHELM, a framework that implements masked language model pre-training in hyperbolic space for mRNA sequences. Using a hybrid design with hyperbolic layers atop Euclidean backbones, HyperHELM aligns representations with biological hierarchy defined by the relationship between mRNA and amino acids. Across multiple multi-species datasets, it outperforms Euclidean baselines on 8 of 9 tasks involving property prediction, with 10% improvement on average, and excels in out-of-distribution generalization to long, low-GC sequences; for antibody region annotation, it surpasses hierarchy-aware Euclidean models by 3% in annotation accuracy. Our results highlight hyperbolic geometry as an effective inductive bias for language modeling of mRNA sequences.

1 Introduction

Language models have been increasingly applied to biological sequence data, fueled by the growth of large-scale omics datasets [40, 10, 5]. While originally designed for natural language, these models demonstrate promising performance in capturing dependencies within DNA [83, 52, 51, 5], RNA [10, 60, 78, 79], and protein sequences [40, 23]. The biological sequences, however, are structured differently from natural language, particularly in their hierarchical organization, where nucleotides or amino acids form motifs that can be nested within larger functional groups [7]. In this work, we take the rapidly expanding therapeutic domain of mRNA, where the tree of the codon–amino acid hierarchy plays a key role in determining the biophysical properties of mRNA sequences and their expressed proteins [18], and we focus on encoding this hierarchy by leveraging the geometry of the representation space of a bio-language model.

While standard language models rely on Euclidean geometry, the number of concepts in hierarchies grows exponentially, outpacing the polynomial expansion of Euclidean volumes [46, 47]. This can severely limit the representation capacity of a model and hinder the generalization [42]. In contrast, the volume of the hyperbolic space expands exponentially, maintaining well-separated representations across different branches of the hierarchy and reducing distortion in hierarchical relationships. The advantages of hyperbolic geometry are demonstrated in graph representation learning [12] and computer vision [48], and are beginning to inform language modeling [33, 32], though they have yet to be systematically applied to mRNA data.

[*]This work was done while the author was an intern at Johnson & Johnson.
[†]Equal contribution as last authors.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) NEGEL Workshop .

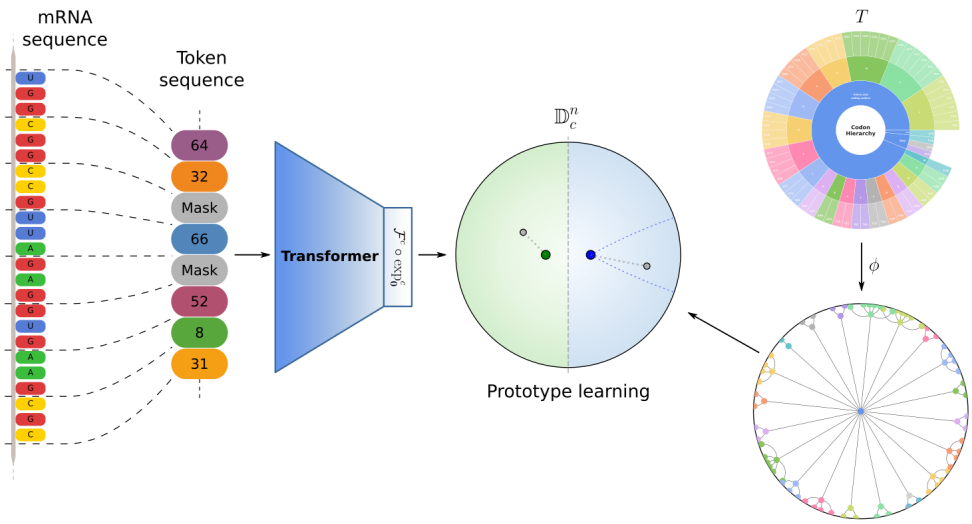

Figure 1: **High-level overview of the HyperHELM method** for MLM. The method consists of three main components: 1) the language modeling of mRNA, where a sequence transformer is used to obtain token representations, as shown in the *left*; 2) a hyperbolic embedding of the codon hierarchy is generated to serve as prototypes for guiding the language model during pre-training, shown on the *right*; and 3) hyperbolic hierarchical prototype learning, where the prototypes are used to predict the true label of masked tokens, visualized in the *center*.

In this work, we present Hyperbolic Hierarchical Encoding for mRNA Language Modeling (HyperHELM), a hyperbolic language-modeling framework for mRNA sequences. In HyperHELM, we project token representations onto the Poincaré ball [53] and pre-train a language model with masked language modeling (MLM) objective directly in hyperbolic space (Figure 1). Rather than making the entire model hyperbolic, we keep the backbone Euclidean and project only the final-layer representations, thus retaining hardware efficiency while leveraging the hierarchical inductive bias of hyperbolic geometry.

For hyperbolic MLM pre-training, we mask a portion of input tokens and use a modular hyperbolic prediction head that scores candidates while respecting hierarchical relations. In particular, we instantiate three head options for hyperbolic learning: hyperbolic multinomial logistic regression (MLR) [25], distance-to-prototype learning [28], and prototype classifiers based on hyperbolic entailment cones [24]. While [24] primarily introduces entailment cones as a means to model hierarchical relations, our work extends this concept further by exploring the use of similarity functions beyond hyperbolic distances, aiming to capture richer relational structures. Moreover, the adaption of these hyperbolic heads for MLM pre-training of bio-language models has never been explored before. The resulting hyperbolic latent space with hierarchy-aware MLM pre-training aligns representation geometry with the codon–amino-acid structure, clustering synonymous codons under their amino-acid parents and separating non-coding tokens (Figure 1). To our knowledge, HyperHELM is the first systematic development of hyperbolic language models for mRNA sequence data.

We conduct experiments to compare our HyperHELM with its standard Euclidean and hierarchical language modeling counterparts. We keep the language model backbone architecture and pre-training dataset fixed for all models, to isolate the impact of hyperbolic geometry on hierarchy learning. We evaluate the pre-trained models on 10 diverse multi-species mRNA datasets for downstream property prediction and region annotation tasks. Across 8 out of 9 property prediction tasks, the hyperbolic approach consistently outperforms its Euclidean counterparts, even when the latter is trained to be hierarchy-aware [78], achieving an average improvement of 10%. We also observe that in property prediction tasks, our hyperbolic language model generalizes exceptionally well to out-of-distribution data, maintaining strong performance even on long sequences with low GC-content, where standard bio-language models tend to struggle. Moreover, for the task of antibody region annotation, HyperHELM surpasses another hierarchy-aware Euclidean baseline by 3%. Our

experimental results suggest that hyperbolic geometry provides a powerful inductive bias for capturing hierarchical structures in mRNA sequences.

To sum up, we make the following contributions:

- We explore hierarchical learning for bio-language models through the lens of the hyperbolic geometry, aiming to align the structure of its representation space with the hierarchical structure of mRNA sequences.
- We adapt, implement, and evaluate multiple hyperbolic learning methods for masked language pre-training of a language model on mRNA sequences.
- We experimentally demonstrate the benefits of the hyperbolic language model on downstream mRNA property prediction and antibody region annotation, where it outperforms Euclidean models, and excels in out-of-distribution settings.

2 Related works

RNA and mRNA Language Models RNA and mRNA language models now enable diverse downstream tasks in property prediction, annotation, and generation. RNA foundation models like RNA-FM [14] and RiNALMo [59] pre-train on millions of sequences from varied RNA regions to learn generalist representations. Specialized models such as SpliceBERT [15], UTRBERT [77], and UTR-LM [17] target specific sequence regions for tasks like splicing or UTR analysis. For mRNA, codon-level models such as CodonBERT [39] use codon tokenization with MLM to optimize coding-region embeddings, while Helix-mRNA [75] employs nucleotide level tokenization and hybrid attention-state space architectures for improved sequence resolution and generation. Transfer learning from DNA and protein models to mRNA has also proven effective [60, 49]. Most recent works incorporate domain priors: Equi-mRNA [79] encodes codon symmetries via group-theoretical approach, and HELM [78] promotes hierarchy learning in Euclidean space through specialized loss. Meanwhile, joint geometry-language frameworks have been developed to link RNA sequences with their secondary structures [50]. Despite these advances, existing methods rely on Euclidean spaces; to our knowledge, no prior work explores language-model pre-training in hyperbolic space for RNA or mRNA.

Hyperbolic learning The exponential growth of hyperbolic space makes it a suitable domain for learning on data with an inherent hierarchical structure [64, 11, 53]. This realization has led to a surge in the popularity of hyperbolic learning [58]. Deep hyperbolic architectures have been developed [25, 66, 16] alongside the algorithms for optimizing such networks [3, 2]. As a result, hyperbolic geometry has seen successful applications across many areas of machine learning, such as in computer vision [37, 42, 43, 28, 71, 48], graph learning [41, 12, 82, 76], NLP [69, 21] and multimodal learning [20, 56]. These have shown the potential of hyperbolic learning, particularly in scenarios where the data has a clear hierarchical structure. While the structuring of mRNA is highly hierarchical in nature, existing mRNA language modeling approaches do not leverage hyperbolic geometry.

Prototype learning The prototype learning setting [67] has become a commonly used approach for classification tasks, where each class is represented by a prototype, resembling in some way the perfect instance of the class. Within hyperbolic learning, prototype learning approaches are mostly distinguishable by their method of obtaining prototypes [48]. Many works follow the original approach for generating prototypes based on labeled input data [37, 26, 27, 30]. Theses typically create prototypes by aggregating features of labeled instances of the corresponding class using, for example, the Fréchet mean. Another approach is to use prior knowledge of the label set to generate prototypes. Examples are [28] and [43], which create prototypes using a known hierarchy over the labels, or [80], which optimizes prototypes concurrently with their model through to use of known hierarchical relations. While each of these works deals with an image classification setting, we instead focus on masked language modeling. Moreover, unlike our work, none of these works explores the use of similarity functions beyond hyperbolic distances, nor the use of recent low-distortion embedding methods for generating prototypes from hierarchies.

3 Background on hyperbolic space

In this paper we make use of the $n-$dimensional Poincaré ball model $(\mathbb{D}_c^n, \mathfrak{g})$ of hyperbolic space with curvature c and Riemannian metric $\mathfrak{g}$, where

$$\mathbb{D}_c^n = \{\mathbf{x} \in \mathbb{R}^n : ||\mathbf{x}||^2 < 1\}, \quad \mathfrak{g}_c^n = \lambda_{\mathbf{x}}^c I_n, \quad \lambda_{\mathbf{x}}^c = \frac{2}{1 - c||\mathbf{x}||^2}, \tag{1}$$

with I_n being the n-dimensional identity matrix. For an extensive background on other isometric models and on hyperbolic geometry in general, we refer the reader to [8, 1]. Here, we will introduce the operations that will be used throughout the paper.

Using the Riemannian metric, one can compute the distances between any two points $\mathbf{x}, \mathbf{y} \in \mathbb{D}^n$ as

$$d_{\mathbb{D}}^c(\mathbf{x}, \mathbf{y}) = \frac{1}{\sqrt{c}} \cosh^{-1}\left(1 + 2c\frac{||\mathbf{x} - \mathbf{y}||^2}{(1 - c||\mathbf{x}||^2)(1 - c||\mathbf{y}||^2)}\right). \tag{2}$$

Using the Möbius addition operation [70], defined as

$$\mathbf{x} \oplus_c \mathbf{y} = \frac{(1 + 2c\langle \mathbf{x}, \mathbf{y}\rangle + c||\mathbf{y}||^2)\mathbf{x} + (1 - c||\mathbf{x}||^2)\mathbf{y}}{1 + 2c\langle \mathbf{x}, \mathbf{y}\rangle + c^2||\mathbf{x}||^2||\mathbf{y}||^2}, \tag{3}$$

we can define exponential and logarithmic maps [25]

$$\exp_{\mathbf{x}}^c(\mathbf{v}) = \mathbf{x} \oplus_c \left(\tanh\left(\frac{\sqrt{c}\lambda_{\mathbf{x}}^c||\mathbf{v}||}{2}\right)\frac{\mathbf{v}}{\sqrt{c}||\mathbf{v}||}\right), \tag{4}$$

$$\log_{\mathbf{x}}^c(\mathbf{y}) = \frac{2}{\sqrt{c}\lambda_{\mathbf{x}}^c} \tanh^{-1}\left(\sqrt{c}||-\mathbf{x} \oplus_c \mathbf{y}||\right)\frac{-\mathbf{x} \oplus_c \mathbf{y}}{||-\mathbf{x} \oplus_c \mathbf{y}||}, \tag{5}$$

which are used to map tangent vectors from the tangent space $\mathcal{T}_{\mathbf{x}}\mathbb{D}_c^n$ at $\mathbf{x}$ onto $\mathbb{D}_c^n$ and vice versa, respectively.

[25] have generalized multinomial logistic regression (MLR) to the Poincaré ball model by interpreting the MLR scores as signed distances to hyperplanes. The resulting hyperbolic MLR computes scores as

$$\ell_k(\mathbf{x}) = \frac{2}{\sqrt{c}}||\mathbf{z}_k|| \sinh^{-1}\left(\lambda_{\mathbf{x}}^c\left\langle\sqrt{c}\mathbf{x}, \frac{\mathbf{z}_k}{||\mathbf{z}_k||}\right\rangle \cosh(2\sqrt{c}r_k) - (\lambda_{\mathbf{x}}^c - 1)\sinh(2\sqrt{c}r_k)\right), \tag{6}$$

where $\mathbf{z}_k$ and r_k are the parameters corresponding to the k-th class. This MLR has been further extended into a hyperbolic fully connected layer $\mathcal{F}^c : \mathbb{D}_c^n \to \mathbb{D}_c^m$ by [66], which is computed as

$$\mathcal{F}^c(\mathbf{x}; \mathbf{Z}, \mathbf{r}) = \frac{\mathbf{w}}{1 + \sqrt{1 + c||\mathbf{w}||^2}}, \quad \mathbf{w} = \left(\frac{1}{\sqrt{c}}\sinh\left(\sqrt{c}\ell_k(\mathbf{x})\right)\right)_{k=1}^n, \tag{7}$$

where $\mathbf{Z}$ and $\mathbf{r}$ contain the learnable parameters.

4 HyperHELM

The setting that we consider is the pre-training of an mRNA sequence model through masked language modeling (MLM) with the goal of obtaining a strong backbone for any downstream predictive task. For our approach, we take the HELM method (mRNA model in Euclidean space for modeling hierarchy) [78] as a starting point and make improvements to the classifier to help guide the backbone model more effectively. More specifically, we replace the multinomial logistic regression classifier by a prototypical classifier, inspired by works such as [67, 80]. The prototypes are generated directly from the codon-amino acid hierarchy which is shown in Figure 1 and, more clearly, in Figure 3 in Appendix A. A high-level overview of our method is given in Figure 1. Each individual component will be discussed in detail in the following subsections.

4.1 Language Modeling of mRNA Sequences

Our goal is to train some sequence transformer model f of mRNA sequences through MLM. Following recent works [39, 78], we first apply codon-level tokenization to the mRNA sequences, where each triplet of nucleotides is represented as a single token, giving $4^3 = 64$ potential tokens, excluding special tokens. During MLM, we mask 15% of the tokens in sequences and feed these into model f, which outputs a representation in $\mathbb{R}^n$ for each individual token. Then, we use a classifier $g : \mathbb{R}^n \to [64]$ to predict the true label of the masked tokens. Following the HELM approach [78], the hierarchical cross-entropy loss with respect to the codon hierarchy in Figure 1 is computed and used to update f and g.

4.2 Hyperbolic embeddings of hierarchies

The manner in which mRNA encodes for proteins can be understood through a hierarchy defined over the codons, visualized in Figure 1. [78] softly enforces this hierarchy onto their model in Euclidean space by using the hierarchical cross-entropy loss. Here, we explicitly structure our token representation space by directly embedding the hierarchy. A hierarchy typically consists of a tree $T = (V, E)$, where the nodes V contain the relevant concepts and the edges E the relations between these. Moreover, we denote the leaf nodes of the tree by L. The tree metric d_T, resulting from T, defined as the length of the path between 2 nodes, contains the information of how strongly related any pair of concepts is. Therefore, the goal of embedding some hierarchy into a continuous space is to keep this tree metric intact. More formally, we want an embedding $\phi : V \to M$ into some connected Riemannian manifold M such that ϕ is approximately an isometry onto $\phi(V)$, i.e.,

$$d_M\big(\phi(u), \phi(v)\big) \approx d_T(u, v). \tag{8}$$

The amount by which the metric is changed by the embedding is called the distortion. It can be shown that Euclidean spaces are unsuitable as targets for embedding trees [64], generally leading to highly distorted embeddings. Therefore, we opt to use hyperbolic space instead.

Several methods exist for embedding graphs or trees into hyperbolic space [64, 53, 63, 72]. We embed the codon hierarchy using the HS-DTE method [72], as it achieves the lowest distortion and thus most effectively preserves the underlying hierarchical structure. We use the embeddings of the leaf nodes obtained with HS-DTE, corresponding to individual codons, as prototypes within the classifier g. A 2-dimensional embedding of the entire codon hierarchy obtained with HS-DTE is shown in Figure 1.

4.3 Prototype learning in hyperbolic space

From the hierarchy embedding, we have a set of prototypes $\phi(L) \subset \mathbb{D}^{n_p}$ where each prototype corresponds to a particular codon and where n_p is the prototype dimension. Since the embedding ϕ respects the tree metric d_T, these prototypes structure the space according to the hierarchy, without having seen any sequence data. We want to define a classifier that uses these prototypes to generate token-level predictions. Since our backbone model f outputs representations in $\mathbb{R}^n$, these are first projected onto $\mathbb{D}^{n_p}$ through two steps: 1) the representations are projected into hyperbolic space $\mathbb{D}^n$ and 2) A hyperbolic linear layer is used to project to $\mathbb{D}^{n_p}$. Following the convention in hyperbolic learning [48], the first step is performed by treating the representations as tangent vectors at the origin and applying the corresponding exponential map. The second step is performed using the hyperbolic linear layer $\mathcal{F}^c : \mathbb{D}^n \to \mathbb{D}^{n_p}$ from equation 7. So, the projection can be written as

$$\mathbf{z}_i = \mathcal{F}^c\big(\exp_{\mathbf{0}}^c(\mathbf{h}_i)\big), \quad \mathbf{h}_i = f(\mathbf{t}^*)_i, \tag{9}$$

where $\mathbf{t}^*$ is the masked token sequence.

Generally, to generate token-level predictions using prototypes, softmaxed pairwise similarities between representations and prototypes are computed [67]:

$$p(t_i = u|\mathbf{t}^*) = \frac{\exp\big(\beta \cdot s(\mathbf{z}_i, \phi(u))\big)}{\sum_{v \in L} \exp\big(\beta \cdot s(\mathbf{z}_i, \phi(v))\big)}, \tag{10}$$

where $\beta > 0$ is a temperature hyperparameter (set to 1.0), t_i is the true i-th token and where $s : \mathbb{D}^{n_p} \times \mathbb{D}^{n_p} \to \mathbb{R}$ is some similarity function. Typically, negative distances $s = -d_{\mathbb{D}}$ are used as

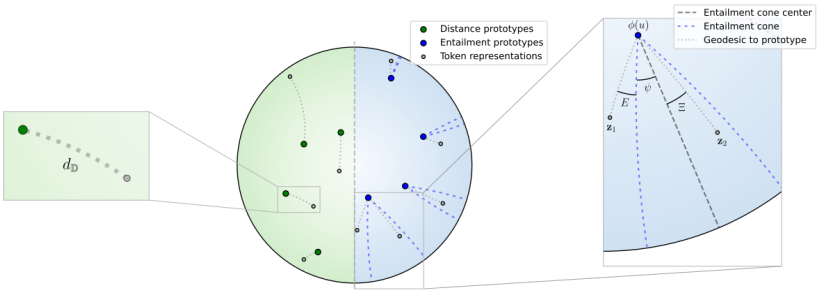

Figure 2: **Hyperbolic prototype learning.** The *center* part presents a Poincaré disk where either distances (green) or entailment cone energies (blue) are used to predict the label of embedded tokens. On the *left*, a close up of a masked token representation with its closest prototype, together with the geodesic between these is shown. The *right* part takes a closer look at one of the entailment cones, showing the geometric interpretation of equations 11, 12 and 13.

similarities, which leads the model to simply assign the token to the closest prototype. This approach is shown in Figure 2 *left*.

Alternatively, we can compute similarities using the hyperbolic entailment cone energy [24]. Entailment cones are a geometric approach to defining hierarchical relationships in hyperbolic space. These are defined for any point $\mathbf{z} \in \mathbb{D}^{n_p}$ as the hyperbolic cone with $\mathbf{z}$ as its apex and with the axis of symmetry being the Euclidean straight line segment from $\mathbf{z}$ perpendicular onto the boundary of the manifold. The half aperture of the cone is

$$\psi(\mathbf{z}) = \sin^{-1}\left(\frac{K(1-||\mathbf{z}||^2)}{||\mathbf{z}||}\right), \tag{11}$$

where K is a hyperparameter which we set to $K = 0.1$. The hyperbolic entailment cone energy is then computed as

$$E(\mathbf{x}, \mathbf{y}) = \max(0, \Xi(\mathbf{x}, \mathbf{y}) - \eta\psi(\mathbf{x})), \tag{12}$$

where $\eta > 0$ is a threshold hyperparameter [56] (set to 1.05) and where

$$\Xi(\mathbf{x}, \mathbf{y}) = \cos^{-1}\left(\frac{\langle \mathbf{x}, \mathbf{y}\rangle(1+||\mathbf{x}||^2) - ||\mathbf{x}||^2(1+||\mathbf{y}||^2)}{||\mathbf{x}|| \cdot ||\mathbf{x}-\mathbf{y}||\sqrt{1+||\mathbf{x}||^2||\mathbf{y}||^2 - 2\langle x, y\rangle}}\right), \tag{13}$$

is the aperture required for $\mathbf{y}$ to be within the entailment cone at $\mathbf{x}$. In other words, the hyperbolic entailment cone energy is the angle by which $\mathbf{y}$ is removed from $\mathbf{x}$'s entailment cone. Examples of entailment cones and a visualization of the entailment cone energy are shown in Figure 2 *right*. The hyperbolic entailment cone energy has recently grown in popularity in areas such as vision-language learning [20, 56] for encoding hierarchical relations. We propose to use both distance-based prototypes and energy-based prototypes. For both approaches, we set curvature $c = 1.0$. We also present a sensitivity analysis for the key hyperparameters in Appendix D.

5 Experiments

In our experiments, we follow the pre-training guidelines established in HELM [78], adopting codon-level tokenization and the masked language modeling (MLM) objective. We use the same curated OAS pre-training corpus, codon vocabulary, and standard transformer backbone released in the official HELM repository [3], ensuring full comparability. The key difference lies in the MLM head: we evaluate three hyperbolic variants: hyperbolic multinomial logistic regression, hyperbolic distance-based prototypes, and hyperbolic prototypes based on entailment cones discussed in Sections 3 and 4. We keep the rest of the method unchanged, allowing us to isolate the effect of learning the hierarchy in hyperbolic space for mRNAs. For downstream tasks, we freeze the pre-trained backbone and train a classification or regression head to perform probing experiments with pre-trained language

[3] https://github.com/johnsonandjohnson/HELM

model representations. Following prior works [31, 39, 78, 49, 79] to assess representation quality and transferability, we utilize TextCNN [38] as a downstream head model. Further experimental details can be found in Appendices B and C.

Datasets and evaluation metrics We use nine datasets: Ab1 [60] (1,200 antibody-encoding mRNAs with protein expression labels); Ab2 [60] (3,442 antibody-encoding mRNAs from a different experimental platform with protein expression labels); mRFP [54] (1,459 sequences with protein production levels); COVID-19 Vaccine [73] (2,400 degradation-labeled sequences); *Drosophila melanogaster* [55] (10,338 sequences with protein abundance); *Saccharomyces cerevisiae* [55] (4,937 sequences with protein abundance); *Pichia pastoris* [55] (4,682 sequences with transcript abundance labels); Fungal [74] (7,056 genes from fungal genomes with expression labels); and *E. coli* [22] (6,348 mRNA with low/medium/high protein expression labels). The *E.Coli* dataset is a classification task while all other downstream datasets provide regression labels for evaluating the quality of mRNA property prediction.

Similar to prior works [78, 39, 79, 75], we evaluate mRNA property prediction using Spearman rank correlation for regression tasks and accuracy for classification tasks to assess the quality and transferability of learned representations. For the *Drosophila melanogaster*, *Saccharomyces cerevisiae*, and *Pichia pastoris* datasets, no predefined train/validation/test splits are available, so we generate random splits. For all other datasets, we follow the pre-defined splits used in prior works [78, 39].

Baselines We present the performance of HyperHELM variants against a non-hierarchical Transformer trained with cross-entropy loss (Transformer XE) and hierarchy-aware Euclidean transformer HELM [32], which has recently been reported to achieve state-of-the-art results for mRNA property prediction in multiple studies [78, 75] when compared to other publicly available RNA language models. Note that our HyperHELM and all baselines are pre-trained on the same dataset and use the same backbone architecture with the same number of parameters, thus any performance difference is attributed only to the methodology and the impact of hyperbolic learning.

5.1 HyperHELM improves downstream mRNA property prediction performance over Euclidean models

Table 1 summarizes the performance of HyperHELM variants across 9 mRNA property prediction datasets. On 8 out of 9 datasets, HyperHELM models outperform their Euclidean counterparts, demonstrating the benefits of modeling hierarchical relationships in hyperbolic spaces for mRNA sequences. Of these, HyperHELM with distance-based prototypes (Proto Dist) and HyperHELM with entailment cones-based prototypes (Proto Entailment) achieve the best and second-best performance on 7 out of 9 datasets. Compared to the non-hierarchical Transformer XE baseline, HyperHELM improves downstream performance by 2.8–35.5%, with the largest gains observed for D. melanogaster (35.5%) and S. cerevisiae (31.4%). When compared to HELM, performance improvements range up to 32%, with particularly strong improvements on D. melanogaster (32.0%) and *E. coli* (10.9%) datasets. Interestingly, simple hyperbolic MLR (HyperHELM MLR) only performs best on a single S.cerevisiae dataset while underperforming on all other tasks relative to even the Euclidean models, indicating that the combination of hyperbolic geometry with prototype-based heads is crucial for capturing hierarchical structure in mRNA embeddings.

Table 1: Spearman rank correlation and accuracy (for *E.coli.*) performance of HyperHELM variants compared to standard and hiearachy-aware Euclidean Transformer XE and HELM models. Bold indicates the best performing model per dataset and underline indicates second best model.

Dataset	Transformer XE	HELM	HyperHELM MLR	HyperHELM (Proto Dist)	HyperHELM (Proto Entailment)
Ab1	0.701	<u>0.714</u>	0.650	0.713	**0.751**
Ab2	0.507	0.548	0.532	**0.575**	<u>0.569</u>
mRFP	<u>0.825</u>	**0.848**	0.744	0.819	0.802
COVID-19	0.757	0.775	0.411	<u>0.785</u>	**0.807**
D. melanogaster	0.332	0.341	0.374	<u>0.394</u>	**0.450**
S. cerevisiae	0.354	0.398	**0.465**	<u>0.434</u>	0.397
P. pastoris	0.596	0.620	0.605	**0.676**	<u>0.671</u>
Fungal	0.690	0.702	0.712	<u>0.735</u>	**0.741**
E. coli	44.7	45.8	40.0	**50.8**	<u>48.4</u>

5.2 HyperHELM improves Antibody Sequence Annotation

We further assess HyperHELM on the task of antibody (Ab) sequence region annotation, a benchmark introduced in prior work [78], important for immunological studies [4]. This task involves predicting the identity of nucleotides in Ab-coding mRNA into one of four biologically meaningful regions: signal peptides, V, DJ, or constant regions.

We use the same held-out test set of 2000 curated antibody sequences as used in [78] for this task and compare our HyperHELM against the HELM model. As shown in Table 2(a), both prototype- and cones-based HyperHELM variants outperform Euclidean HELM, with the prototype distance model achieving the best accuracy of 76.48%, and the prototype entailment variant being second best with accuracy of 75.21%, compared to 73.48% achieved by HELM. The results highlight the advantage of hierarchy-aware learning in hyperbolic space to effectively capture the structure of antibody mRNA regions.

5.3 Impact of Sequence Length and GC Content on Model Performance

We examine model robustness across different bologically meaningful mRNA sequence characteristics by stratifying datasets according to sequence length and GC content. The motivation for these strata is twofold: (i) sequence length can strongly influence modeling difficulty, as longer sequences often contain more complex structural dependencies and are often underrepresented in pre-training corpora; and (ii) GC content variation significantly affects secondary structure formation, with extremely high or low GC content posing additional modeling challenges [65, 57]. These factors are biologically relevant for mRNA engineering [19, 81, 35] and have been linked to differences in model generalization [9, 61, 68].

Sequence Length Analysis For the `P. pastoris` dataset, we divide sequences into three categories: *short* (30–1000 nucleotides), *medium* (1000–2000 nucleotides), and *long* (2000–3000 nucleotides). Typical mRNA vaccine sequences fall in the 1000–1500 nucleotide range [29]. Our pre-training dataset includes sequences with a maximum length of approximately 1400 nucleotides. The sequences longer than this are hence not represented during pre-training and present an out-of-distribution setting for evaluating the pre-trained models.

As shown in Table 2(b), Euclidean HELM's performance decreases with increasing length, particularly for long sequences, consistent with prior findings that longer sequences are less effectively captured by models pre-trained on shorter examples [78]. However, *both* HyperHELM variants not only mitigate this drop but actually reverse it: performance *increases* for the long category compared to medium-length sequences. The entailment-based variant achieves the highest score, reaching a Spearman correlation of 0.70—a $+0.24$ absolute improvement over HELM, while the distance-based variant also attains a substantial improvement of $+0.19$. This indicates that HyperHELM's hyperbolic-space representation is beneficial even for out-of-distribution length shifts, a trend also reported for hyperbolic models in other domains [34, 36].

GC Content Analysis For the COVID dataset, we categorize sequences based on GC content into: *low* (GC $\leq 47\%$), *medium* ($47\% <$ GC $\leq 55\%$), and *high* (GC $> 55\%$). These thresholds align with widely used biological definitions, where GC content below 45% is considered low and above 56% is high [6, 19].

Performance for both HELM and HyperHELM (shown in Table 2(c)) is reasonably high in the low GC range but diminishes for high GC content sequences due to their relative scarcity in the pre-training corpora. Notably, the entailment-based HyperHELM attains a Spearman rank correlation of 0.62 in the high GC category compared to HELM's 0.56, and achieves a strong Spearman rank correlation of 0.73 in the medium GC category, a gain of $+0.09$ over HELM.

Overall, these results demonstrate that HyperHELM maintains strong performance across typical conditions while providing notable improvements in more extreme sequence regimes, longer lengths, and higher GC content, where conventional HELM shows more pronounced performance degradation.

Table 2: (a) Accuracy of antibody sequence region annotation. (b) Performance reported as Spearman rank correlation across sequence length for `P. pastoris`. (c) Performance reported as Spearman rank correlation across GC content for the COVID-19 dataset.

Model	**Acc. (%)**
HELM	73.48
HyperHELM (Dist.)	**76.48**
HyperHELM (Entail.)	<u>75.21</u>

(a) Antibody annotation

	Short	Med.	Long
HELM	0.54	0.58	0.46
HyperHELM (Dist.)	0.65	**0.59**	**0.65**
HyperHELM (Entail.)	**0.61**	0.56	0.70

(b) Seq. length analysis

	Low	Med.	High
HELM	**0.78**	0.64	0.56
HyperHELM (Dist.)	0.77	0.62	0.54
HyperHELM (Entail.)	**0.78**	**0.73**	**0.62**

(c) GC content analysis

6 Discussion

Our findings demonstrate that aligning model geometry with the inherent hierarchical structure of RNA sequences provides tangible benefits for representation learning. Hyperbolic embeddings not only improve downstream property prediction but also offer a more faithful reflection of codon-amino-acid relationships, particularly in sequences with strong codon usage bias. This suggests that geometry-aware modeling can mitigate challenges arising from imbalanced sequence distributions and enhance generalization to out-of-distribution sequences.

The observed improvements highlight the potential of hybrid architectures, where Euclidean backbones are paired with hyperbolic heads, as a practical strategy to integrate hierarchical inductive biases without incurring the computational overhead of fully hyperbolic networks. Moreover, the success of hyperbolic entailment cones and prototype-based methods indicates that explicitly modeling hierarchical relationships can be more effective than standard Euclidean hierarchy-aware approaches.

Limitations and Future Work In this work, we assume fixed prototypes for both of our HyperHELM variants-distance based and entailment cones based, and it will be worth exploring mechanisms to update these prototypes during the learning process itself in the future. Furthermore, extending HyperHELM for Causal Language Modeling will enable generative applications of the proposed techniques in future. Also our results open several avenues for future work. Extending hyperbolic language models to other biological modalities, such as protein sequences or regulatory genomic regions, could further exploit hierarchical patterns across scales. Additionally, investigating adaptive curvature or mixed-geometry latent spaces may enhance the flexibility of representations for diverse hierarchical structures. Overall, our study underscores the importance of considering latent space geometry when designing models for complex biological sequence data.

References

[1] James W Anderson. *Hyperbolic geometry*. Springer Science & Business Media, 2006.

[2] Gary Bécigneul and Octavian-Eugen Ganea. Riemannian adaptive optimization methods. *arXiv preprint arXiv:1810.00760*, 2018.

[3] Silvere Bonnabel. Stochastic gradient descent on riemannian manifolds. *IEEE Transactions on Automatic Control*, 58(9):2217–2229, 2013.

[4] Bryan Briney and Dennis R Burton. Massively scalable genetic analysis of antibody repertoires. *BioRxiv*, page 447813, 2018.

[5] Garyk Brixi, Matthew G Durrant, Jerome Ku, Michael Poli, Greg Brockman, Daniel Chang, Gabriel A Gonzalez, Samuel H King, David B Li, Aditi T Merchant, Mohsen Naghipourfar, Eric Nguyen, Chiara Ricci-Tam, David W Romero, Gwanggyu Sun, Ali Taghibakshi, Anton Vorontsov, Brandon Yang, Myra Deng, Liv Gorton, Nam Nguyen, Nicholas K Wang, Etowah Adams, Stephen A Baccus, Steven Dillmann, Stefano Ermon, Daniel Guo, Rajesh Ilango, Ken Janik, Amy X Lu, Reshma Mehta, Mohammad R.K. Mofrad, Madelena Y Ng, Jaspreet Pannu, Christopher Re, Jonathan C Schmok, John St. John, Jeremy Sullivan, Kevin Zhu, Greg Zynda, Daniel Balsam, Patrick Collison, Anthony B. Costa, Tina Hernandez-Boussard, Eric Ho, Ming-Yu Liu, Tom McGrath, Kimberly Powell, Dave P. Burke, Hani Goodarzi, Patrick D Hsu, and Brian Hie. Genome modeling and design across all domains of life with evo 2. *bioRxiv*, 2025.

[6] J. Brown. High g+c content of herpes simplex virus dna: Proposed role in protection against retrotransposon insertion. *Open Biochem J*, 1:33–42, 2007.

[7] Florian Buhr, Sujata Jha, Michael Thommen, Joerg Mittelstaet, Felicitas Kutz, Harald Schwalbe, Marina V Rodnina, and Anton A Komar. Synonymous codons direct cotranslational folding toward different protein conformations. *Molecular cell*, 61(3):341–351, 2016.

[8] James W Cannon, William J Floyd, Richard Kenyon, Walter R Parry, et al. Hyperbolic geometry. *Flavors of geometry*, 31(59-115):2, 1997.

[9] Sebastian M Castillo-Hair and Georg Seelig. Machine learning for designing next-generation mrna therapeutics. *Accounts of chemical research*, 55(1):24–34, 2021.

[10] Albi Celaj, Alice Jiexin Gao, Tammy TY Lau, Erle M Holgersen, Alston Lo, Varun Lodaya, Christopher B Cole, Robert E Denroche, Carl Spickett, Omar Wagih, et al. An rna foundation model enables discovery of disease mechanisms and candidate therapeutics. *bioRxiv*, pages 2023–09, 2023.

[11] Benjamin Paul Chamberlain, James Clough, and Marc Peter Deisenroth. Neural embeddings of graphs in hyperbolic space. *arXiv preprint arXiv:1705.10359*, 2017.

[12] Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. Hyperbolic graph convolutional neural networks. *Advances in neural information processing systems*, 32, 2019.

[13] Bo Chen, Xingyi Cheng, Pan Li, Yangli-ao Geng, Jing Gong, Shen Li, Zhilei Bei, Xu Tan, Boyan Wang, Xin Zeng, et al. xtrimopglm: unified 100b-scale pre-trained transformer for deciphering the language of protein. *arXiv preprint arXiv:2401.06199*, 2024.

[14] Jiayang Chen, Zhihang Hu, Siqi Sun, Qingxiong Tan, Yixuan Wang, Qinze Yu, Licheng Zong, Liang Hong, Jin Xiao, Tao Shen, et al. Interpretable rna foundation model from unannotated data for highly accurate rna structure and function predictions. *arXiv preprint arXiv:2204.00300*, 2022.

[15] Ken Chen, Yue Zhou, Maolin Ding, Yu Wang, Zhixiang Ren, and Yuedong Yang. Self-supervised learning on millions of pre-mrna sequences improves sequence-based rna splicing prediction. *bioRxiv*, pages 2023–01, 2023.

[16] Weize Chen, Xu Han, Yankai Lin, Hexu Zhao, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. Fully hyperbolic neural networks. *arXiv preprint arXiv:2105.14686*, 2021.

[17] Yanyi Chu, Dan Yu, Yupeng Li, Kaixuan Huang, Yue Shen, Le Cong, Jason Zhang, and Mengdi Wang. A 5' utr language model for decoding untranslated regions of mrna and function predictions. *Nature Machine Intelligence*, 6(4):449–460, 2024.

[18] Suzanne Clancy and William Brown. Translation: Dna to mrna to protein. *Nature Education*, 1(1):101, 2008.

[19] Maïté Courel, Yves Clément, Clémentine Bossevain, Dominika Foretek, Olivia Vidal Cruchez, Zhou Yi, Marianne Bénard, Marie-Noelle Benassy, Michel Kress, Caroline Vindry, et al. Gc content shapes mrna storage and decay in human cells. *elife*, 8:e49708, 2019.

[20] Karan Desai, Maximilian Nickel, Tanmay Rajpurohit, Justin Johnson, and Shanmukha Ramakrishna Vedantam. Hyperbolic image-text representations. In *International Conference on Machine Learning*, pages 7694–7731. PMLR, 2023.

[21] Bhuwan Dhingra, Christopher J Shallue, Mohammad Norouzi, Andrew M Dai, and George E Dahl. Embedding text in hyperbolic spaces. *arXiv preprint arXiv:1806.04313*, 2018.

[22] Zundan Ding, Feifei Guan, Guoshun Xu, Yuchen Wang, Yaru Yan, Wei Zhang, Ningfeng Wu, Bin Yao, Huoqing Huang, Tamir Tuller, et al. Mpepe, a predictive approach to improve protein expression in e. coli based on deep learning. *Computational and Structural Biotechnology Journal*, 20:1142–1153, 2022.

[23] Noelia Ferruz, Steffen Schmidt, and Birte Höcker. Protgpt2 is a deep unsupervised language model for protein design. *Nature communications*, 13(1):4348, 2022.

[24] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic entailment cones for learning hierarchical embeddings. In *International conference on machine learning*, pages 1646–1655. PMLR, 2018.

[25] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic neural networks. *Advances in neural information processing systems*, 31, 2018.

[26] Zhi Gao, Yuwei Wu, Yunde Jia, and Mehrtash Harandi. Curvature generation in curved spaces for few-shot learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8691–8700, 2021.

[27] Zhi Gao, Yuwei Wu, Yunde Jia, and Mehrtash Harandi. Hyperbolic feature augmentation via distribution estimation and infinite sampling on manifolds. *Advances in neural information processing systems*, 35:34421–34435, 2022.

[28] Mina Ghadimi Atigh, Martin Keller-Ressel, and Pascal Mettes. Hyperbolic busemann learning with ideal prototypes. *Advances in neural information processing systems*, 34:103–115, 2021.

[29] Helen M Gunter, Senel Idrisoglu, Swati Singh, Dae Jong Han, Emily Ariens, Jonathan R Peters, Ted Wong, Seth W Cheetham, Jun Xu, Subash Kumar Rai, et al. mrna vaccine quality analysis using rna sequencing. *Nature Communications*, 14(1):5663, 2023.

[30] Yunhui Guo, Xudong Wang, Yubei Chen, and Stella X Yu. Clipped hyperbolic classifiers are super-hyperbolic classifiers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11–20, 2022.

[31] Ameya Harmalkar, Roshan Rao, Yuxuan Richard Xie, Jonas Honer, Wibke Deisting, Jonas Anlahr, Anja Hoenig, Julia Czwikla, Eva Sienz-Widmann, Doris Rau, et al. Toward generalizable prediction of antibody thermostability using machine learning on sequence and structure features. In *Mabs*, volume 15, page 2163584. Taylor & Francis, 2023.

[32] Neil He, Rishabh Anand, Hiren Madhu, Ali Maatouk, Smita Krishnaswamy, Leandros Tassiulas, Menglin Yang, and Rex Ying. Helm: Hyperbolic large language models via mixture-of-curvature experts. *arXiv preprint arXiv:2505.24722*, 2025.

[33] Yuan He, Moy Yuan, Jiaoyan Chen, and Ian Horrocks. Language models as hierarchy encoders. *Advances in Neural Information Processing Systems*, 37:14690–14711, 2024.

[34] Sarah Ibrahimi, Mina Ghadimi Atigh, Nanne Van Noord, Pascal Mettes, and Marcel Worring. Intriguing properties of hyperbolic embeddings in vision-language models. *Transactions on Machine Learning Research*, 2024.

[35] Longfei Jia and Shu-Bing Qian. Therapeutic mrna engineering from head to tail. *Accounts of Chemical Research*, 54(23):4272–4282, 2021.

[36] Tejaswi Kasarla, Max van Spengler, and Pascal Mettes. Balanced hyperbolic embeddings are natural out-of-distribution detectors. *arXiv preprint arXiv:2506.10146*, 2025.

[37] Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. Hyperbolic image embeddings. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6418–6428, 2020.

[38] Yoon Kim. Convolutional neural networks for sentence classification. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October 2014. Association for Computational Linguistics.

[39] Sizhen Li, Saeed Moayedpour, Ruijiang Li, Michael Bailey, Saleh Riahi, Lorenzo Kogler-Anele, Milad Miladi, Jacob Miner, Dinghai Zheng, Jun Wang, et al. Codonbert: Large language models for mrna design and optimization. *bioRxiv*, pages 2023–09, 2023.

[40] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

[41] Qi Liu, Maximilian Nickel, and Douwe Kiela. Hyperbolic graph neural networks. *Advances in neural information processing systems*, 32, 2019.

[42] Shaoteng Liu, Jingjing Chen, Liangming Pan, Chong-Wah Ngo, Tat-Seng Chua, and Yu-Gang Jiang. Hyperbolic visual embedding learning for zero-shot recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9273–9281, 2020.

[43] Teng Long, Pascal Mettes, Heng Tao Shen, and Cees GM Snoek. Searching for actions on the hyperbole. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1141–1150, 2020.

[44] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[45] Céline Marquet, Michael Heinzinger, Tobias Olenyi, Christian Dallago, Kyra Erckert, Michael Bernhofer, Dmitrii Nechaev, and Burkhard Rost. Embeddings from protein language models predict conservation and variant effects. *Human genetics*, 141(10):1629–1647, 2022.

[46] Jiří Matoušek. On the distortion required for embedding finite metric spaces into normed spaces. *Israel Journal of Mathematics*, 93(1):333–344, 1996.

[47] Jiří Matoušek. On embedding trees into uniformly convex banach spaces. *Israel Journal of Mathematics*, 114(1):221–237, 1999.

[48] Pascal Mettes, Mina Ghadimi Atigh, Martin Keller-Ressel, Jeffrey Gu, and Serena Yeung. Hyperbolic deep learning in computer vision: A survey. *International Journal of Computer Vision*, 132(9):3484–3508, 2024.

[49] Amina Mollaysa, Artem Moskale, Pushpak Pati, Tommaso Mansi, Mangal Prakash, and Rui Liao. Biolangfusion: Multimodal fusion of dna, mrna, and protein language models. *arXiv preprint arXiv:2506.08936*, 2025.

[50] Artem Moskalev, Mangal Prakash, Rui Liao, and Tommaso Mansi. Se(3)-hyena operator for scalable equivariant learning, 2024.

[51] Eric Nguyen, Michael Poli, Matthew G Durrant, Armin W Thomas, Brian Kang, Jeremy Sullivan, Madelena Y Ng, Ashley Lewis, Aman Patel, Aaron Lou, et al. Sequence modeling and design from molecular to genome scale with evo. *BioRxiv*, pages 2024–02, 2024.

[52] Eric Nguyen, Michael Poli, Marjan Faizi, Armin Thomas, Michael Wornow, Callum Birch-Sykes, Stefano Massaroli, Aman Patel, Clayton Rabideau, Yoshua Bengio, et al. Hyenadna: Long-range genomic sequence modeling at single nucleotide resolution. *Advances in neural information processing systems*, 36, 2024.

[53] Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. *Advances in neural information processing systems*, 30, 2017.

[54] Thijs Nieuwkoop, Barbara R Terlouw, Katherine G Stevens, Richard A Scheltema, Dick De Ridder, John Van der Oost, and Nico J Claassens. Revealing determinants of translation efficiency via whole-gene codon randomization and machine learning. *Nucleic acids research*, 51(5):2363–2376, 2023.

[55] Carlos Outeiral and Charlotte M Deane. Codon language embeddings provide strong signals for use in protein engineering. *Nature Machine Intelligence*, 6(2):170–179, 2024.

[56] Avik Pal, Max van Spengler, Guido Maria D'Amely di Melendugno, Alessandro Flaborea, Fabio Galasso, and Pascal Mettes. Compositional entailment learning for hyperbolic vision-language models. *arXiv preprint arXiv:2410.06912*, 2024.

[57] Sujatha Thankeswaran Parvathy, Varatharajalu Udayasuriyan, and Vijaipal Bhadana. Codon usage bias. *Molecular biology reports*, 49(1):539–565, 2022.

[58] Wei Peng, Tuomas Varanka, Abdelrahman Mostafa, Henglin Shi, and Guoying Zhao. Hyperbolic deep neural networks: A survey. *IEEE Transactions on pattern analysis and machine intelligence*, 44(12):10023–10044, 2021.

[59] Rafael Josip Penić, Tin Vlašić, Roland G Huber, Yue Wan, and Mile Šikić. Rinalmo: General-purpose rna language models can generalize well on structure prediction tasks. *arXiv preprint arXiv:2403.00043*, 2024.

[60] Mangal Prakash, Artem Moskalev, Peter DiMaggio Jr., Steven Combs, Tommaso Mansi, Justin Scheer, and Rui Liao. Bridging biomolecular modalities for knowledge transfer in bio-language models. In *Neurips 2024 Workshop Foundation Models for Science: Progress, Opportunities, and Challenges*, 2024.

[61] Xiangyun Qiu. Sequence similarity governs generalizability of de novo deep learning models for rna secondary structure prediction. *PLOS Computational Biology*, 19(4):e1011047, 2023.

[62] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[63] Frederic Sala, Chris De Sa, Albert Gu, and Christopher Ré. Representation tradeoffs for hyperbolic embeddings. In *International conference on machine learning*, pages 4460–4469. PMLR, 2018.

[64] Rik Sarkar. Low distortion delaunay embedding of trees in hyperbolic plane. In *International symposium on graph drawing*, pages 355–366. Springer, 2011.

[65] Paul M Sharp and Wen-Hsiung Li. The codon adaptation index-a measure of directional synonymous codon usage bias, and its potential applications. *Nucleic acids research*, 15(3):1281–1295, 1987.

[66] Ryohei Shimizu, Yusuke Mukuta, and Tatsuya Harada. Hyperbolic neural networks++. *arXiv preprint arXiv:2006.08210*, 2020.

[67] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017.

[68] Marcell Szikszai, Michael Wise, Amitava Datta, Max Ward, and David H Mathews. Deep learning models for rna secondary structure prediction (probably) do not generalize across families. *Bioinformatics*, 38(16):3892–3899, 2022.

[69] Alexandru Tifrea, Gary Bécigneul, and Octavian-Eugen Ganea. Poincar\'e glove: Hyperbolic word embeddings. *arXiv preprint arXiv:1810.06546*, 2018.

[70] Abraham Ungar. *A gyrovector space approach to hyperbolic geometry*. Springer Nature, 2022.

[71] Max van Spengler, Erwin Berkhout, and Pascal Mettes. Poincare resnet. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5419–5428, 2023.

[72] Max van Spengler and Pascal Mettes. Low-distortion and gpu-compatible tree embeddings in hyperbolic space. *arXiv preprint arXiv:2502.17130*, 2025.

[73] Hannah K Wayment-Steele, Wipapat Kladwang, Andrew M Watkins, Do Soon Kim, Bojan Tunguz, Walter Reade, Maggie Demkin, Jonathan Romano, Roger Wellington-Oguri, John J Nicol, et al. Deep learning models for predicting rna degradation via dual crowdsourcing. *Nature Machine Intelligence*, 4(12):1174–1184, 2022.

[74] Rhondene Wint, Asaf Salamov, and Igor V Grigoriev. Kingdom-wide analysis of fungal protein-coding and trna genes reveals conserved patterns of adaptive evolution. *Molecular biology and evolution*, 39(2):msab372, 2022.

[75] Matthew Wood, Mathieu Klop, and Maxime Allard. Helix-mrna: A hybrid foundation model for full sequence mrna therapeutics. *arXiv preprint arXiv:2502.13785*, 2025.

[76] Menglin Yang, Min Zhou, Zhihao Li, Jiahong Liu, Lujia Pan, Hui Xiong, and Irwin King. Hyperbolic graph neural networks: A review of methods and applications. *arXiv preprint arXiv:2202.13852*, 2022.

[77] Yuning Yang, Gen Li, Kuan Pang, Wuxinhao Cao, Xiangtao Li, and Zhaolei Zhang. Deciphering 3'utr mediated gene regulation using interpretable deep representation learning. *bioRxiv*, pages 2023–09, 2023.

[78] Mehdi Yazdani-Jahromi, Mangal Prakash, Tommaso Mansi, Artem Moskalev, and Rui Liao. HELM: Hierarchical encoding for mRNA language modeling. In *The Thirteenth International Conference on Learning Representations*, 2025.

[79] Mehdi Yazdani-Jahromi, Ali Khodabandeh Yalabadi, and Ozlem Ozmen Garibay. Equi-mrna: Protein translation equivariant encoding for mrna language models. *arXiv preprint arXiv:2508.15103*, 2025.

[80] Zhen Yu, Toan Nguyen, Yaniv Gal, Lie Ju, Shekhar S Chandra, Lei Zhang, Paul Bonnington, Victoria Mar, Zhiyong Wang, and Zongyuan Ge. Skin lesion recognition with class-hierarchy regularized hyperbolic embeddings. In *International conference on medical image computing and computer-assisted intervention*, pages 594–603. Springer, 2022.

[81] Jing Zhang, CC Jay Kuo, and Liang Chen. Gc content around splice sites affects splicing through pre-mrna secondary structures. *BMC genomics*, 12(1):90, 2011.

[82] Yiding Zhang, Xiao Wang, Chuan Shi, Xunqiang Jiang, and Yanfang Ye. Hyperbolic graph attention network. *IEEE Transactions on Big Data*, 8(6):1690–1701, 2021.

[83] Zhihan Zhou, Yanrong Ji, Weijian Li, Pratik Dutta, Ramana Davuluri, and Han Liu. Dnabert-2: Efficient foundation model and benchmark for multi-species genome. *arXiv preprint arXiv:2306.15006*, 2023.

A Hierarchical relationship of codons and amino acids in mRNA

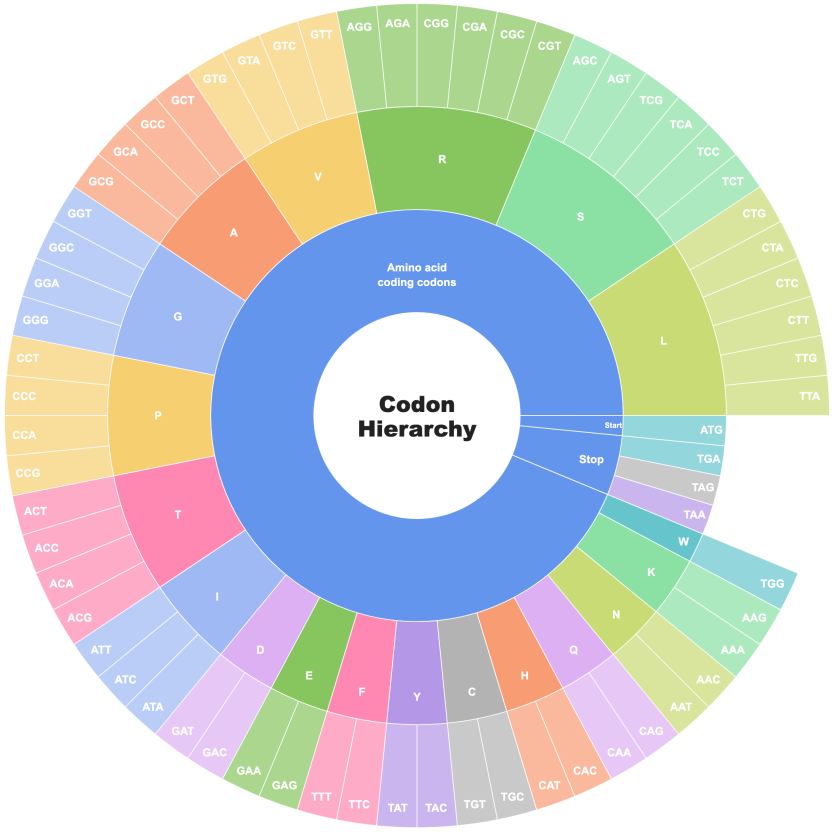

Figure 3: The codon hierarchy that is used for creating prototypes and structuring the representation space.

B Pre-training details

All our experiments were run with a transformer backbone, consisting of 10 transformer layers with an intermediate size of 2560 and a hidden size of 640, resulting in a total of $\sim$50M parameters. All models were pretrained for 40 epochs with a batch size of 1024 spread across 8 Nvidia A100 GPUs using the hierarchical cross-entropy (HXE) loss with respect to the codon hierarchy shown in Figure 3 following [78].

Sequences were tokenized using codon-level tokenization, resulting in vocabulary size of 70, including special tokens. The maximum context-length was set to 444, which is enough to accommodate all sequences in the pretraining dataset. However, the positional embedding layer was configured to support up to 2048 tokens, as such longer sequences can appear in certain downstream tasks. Positional embedding was applied following the strategy from GPT-2 [62].

Optimization was performed using the AdamW optimizer [44] with a weight decay of 1e-1. The learning rate was scheduled using linear warmup, followed by cosine decay, using an initial learning rate of 1e-4 which decayed to a minimum of 1e-5. Following [78], the α of the HXE loss was set to 0.2.

For the prototype classifiers, we used a prototype embedding dimension of 128 and used a scaling factor $\tau = 2.0$ for the embedding with h-MDS [72]. A hyperbolic linear layer [66] was used to project to the representation space. The temperature β was set to 10.

C Downstream tasks details

For downstream evaluation, we used a TextCNN [38] for each downstream task, following [45, 13, 55, 31, 78]. Our downstream configuration exactly matches that of [78]. So, we use a hidden size of 640 and 100 channels in the convolutions. The pretrained weights of the backbone are frozen during training. For each model we perform a hyperparameter search on the grid spanned by learning rates of 3e-4, 1e-4, 1e-5 and batch sizes 8, 16, 32, 64. The optimal hyperparameter configuration was chosen based on an unseen validation set. The final reported performance is determined on a separate test set. Each downstream dataset is split into 70% training, 15% validation and 15% test data.

D Sensitivity analysis with respect to choice of hyperparameters

To evaluate the robustness of our hyperbolic modeling approach, we performed a sensitivity analysis examining variations in curvature and threshold hyperparameters. The results, summarized in Table 3, indicate that the model's performance is relatively stable across the tested ranges.

Across most datasets, changes in hyperparameters lead to minor fluctuations in performance, demonstrating that the model does not rely heavily on precise hyperparameter tuning within this scope. For example, the performance on COVID-19, Ab1, and Fungal, the performance varies by a few percentage points across different hyperparameter settings.

Table 3: Sensitivity of model performance to hyperparameter variations.

Dataset	c=0.20, η=1.05	c=0.50, η=1.05	c=1.00, η=1.1	c=1.00, η=1.2	c=1.00, η=1.05
COVID-19	0.779	0.816	0.8	0.806	0.807
Ab1	0.739	0.742	0.717	0.724	0.751
Ab2	0.593	0.584	0.578	0.583	0.569
Fungal	0.733	0.748	0.733	0.732	0.741
P. pastoris	0.667	0.65	0.678	0.68	0.671