Towards interpretable Genomics Language Models:
Unlocking the Potential of Intermediate Embeddings

Prashant Gupta[1], Andrew Waters[1], David J Adams[1] ([1]Wellcome Sanger Institute, Cambridge, UK)

INTRODUCTION

Post Alpha fold, application of deep learning in biology has skyrocketed. Recently many models called "**genomics language models**" (gLMs) have been proposed – aimed at discovering the grammar of genome. These models have shown remarkable performance in various genomics prediction tasks; but their mechanism to generate those predictions are poorly understood. For instance, most existing interpretability efforts focus on model outputs and low dimensional visualisation, with little attention to the structure and dynamics of internal representations. Here, we take a first systematic step towards addressing this gap by introducing a representation-centric framework for analysing genomics language models. Our analysis outlines expectations for models that capture the grammar of the genome, including stable intermediate representations, structured responses to biologically meaningful perturbations, and smooth transitions across depth. Together, these results provide a principled foundation for analysing representation learning in genomics language models and support more interpretable, mechanistically grounded model evaluation.

METHODS

Variant Embeddings

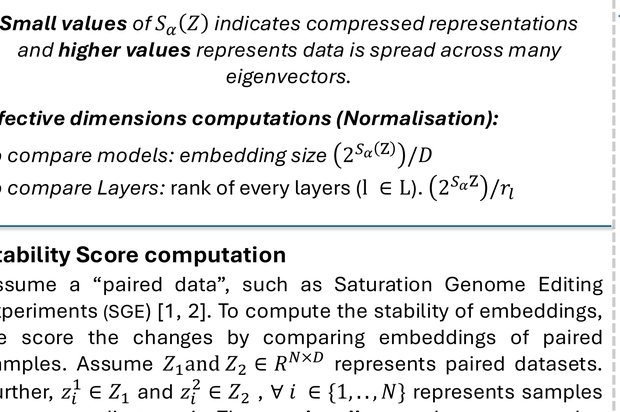

Figure 1: Methodology to compute variant embeddings. *Embeddings are the fixed length numerical representation of the input genomics sequence.* We extract representation for both reference and altered sequence and subtract them to create the embedding for a variant.

Entropy Computation (Rényi entropy)

For a dataset $Z \in R^{N\times D}$ where N is number of samples or tokens and D is embedding size. We further assume that a model has L layers.

$$S_\alpha(Z) = \frac{1}{1-\alpha}\log 2 \sum_{i=1}^{i=r}\left(\frac{\lambda_i(K)}{tr(K)}\right)^\alpha$$

Where $\mathrm{K} = \mathrm{Z}\mathrm{Z}^T, \mathrm{r} = \mathrm{rank}(\mathrm{K}) \leq \min(\mathrm{N}, \mathrm{D}), \lambda_i = i^{th}$ eigenvalue of K, and $\mathrm{tr}(\mathrm{K}) = \sum_{i=1}^{i=N} K_{ij}$ also known as trace of K.

We used $\alpha = 1.001$ in all our experiments.

Small values *of $S_\alpha(Z)$ indicates compressed representations and* ***higher values*** *represents data is spread across many eigenvectors.*

Effective dimensions computations (Normalisation):

To compare models: embedding size $\left(2^{S_\alpha(\mathrm{Z})}\right)/D$

To compare Layers: rank of every layers ($\mathrm{l} \in \mathrm{L}$). $\left(2^{S_\alpha \mathrm{Z}}\right)/r_l$

Stability Score computation

Assume a "paired data", such as Saturation Genome Editing experiments (SGE) [1, 2]. To compute the stability of embeddings, we score the changes by comparing embeddings of paired samples. Assume Z_1 and $Z_2 \in R^{N\times D}$ represents paired datasets. Further, $z_i^1 \in Z_1$ and $z_i^2 \in Z_2$, $\forall\, i \in \{1,..,N\}$ represents samples in corresponding matrix. Then **cosine distance** between samples can be given as:

$$c_{ij} = 1 - \frac{\left(z_i^1\right)^T z_j^2}{\|z_i^1\|\|z_j^2\|} \;\forall i,j \in \{1,..,N\}$$

Assume $C = \left[c_{ij}\right]_{N\times N}$, and $C^2 = \left[c_{ij}^2\right]_{N\times N}$ then baseline mean μ and std σ can be computed as

$$\mu = \frac{\sum_{i=1}^{i=N}\sum_{j=1}^{j=N} c_{ij} - tr(C)}{N\times(N-1)} \qquad \mu_2 = \frac{\sum_{i=1}^{i=N}\sum_{j=1}^{j=N} c_{ij}^2 - tr(C^2)}{N\times(N-1)}$$

$$\sigma = \sqrt{\mu_2 - \mu^2}$$

We compute stability score only for diagonal elements, as they represent the comparison of paired samples.

Stability score $s_{ij} = -\frac{c_{ij}-\mu}{\sigma}\; \forall i,j \in \{1,..,N\} \wedge i = j$

Stability score can be *interpreted* as:

$s_{ij} > 0$: No real change in representation.

$s_{ij} < 0$: Significant change in representation.

$s_{ij} \approx 0$: Changes are no greater than noise.

For good variant effect prediction, we want $s_{ij} < 0$.

Smoothness Score (curvature) Computation

Assume for an input, *token matrix* is given by $T \in R^{S\times D}$, where S is represents number of tokens or DNA length and D is embedding size. Further $t_i \in T$ represents embedding of a token. Thus, difference in consecutive embeddings v_i is given by:

$$v_k = t_{k+1} - t_k \forall k \in \{1,..,S-1\}$$

Then token curvature [3] can be computed as:

$$\kappa = \frac{1}{S-2}\sum_{k=1}^{k=S-2} \arccos\left(\frac{v_{k+1}^T v_k}{\|v_{k+1}\|\|v_k\|}\right)$$

In case of random walk (anti-corelated consecutive tokens) the expected curvature is $2\pi/3 = 2.094$ radian. Thus, to compute smoothness score, we normalise curvature with expected value as:

$$\text{Smoothness Score } \mathrm{Y} = 1 - \frac{\kappa}{2.094}$$

Smoothness score can be *interpreted* as:

$\mathrm{Y} > 0$: represents smoother curvature, representing that three consecutive tokens are expected to be in the neighbourhood.

$\mathrm{Y} < 0$: represent jagged curvature, representing three consecutive tokens probably are in foreign neighbourhood.

This score is particularly useful to probe the models understanding of genomics context. For most common sequences, the smoothness of the curve should be positive and higher than the rest, with the increased rareness of sequences the smoothness should go down.

Entropy-Curvature joint analysis:

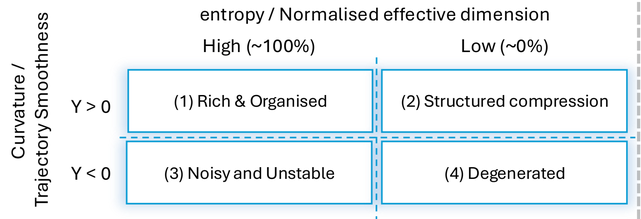

Figure 2: ***Joint analysis of entropy and curvature.*** We get most expressive and useful representation when the trajectory is smooth and more dimensions are used to express a sequence (first quadrant). With smooth trajectory, and high compression, embeddings may only be useful for a limited number of tasks as they may not have captured various genomics properties (second quadrant). High entropy embedding with jagged trajectory equates to noise and suggest model inability to learn useful representation (third quadrant). Low entropy and jagged curvature is indication of model collapse (fourth quadrant).

Layer level expectations for foundation models:

Lower layers: *Diverse & Unstable*	***Middle layers:*** *Rich & Organised*	***Upper layers:*** *Structured compression or Rich and Organised*
Moderate to high effective dimensions Random or slightly negative curvature.	High effective dimensions Positive curvature	Probable Decrease in effective dimension High positive curvature

RESULTS

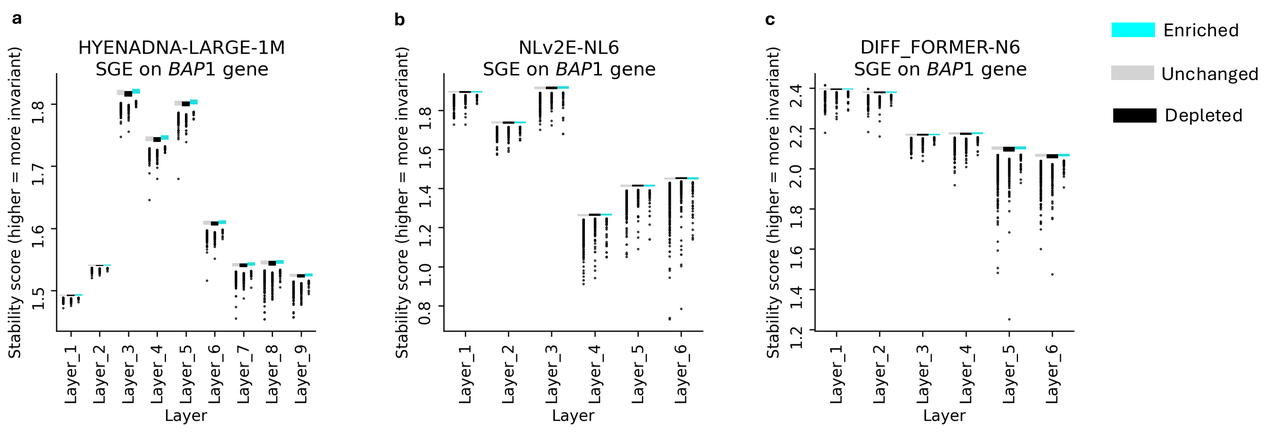

Figure 3: ***Embedding stability under sequence variation.*** SGE data consist of paired reference and altered sequences with known functional consequences, enabling assessment of embedding changes under minor perturbations. If a model captures genomic function, variants causing functional aberrations are expected to produce larger embedding changes and thus lower stability scores than non-aberrant variants. However, as shown in panels **a–c**, embeddings exhibit high stability regardless of functional consequence, indicating limited sensitivity to small sequence perturbations. Although. *(Resul*embedding differences increase modestly in deeper layers, they remain insufficient to capture functional effects*ts for NLv2E-NL15, NLv2E-NL29 and Diff Transformer–NL12 are omitted for visibility; DNABERT-2 and genaLM-BB are excluded due to dynamic tokenization.)*

Default Figure Legends:

Model Name	Emb size	# Layers*	# Parameters	Input Tokens	Legends
HyenaDNA-Large-1M [4]	256	9	6.5 M	Nucleotides	
**NTv2E-NL6 [5]	512	6	30.7 M	6-mers	
**NTv2E-NL12 [5]	512	12	55.9 M	6-mers	
**NLv2E-NL15 [5]	1024	15	263.5 M	6-mers	
**NLv2E-NL29 [5]	1024	29	498.6 M	6-mers	
genaLM-BB [6]	168	12	111.2 M	Dynamic length	
DNABERT-2 [7]	768	12	117 M	Dynamic length	
Diff_former – NL6 [8]	512	6	24.4 M	6-mers	
Diff_former – NL12 [8]	512	12	43.3 M	6-mers	

***Excluding input embeddings.** **Diff former** or **Differential Transformer** is custom implementation.

**** Locally trained from publicly available *instadeep* models as starting state [5],**

- **NL6 and NL12 are from** *nucleotide-transformer-v2-50m-multi-species*
- **NL15 and NL29 are from** *nucleotide-transformer-v2-500m-multi-species*

HyenaDNA-Large-1M, genaLM-BB, and DNABERT-2 were downloaded from huggingface.

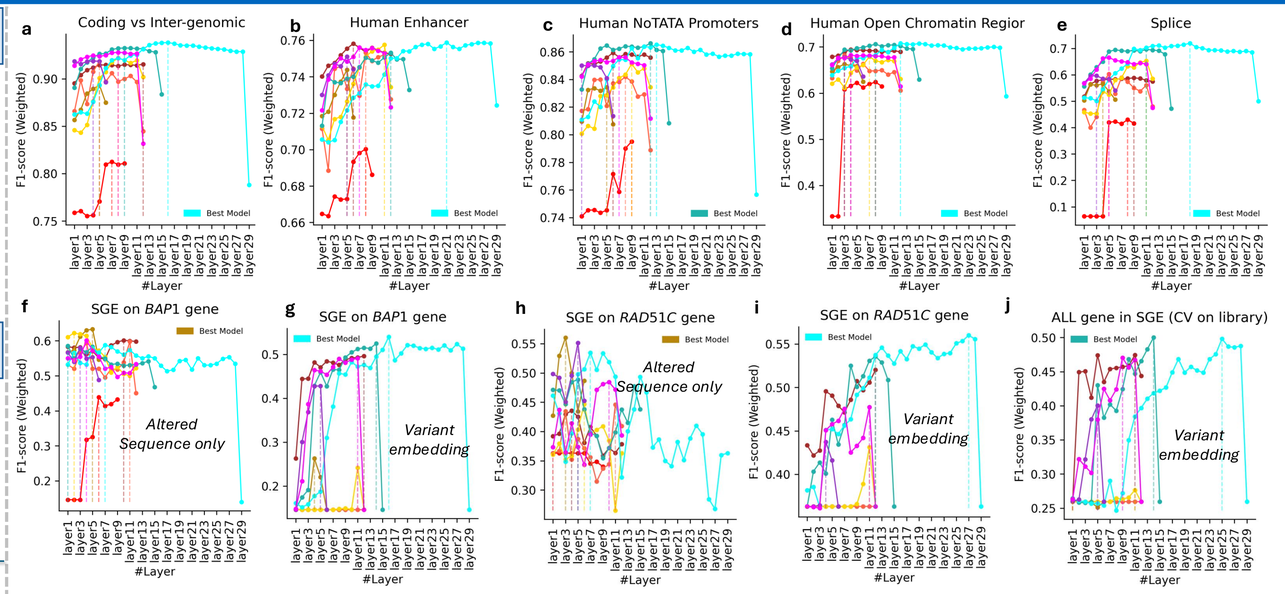

Figure 4: ***Classification accuracy (logistic regression) of embeddings across all layers of various genomics language models.*** For all datasets mean embeddings were created by averaging over token embeddings. Datasets **a–d** are from the Genomics Benchmark [9], and **e** is from the GUE benchmark [7]. Datasets **f** and **h** are from SGE experiments [8, 9]. Datasets **g** and **i** are derived from **f** and **h**, respectively, by subtracting the reference sequence embedding from the altered sequence embedding; these are termed *variant embeddings* (**Figure 1**). Dataset **j** comprises SGE data for ***BAP1*** [1], ***RAD51C*** [2], and ***POT1*** (unpublished). For datasets **a–e**, performance is reported on the test sets provided with the original datasets. Hyperparameter tuning was performed using 5-fold cross-validation on the combined train and development splits provided with the original datasets. For datasets **f–j**, average performance across five folds is reported, where folds were constructed by splitting libraries. Across all trained models, the best performance is consistently achieved at intermediate layers, after which performance plateaus or declines. This suggests that different layers encode distinct information and are suited to different types of tasks.

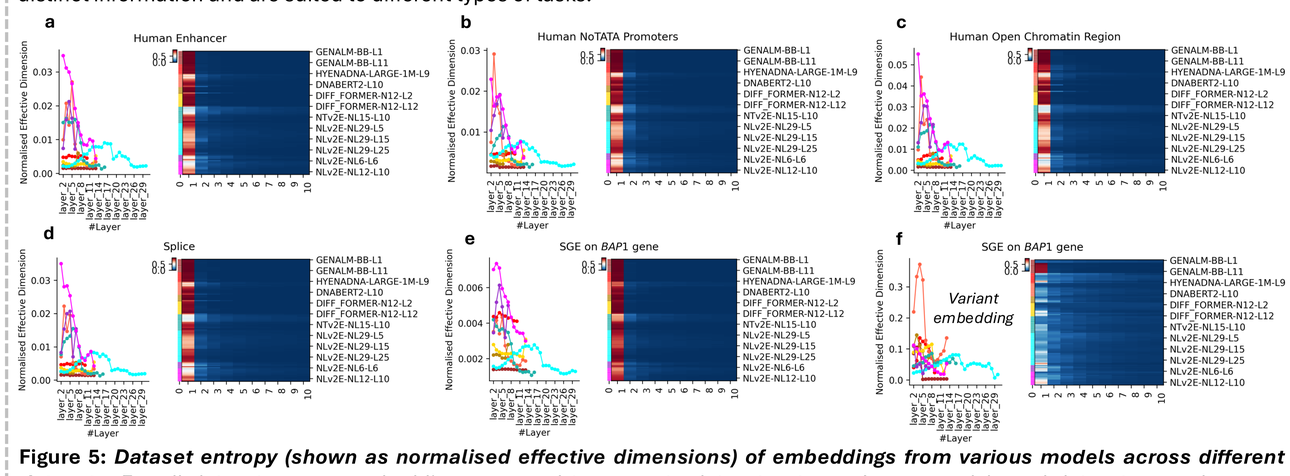

Figure 5: ***Dataset entropy (shown as normalised effective dimensions) of embeddings from various models across different datasets.*** For all datasets, mean embeddings are used to compute dataset entropy. Across models and datasets, we observe a phenomenon known as *embedding collapse*, in which embeddings lie in a low-dimensional manifold. If data reside in a low-dimensional manifold, their entropy is correspondingly low. Such low-dimensional embeddings may indicate that the model has not learned diverse contextual or functional properties, or that the input data are overly simple. In subplots **a–f**, this phenomenon is illustrated using two visualizations: (i) line plots showing dataset entropy across all layers of each model, and (ii) heatmaps showing the explained variance at each layer. For nearly all models, embeddings are confined to a low-dimensional manifold.

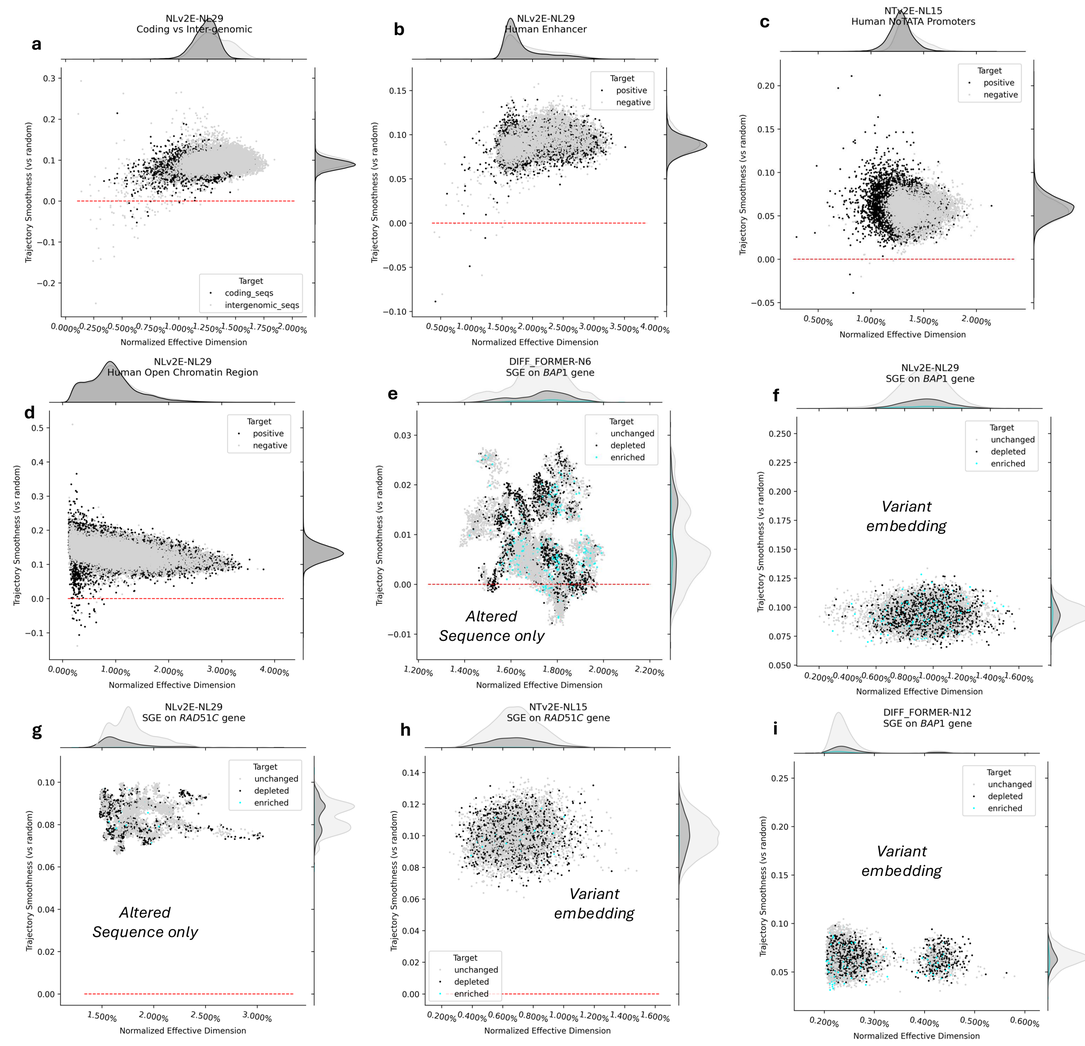

Figure 6: ***Joint distribution of normalised effective dimension and trajectory smoothness.*** Effective dimension, normalised by the rank of the token embedding matrix *(sequence length × embedding dimension)*, reflects the fraction of representational capacity utilised by a sample. Trajectory smoothness measures geometric continuity of token representations relative to a random baseline. Per-sample values are obtained by averaging both metrics across layers. Dataset labels and marginal label distributions are overlaid. Only the best-performing model for each classification task (**Figure 4;** except subpanel **6.i**) is shown. Across datasets **a–i**, embeddings occupy a low-dimensional manifold with smoother-than-random trajectories (>0), consistent with encoding global sequence features. Effective dimension separates labels for coding vs intergenic regions (**a**), human enhancers (**b**), and non-TATA promoters (**c**). SGE datasets (**e,** and **g**) exhibit clustered structure with modest distributional shifts, while variant embeddings lie in lower-dimensional manifolds without clear clustering. The **DIFF_FORMER-12** (and **DIFF_FORMER-6**, *not shown*) model shows two clusters without corresponding label separation. These plots highlight embedding geometry and its partial correspondence with classification performance. However, further investigation is required to establish a robust relationship between these geometric properties and downstream performance, and to enable clearer mechanistic interpretation of genomic language models.

CONCLUSIONS & FUTURE DIRECIONS

In this work, we evaluated publicly available and custom-trained genomics language models through their prediction dynamics. Although these models perform well on standard benchmarks, benchmark accuracy alone does not reliably reflect prediction stability. Under the proposed framework, all evaluated models fall into either the second (structured compressed) or fourth (degenerated) quadrant, highlighting limitations in their internal representations. This may partly stem from commonly used training objectives, such as masked language modelling, which may be insufficient for learning mechanistically rich genomic representations. Future work will extend this analysis to models trained with more complex, task-driven objectives, including Enformer [10] and Borzoi [11], and to more diverse and biologically challenging benchmarks [12]. As shown here, saturation genome editing (SGE) data offers a powerful lens for probing embedding dynamics and perturbation sensitivity. Expanding SGE datasets across genes and regulatory contexts will be essential for developing a more robust and generalizable interpretability framework for current and next-generation genomics language models.

REFERENCES

1. PMID: 38969833
2. PMID: 39299233
3. arXiv: 2502.02013
4. arXiv: 2306.15794
5. PMID: 39609566
6. bioRxiv: 2023.06.12.544594v3
7. arXiv: 2306.15006
8. arXiv: 2410.05258
9. PMID: 37127596
10. PMID: 34608324
11. PMID: 39779956
12. PMID: 41253815

pg20@sanger.ac.uk
aw28@sanger.ac.uk
da1@sanger.ac.uk