MECHANISTICALLY UNDERSTANDING HOW A BIOLOGICAL LANGUAGE MODEL PREDICTS BENEFICIAL MUTATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Language models have shown impressive results when applied to biological tasks. Concurrently, recent advancements in interpretability have produced methods that enable the computation of language models to be understood mechanistically. This paper examines what language models learn when trained on proteins. Specifically, we analyze CovFit, a protein language model introduced by Ito et al. (2025) trained to predict viral fitness from input SARS-CoV-2 sequences. We apply interpretability methods to (i) identify a MLP "look-up" computation that generalizes across several SARS-CoV-2 mutations, (ii) identify a particular biological circuit for the L455F mutation and (iii) show that highly conserved sites in SARS-CoV-2 are used as "attention sinks" over the course of computation.

1 INTRODUCTION

Protein language models (PLMs) have shown impressive results when applied to biological tasks, learning representations that capture structural and functional properties from sequence data alone (Rives et al., 2021; Lin et al., 2023). A particularly promising application has been to viral evolution, where PLMs have demonstrated success in predicting immune escape mutations (Hie et al., 2021) and viral fitness (Ito et al., 2025). Despite their empirical success, the mechanisms underlying PLM predictions remain largely opaque, limiting our ability to extract biological insights and understand the basis for their performance.

Mechanistic interpretability seeks to understand how black box models such as transformers arrive at their predictions. Applying mechanistic interpretability methods to protein language models is an emerging subfield of computational biology. Prior work has involved applying sparse autoencoders to different layer of protein language models to understand what biological features the models learn (Simon & Zou, 2024) and how these biological features are used in downstream tasks (Nainani et al., 2025; Tsui et al., 2026)).

In this work, we analyze CovFit, a variant of the popular protein language model ESM-2 that was introduced by Ito et al. (2025) to predict escape from a suite of antibodies and fitness of an input SARS-CoV-2 sequence. We seek to use existing mechanistic interpretability methods to understand how CovFit identifies mutations as benficial.

We examine the behavior of CovFit on 10 mutations of SARS-CoV-2 that are both biologically relevant and predicted to be beneficial by CovFit. We contribute the following findings:

1. First, we find that across almost all of the 10 mutations, CovFit exhibits a common computation pattern: A particular attention head in layer 31 moves sequence information into token 679 and then MLP 31 performs a "look-up" based on this information, and this "look-up" is the sole reason why CovFit predicts that mutation to be beneficial.
2. Next, for a particular mutation that is not entirely explained by this MLP "look-up," we reverse-engineer its full circuit.
3. Finally, we show more broadly that highly conserved sites may be used as so-called "attention sinks" (Xiao et al., 2024; Ruscio et al., 2025), connecting this known phenomenon to biology.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

2 BACKGROUND

CovFit is an ESM-2 650M model (Verkuil et al., 2022) with 33 layers (which we zero-index by 0 to 32 throughout the paper) which was fine-tuned on SARS-CoV-2 spike sequences via a 2-step training procedure: First, CovFit was trained on a Masked-Language-Modeling objective on spike proteins. We refer to this intermediate fine-tuned model hereon as "$\text{ESM}_{\text{coronaviridae}}$" as Ito et al. (2025) do in their paper. Then, an auxiliary 2-layer neural network is appended to the model which takes as input the $\langle$cls$\rangle$-token embedding and produces fitness and escape scores. We refer to this auxiliary 2-layer neural network hereon as the "CovFit-regressor."

CovFit takes as inputs SARS-CoV-2 spike protein sequences. The spike protein of SARS-CoV-2 consists of 1274 amino acids; since CovFit has a max context length of 1024, Ito et al. (2025) trained CovFit to only accept the first 1022 amino acids of the spike protein (the $\langle$cls$\rangle$-token is then appended to the beginning and the $\langle$eos$\rangle$-token is appended to the end to give the final 1024 context length). Biologically, Ito et al. (2025) justifies this choice because the last 250 amino acids do not encode for biological mechanisms involved with fitness (such as viral entry into a host cell and membrane fusion). We provide a schematic of CovFit, the CovFit-regressor, and $\text{ESM}_{\text{coronaviridae}}$, along with an explanation of the "fitness" logit in the appendix.

An early finding in our analysis is that the CovFit-regressor does not play any role in the final fitness prediction. To show this, we train a linear regression to predict the output of CovFit-regressor on a sequence given its $\langle$cls$\rangle$-token embedding. We find that there is almost perfect linear correlation (≈ 0.999 R^2) between the CovFit-regressor and the linear model. The details of this experiment are fleshed out in the appendix.

3 THE GENERAL "LOOK-UP" BY MLP 31

We now move to our first major finding: Discovering a general computation pattern across several different mutations. We analyze the following mutations: E484A, F456L, G339H, K417N, L455F, P681H, Q493R, R346T, S486P, and V445P, which were respectively reported by Liu et al. (2021); Focosi et al. (2023); Shikha et al. (2023); Barton et al. (2021); Focosi et al. (2023); Lista et al. (2022); Focosi et al. (2021); Groenheit et al. (2023); Tamura et al. (2024); Wang et al. (2023). These mutations naturally arose among different variants of SARS-CoV-2 and were highly beneficial for fitness.

Each of these mutations are written in the form "aa_1 Site aa_2", where the first letter "aa_1" corresponds to the wild-type amino acid, the number "site" corresponds to the location in the spike protein that is mutated, and the last letter "aa_2" corresponds to the mutated amino-acid. As an example, "L455F" denotes a mutation at site 455 from leucine (L) to phenylalanine (F). For each of our 10 mutations, we curate 3 datasets: (i) a "wild-type" dataset, consisting of sequences with aa_1 at the site, (ii) a counter-factual "mutation" dataset, consisting of corresponding sequences with aa_2 at the site (and identical at all other sites), and (iii) a "masked" dataset, consisting of corresponding sequences with a $\langle$mask$\rangle$-token at the site (and identical at all other sites). There are some technical details which we omit where "wild-type" sequences must be taken from the correct lineages and how we construct our "mutated" sequences dataset. We go through our full dataset curation pipeline in the appendix.

As noted earlier, CovFit predicts each of these mutations to be a pareto improvement of the sequence. As a result, the sequences in the "mutated" dataset almost always have a higher predicted fitness than their corresponding sequence pair in the "wild-type" dataset. We plot the distribution of predicted fitness differences in the appendix.

3.1 ACTIVATION PATCHING AND THE L31.H7 KEY-COMPOSITION CIRCUIT

We then perform activation patching, introduced by Wang et al. (2022) for each attention head (patching to the one of the "key," "query," "value," or "z," vectors or attention pattern) as well as for each MLP block in CovFit. For each mutation, we run CovFit on the "mutated" dataset and, for each component, we patch in what its output would have been on the "wild-type" dataset and re-compute all downstream components. We then measure the average change in the predicted fitness between the patched sequences and the "wild-type" sequences after patching to this component, normalizing

this value so that it is 0 if the predicted fitness is unaffected (i.e. the patched sequence has the same mean predicted fitness as the "mutated" sequences) and -1 if the predicted fitness difference is completely degraded (i.e. the patched sequence has the same mean predicted fitness as the "wild-type" sequences). We find that there are attention heads in layers 31 and 32 that consistently cause large changes in the predicted fitness when perturbed (in particular L31.H7, L32.H5, L32.H8 and L32.H13, where $\mathrm{L}\,n_1\,.\,\mathrm{H}\,n_2$ refers to attention head n_2 in layer n_1). We show activation results for one mutation (E484A) and show the rest in the appendix.

We then investigate whether any of these heads have a pattern in their attention matrix. Note that only the ⟨cls⟩-token is used by the CovFit-regressor for the final fitness prediction and all other token embeddings are discarded. Thus, for attention heads in layer 32, we only examine what tokens the ⟨cls⟩-token is attending to. We find that token 679 is consistently attended to by the ⟨cls⟩-token in the layer 32 attention heads. We note that some mutations are more straightforward, and the ⟨cls⟩-token simply attends to the mutated site to access the mutation information (such as F456L, P681H, and R346T). For most mutations, across their layer 32 attention heads, we find that the attention patterns of the ⟨cls⟩-token are consistent on both the "wild-type" and "mutated" datasets (in contrast to the layer 31 heads, which exhibit an increase in attention given to the mutated site on the "mutated" dataset). This indicates that the layer 32 attention heads primarily serve to passively move processed sequence information into the ⟨cls⟩-token to be decoded by the CovFit-regressor into the final predicted fitness logit (similar to "name mover heads" described in Wang et al. (2022)).

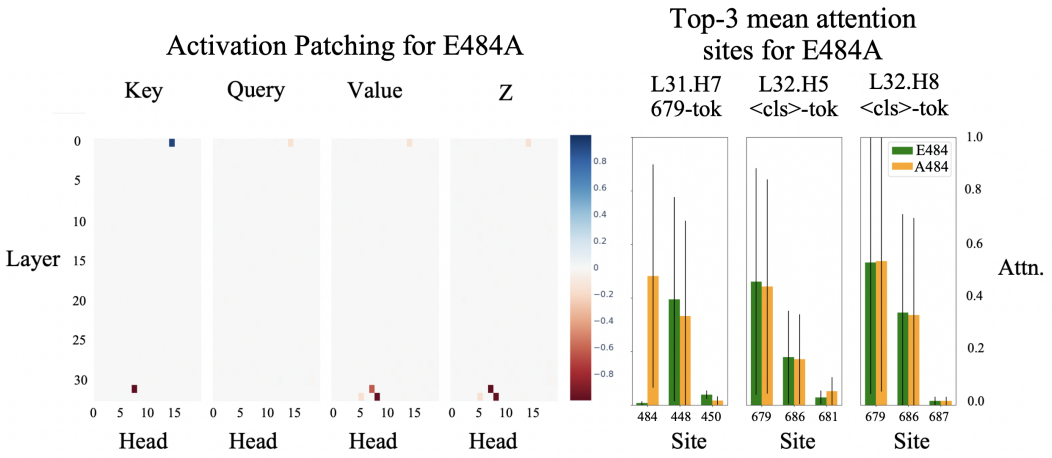

Figure 1: (Left) The activation patch results on E484A. The heads that cause the most downstream changes to fitness for this mutation when perturbed are L31.H7, L32.H5, and L32.H8. L31.H7 causes large downstream when its key-vector is perturbed. (Right) We examine the top-3 tokens that the ⟨cls⟩-token attends to in L32.H5 and L32.H8, finding that 679 receives around 50% of the attention. We also examine what token 679 attends to in L31.H7, and find that there is strong compositional behavior where the mutated site receives much more attention in the presence of the E484A mutation. The error bars on the histograms indicate the standard deviation.

We then use a variant of path-patching that we call absolute path-patching to determine the components that have a direct causal impact on the layer 32 attention heads. Explained briefly, rather than path-patching, where all intermediate attention heads' outputs are frozen and the intermediate MLP blocks' outputs are recomputed, in "absolute path-patching," we freeze all intermediate component outputs. This removes all confounding elements and, as a result, we can attribute all the downstream change in the model to the perturbed component. The full details are given in the appendix.

We find that the only component that has a direct causal impact on the layer 32 attention heads is MLP 31. Across all mutations, MLP 31 composes with the layer 32 attention heads via their value vectors. This proves that MLP 31 performs the final step of computation to boost the predicted fitness (If all computation finished at the attention block in layer 31 before MLP 31, then the layer 32 attention heads would directly compose with the layer 31 attention heads to move the processed information into the ⟨cls⟩-token; MLP 31 would not have a large causal impact on the layer 32

attention heads. That MLP 31 has a large causal impact on the layer 32 attention heads proves that computation does not finish at the layer 31 attention heads, which leads to the conclusion that MLP 31 plays the critical final role in processing the sequence information to boost the predicted fitness for mutated sequences).

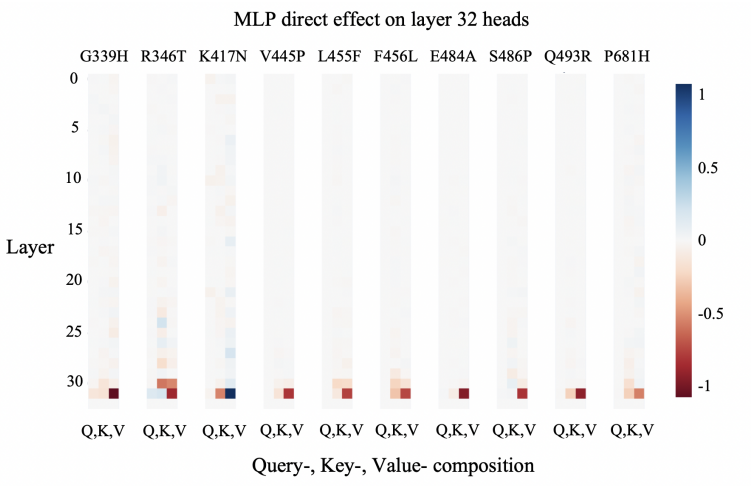

Figure 2: Absolute path-patching to relevant layer 32 attention heads corresponding to each mutation: Note that each mutation has different relevant layer 32 attention heads (see activation patching results). Importantly, MLP 31 can be seen to be to be consistently composing with the corresponding layer 32 attention heads via the value vector. We list the respective layer 32 attention heads that we patch to for each mutation here:
G339H: {L32.H5, L32.H8}, R346T: {L32.H5, L32.H8}, K417N: {L32.H5, L32.H8}, V445P: {L32.H5, L32.H8}, L455F: {L32.H5, L32.H8, L32.H13}, F456L: {L32.H5, L32.H8, L32.H13}, E484A: {L32.H5, L32.H8}, S486P: {L32.H5, L32.H8}, Q493R: {L32.H5, L32.H8}, P681H: {L32.H8}

3.2 Explaining the role of layer 31 attention heads

Across many mutations, we also find that L31.H7 causes a large downstream impact on the predicted fitness when perturbed. In particular, this occurs when counterfactual information is patched to its key vector. For mutations where this occurs (E484A, G339H, K417N, L455F, Q493R, and V445P), we find that, on the "mutated" sequences, token 679 attends much more to the mutated site than on the "wild-type" sequences. Token 679 is the same token position that the ⟨cls⟩-token consistently reads from in the layer 32 attention heads, indicating that L31.H7 composes with the layer 32 attention heads mediated through MLP 31. We confirm that L31.H7 composes with the layer 32 attention heads mediated through MLP 31 via absolute path patching in the appendix.

3.3 Ablating MLP 31 outputs

We have thus far shown the following: Working backwards from the logits, the CovFit-regressor (essentially) takes a linear combination of the elements of the final ⟨cls⟩-token embedding to produce the final fitness output. Across all of our mutations, the relevant information is placed into the ⟨cls⟩-token embedding by layer 32 attention heads moving the processed sequence information, using the output from MLP 31. The ⟨cls⟩-token usually places most of its attention on token 679 in the layer 32 attention heads. In the mutations where this occurs, there is a circuit which, in the presence of a mutation "aa_1 Site aa_2" causes sequence information to be moved from the mutated "Site" to token 679 by L31.H7 via key-composition.

We claim that MLP 31 performs a "look-up" to identify the relevant information at token 679 and the mutated site to move to the ⟨cls⟩-token via value-composition with the layer 32 attention heads. To show this, we perform an ablation experiment where, for each mutation, we patch the "wild-type"

Table 1: Ablation Experiment Results on the Output of MLP31

Ablated Site\Mutation	**E484A**	**F456L**	**G339H**	**K417N**	**L455F**
679 and Site	0.911	**0.383**	**0.706**	**0.958**	**0.614**
679 only	**0.912**	0.001	0.630	0.953	0.292
Site only	-0.000	0.382	0.074	0.004	0.321
⟨mask⟩ at Site	-0.000	0.365	0.073	0.004	0.332

Ablated Site\Mutation	**P681H**	**Q493R**	**R346T**	**S486P**	**V445P**
679 and Site	0.853	0.943	**0.804**	0.172	**0.798**
679 only	**0.871**	**0.944**	0.196	-0.001	0.794
Site only	-0.459	-0.000	0.604	**0.172**	0.004
⟨mask⟩ at Site	-0.001	-0.001	0.539	0.049	0.002

sequences into the output of MLP 31 at (i) site 679 and the mutated site, (ii) only site 679, and (ii) only the mutated site. As a control, we patch the "masked" sequences into the output of MLP 31 at the mutated site. We then measure how much the predicted fitness difference changes using a similar metric as for activation patching, giving a score of 0 if the ablated output has the same mean predicted fitness as the "mutated" sequence and giving a score of 1 (rather than -1) if an ablation removes the effect of the mutation, i.e. the ablated output has the same mean predicted fitness as the "wild-type" sequences. The results are shown in Table 1.

We find that for most mutations (apart from L455F, F456L, and S486P), ablating token 679 and the mutated site almost completely removes the effect of the mutation. It should be noted that, since we are only patching to the output of MLP 31, all the aggregated information from earlier layers will still be present in the residual stream after MLP 31. That we are able to remove the mutation effect confirms that layer 32 attention heads directly use the output of MLP 31, as opposed to cached information in the residual stream. We find that it is crucial to ablate token 679, as simply ablating the site does not always remove the effect of the mutation, indicating that MLP 31 consistently uses token 679 for its "look-up." We posit that the reason ablation of MLP 31's outputs does not remove the predicted fitness increase for some mutations is because there are parallel computation paths that move sequence information into the ⟨cls⟩-tokens. In the next section, we turn our attention to fully reverse-engineering the circuit for one such mutation: L455F.

4 THE L455F CIRCUIT

In addition to finding a computation pattern that generalizes across several mutations, we also contribute a fully reverse-engineered circuit for the L455F mutation.

4.1 COMPONENTS INVOLVED IN L455F CIRCUIT

For this section we adopt the following notation: We let M denote the set of attention heads and MLP blocks in CovFit. By repeatedly performing path-patching and absolute path-patching, we obtain what we believe to be the circuit $C \subset M$ responsible for the fitness boost from the L455F mutation. Our proposed circuit C consists of the following components:
Attention heads: {{L0.all}, {L1.H15}, {L2.H4, L2.H17}, {L12.H18}, {L13.H9}, {L14.H6}, {L31.H1, L31.H2, L31.H3, L31.H7, L31.H8, L31.H9, L31.12, L31.H13, L31.H14, L31.H15, L31.H17}, {L32.H5, L32.H8, L32.H13}}
MLP blocks: {0,2,8,11,14,15,16,18-31}

We note that this circuit is quite large. We attribute this to CovFit's large number of layers as a transformer. There are many ways that information can be trickle into the ⟨cls⟩-token either via site 679 and the subsequent MLP "look-up" or via other computational paths. In addition, we believe

Table 2: L455F Circuit Faithfulness and Completeness scores

Eval	Score
Faithfulness	0.8655
Completeness	0.9999

that computation is smeared across many components through indirect relationships. As a result, many components become entangled in processing related information for the L455F mutation and must be included in our circuit, even if they may not have a first- or even second-order interaction with relevant sequence information.

We broadly categorize the attention heads into the following categories: ⟨cls⟩-mover heads, aa455 heads, and prev./succ. heads. ⟨cls⟩-mover heads are used to describe the attention heads in layer 32, which have a direct causal impact on the final fitness prediction and primarily serve to move processed information into the ⟨cls⟩-token via value composition. The aa455 tokens describe the layer 31 attention heads which compose with the layer 32 attention heads mediated through MLP 31. Finally, prev./succ. heads describe almost all other heads and are attention heads that have high mean attention weight on one of the diagonals. These heads serve to aggregate information from previous (prev.) or succeeding (succ.) tokens a fixed distance away, and compose with each other and layer 31 attention heads mediated by the MLP blocks. We describe each of the attention heads with more detail in the appendix.

4.2 L455F CIRCUIT EVALUATIONS

We provide evaluations for our circuit following Marks et al. (2025): We provide the summary scores for Faithfulness and Completeness in table 2, which are defined as:

$$\text{Faithfulness}(C) := \left(\frac{m(C) - m(\phi)}{m(M) - m(\phi)}\right), \qquad \text{Completeness}(C) = 1 - \text{Faithfulness}(M - C)$$

where $m(\cdot)$ denotes the fitness difference after passing the sequence through the circuit. Under these definitions, a perfect circuit would have a completeness and faithfulness score of 1. Rather than zero-ablating components not in our circuit, we follow a procedure similar to Wang et al. (2022) where we ablate the output of the component with its mean output using our "masked" dataset for L455F. Mean-ablating in this manner ensures that semantic information about the sequence unrelated to the L455F task is still outputted by the components not in the circuit, which helps model outputs be more stable under ablations.

In addition, we provide evaluations of the circuit's completeness as in Wang et al. (2022) in figure 3. That is, for selected subsets $K \subset C$ of our circuit, we compare $m(C - K)$ against $m(M - K)$. This evaluation measures how similar our circuit C is to M even under ablations.

5 HIGHLY-CONSERVED SITES AS "ATTENTION SINKS"

In our activation patching results, it can be seen that certain token positions other than token 679 are consistently attended to (such as token 448). In this section, we show that this is not a coincidence, and that highly-conserved sites are used as "attention sinks."

As a brief background, "attention sinks" were first examined by Xiao et al. (2024), where tokens at certain positions received a large proportion of attention regardless of their semantic relevance. They arise due to the mathematical properties of the SoftMax$(\cdot)$ function applied to the attention weights: Since the final attention weights must add up to 1, any decrease in attention given to a token must be balanced by an increase in attention given to another site. "Attention sites" are token positions that consistently receive these increases in attention. In addition, Ruscio et al. (2025) show that "attention sinks" serve as geometric anchor points for transformers, i.e. they provide a consistent coordinate basis for high-dimensional token representations. A mathematically rigorous explanation behind the emergence of these sites can be found in Zuhri et al. (2026); Ruscio et al. (2025).

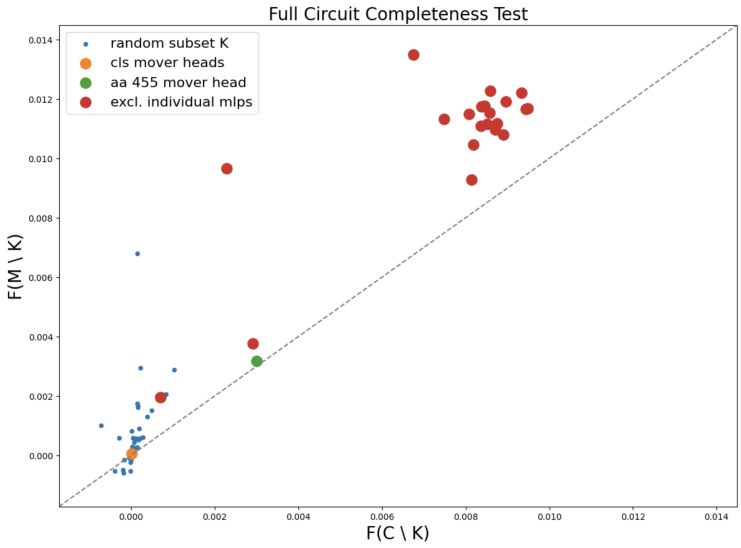

Figure 3: Completeness test for our L455F circuit C as in Wang et al. (2022). We select $K \subset C$ subsets of our circuit to ablate and compare the predicted fitness difference on $C - K$ vs. $M - K$. In the figure, $F(A)$ denotes the fitness difference after passing the "wild-type" and "mutated" sequences for L455F through any subset $A \subset M$

A somewhat related but computationally distinct phenomenon is that of "register tokens" (Darcet et al., 2024). Often seen in Vision Transformers, "register tokens" describe the phenomenon where semnatically related pixels in images are re-purposed for computation. The primary distinction between "register tokens" and "attention sinks" is the information stored in the token: "register tokens" are re-purposed for the purpose of computation, and so they contain global information about the input. In contrast, "attention sinks" are used as a convenient way to distribute attention and do not have the same property of containing global information about the input.

Next, to quantify how conserved a site is, we examine the amino-acid diversity of a site within a protein using the site-wise entropy (Shannon, 1948) of the site:

$$\text{entropy(site)} := -\sum_{x \in aa_{\text{site}}} p(x)\ \log(p(x))$$

where aa_{site} is the set of all unique amino acids that have empirically occurred at that site and $p(x)$ is the empirical probability mass function for aa_{site}. We define a highly-conserved sites to be a site with site-wise entropy $= 0$. Next, we use the mathematical formalization of giving each token position an "attention sink" score as done in Ruscio et al. (2025): Given a fixed attention head h in a fixed layer ℓ, a token position j is defined to have an "attention sink" score $\text{sink}^{(\ell,h)}(j)$ by:

$$\text{sink}^{(\ell,h)}(j) = \frac{1}{n}\sum_{i=1}^{n} \chi_{\{\alpha_{ij} \geq \tau\}}$$

where i ranges over a large n number of source tokens, α_{ij} is the attention weight (in head h in layer ℓ) that source token i gives to position j token, τ represents the 90th percentile of attention weights (taken wrt the attention head and layer of α_{ij} on the source token). Ruscio et al. (2025) defines an "attention sink" to be a token position j that has $\text{sink}^{(\ell,h)}(j)$ greater than or equal to some fixed threshold γ.

To get an idea of global patterns among the two models, in our experiments, we aggregate the per-layer, per-attention head scores $\text{sink}^{(\ell,h)}(j)$ into a single summary statistic. However, we exclude some layers from this aggregation. This comes from our finding on the distribution of token norms over the layers. We find that from layers 18-31, the distribution of token norms has a heavy tail where a small proportion of token embeddings seem to have a higher norm on average than others.

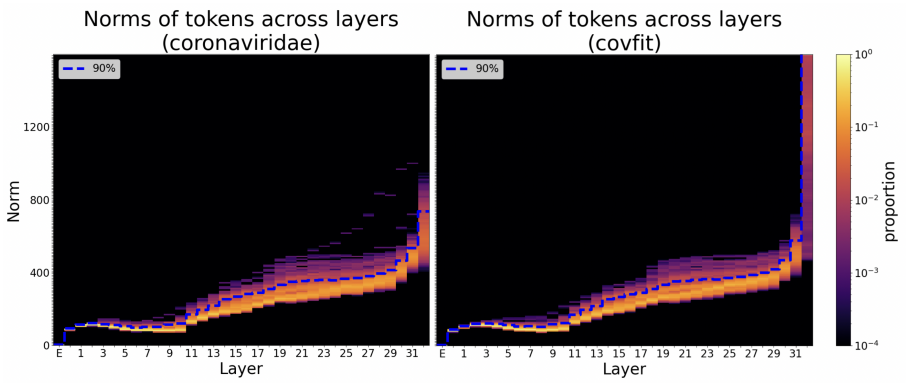

Figure 4: Heatmap of residual stream norms over layers. Over all sequences in our dataset, for each token in the sequence, we cache each of its residual stream embeddings (including its initial embedding) and take its norm. We end up with a matrix of shape $[(\#\text{layers}+1), (\#\text{tokens} \times \#\text{sequences})]$ which we then take a histogram across the final dimension of to produce the plot. The blue line indicates the 90th percentile of norms. From layer 18-31, we see a distinctive increase in the range of the tail of the distribution.

As a result, we only aggregate from layers 18-31, and aggregate over all attention heads to arrive at a summary statistic for the attention sink score for a token position Cumulative-sink(j):

$$\text{Cumulative-sink(j)} = \frac{1}{(\ell_2 - \ell_1 + 1) * H} \sum_{\ell=\ell_1}^{\ell_2} \sum_{h=1}^{H} \text{sink}^{(\ell,h)}(j)$$

where $\ell_1 = 18$, $\ell_2 = 31$ and $H := \#\text{attention heads}$. To account for the potential loss of signal by performing this mean aggregation, we apply an additional filter by examining the norm. We construct similar summary statistics score for each token position's norm information High-Norm(j):

$$\text{High-Norm}(j) = \frac{1}{(\ell_2 - \ell_1 + 1)} \sum_{\ell=\ell_1}^{\ell_2} \text{norm-score}^{(\ell)}(j), \qquad \text{norm-score}^{(\ell)}(j) = \frac{1}{n} \sum_{\text{seq}=1}^{n} \chi_{\{||\text{seq}_j^{(\ell)}||_2 \geq \tau^{(\ell)}\}}$$

where $||\text{seq}_j^{\ell}||_2$ denotes the standard Euclidean norm of the residual stream embedding of the token in layer ℓ on a sequence seq, and τ^{ℓ} represents the 90th percentile of token norms in layer ℓ. We show the joint plot of the High-norm$(\cdot)$ scores against the Cumulative-sink$(\cdot)$ scores. We focus on token positions that are in the 95th percentile of both the High-norm$(\cdot)$ scores and the Cumulative-sink$(\cdot)$ scores. This group of token positions is both consistently attended to and has consistently high norms. We posit that these tokens are used as for "garbage collection" by the model: They are convenient low-priority sites where attention can be increased so that other tokens can receive decreased attention and, because they are consistently attended to with large attention scores, they accumulate extraneous "garbage" information from other tokens. We find that there is somewhat of an overlap between the token positions in the 95th percentile for the marginal densities, where the respective Intersection-Over-Union metrics for CovFit and $\text{ESM}_{\text{coronaviridae}}$ are 0.400 and 0.333.

We find that, among this group of consistently high-norm, high-sink tokens, respectively $17/25$ ($\sim 70.37\%$) and $20/30$ ($\sim 66.67\%$) of them are highly-conserved sites for CovFit and $\text{ESM}_{\text{coronaviridae}}$. Of the 1022 amino acids that the models take as input, we find that $367/1022$ ($\sim 35.91\%$) sites are highly-conserved. That is, under a null hypothesis where a site being consistently high-norm and high-sink is independent of whether that site is highly-conserved, we would expect around 35.91% of the consistently high-norm and high-sink tokens positions to correspond to highly-conserved sites. That the percentages are much higher than 35.91% indicates that there is some relationship between a token being consistently high-norm and high-sink and being highly-conserved.

We posit that CovFit and $\text{ESM}_{\text{coronaviridae}}$ exploit the fact that these tokens have no variability across sequences. That "attention sinks" are used as reference frames supports this hypothesis as well. We believe that this is not unique to CovFit and $\text{ESM}_{\text{coronaviridae}}$, and that for transformers trained on

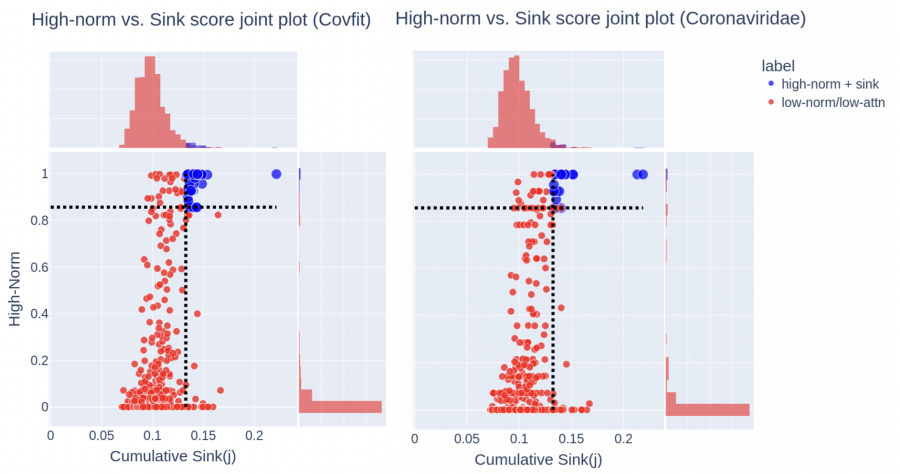

Figure 5: Joint plot of High-Norm scores vs. Cumulative-Sink scores. The blue points are the token positions we analyze. The dotted lines indicate the 95th percentile for the marginal densities. The respective Intersection-Over-Union between the blue dots and all token positions that are in the 95th percentile for atleast one of the marginal densities for CovFit and $\text{ESM}_{\text{coronaviridae}}$ is 0.400 and 0.333 respectively.

biological sequences, identifying tokens that have low biological variability is a necessary step in order for the transformer to construct useful internal representations.

6 Discussion and Limitations

In this work, we used interpretability methods to understand how CovFit predicts certain mutations to be beneficial. We first examined a general computation pattern that generalized across several mutations, then did a deep dive on a particular mutation, and then analyzed the tokens that were consistently attended to and that had consistently high-norm.

We hypothesize that our finding that highly-conserved sites are frequently used as "garbage collector" tokens is part of a larger mechanism underlying all transformers, where redundant information in the distribution of training data is identified and re-purposed for computation.

There are two important limitations of our analysis: The first is our use of synthetic sequences. However, while these sequences may not always well-defined biologically, we believe that their use in our analysis strengthens our claim that the identification of a beneficial mutation is algorithmic in nature. The second is our use of only 80 paired sequences in the path-patching experiments. For the path-patching and absolute path-patching experiments in section 3 ONLY, we use a random subset of 80 "wild-type" and 80 "mutated" sequences per mutation due to the amount of time it takes to run patching experiments ($\sim$ 2 hours per mutation if doing path-patching and $\sim$ 1 hour per mutation if doing absolute path-patching). For all other experiments (such as examining the top-3 attention sites, MLP ablation, attention sink analysis, and so on), we use all of the sequences (or all of the sequences in our "wild-type," "mutated," and "masked" dataset). We believe that this limitation should not reduce the strength of our results due to the amount of experiments run on the full dataset of sequences which corroborate our path-patching findings.

We view CovFit and $\text{ESM}_{\text{coronaviridae}}$ as toy models for the larger biological AI models that are the current state-of-the-art such as DASM (Matsen et al., 2026), AlphaFold (Jumper et al., 2021), and ProteinGPT (Xiao et al., 2025). Developing robust tools of mechanistic analysis to diagnose how black-box biological models are arriving at their predictions is essential to ensure accurate and meaningful biological output as these models are increasingly integrated into real-world biological applications.

AI USE STATEMENT

In this work, we used generative AI for a literature search to find the papers on "attention sinks." All other work for this paper including writing, formatting, and code for experiments was done without the use of generative AI.

REPRODUCIBILITY STATEMENT

We use the version of CovFit (along with its corresponding CovFit-regressor and $\text{ESM}_{\text{coronaviridae}}$ model) that was trained on SARS-CoV-2 spike sequences until October 2024. These are all publicly available and are linked by Ito et al. (2025) in their paper via their github.

Our spike sequence data is publicly available and comes the Nextstrain platform (Hadfield et al., 2018), where we combine the SARS-CoV-2 sequences obtained from the Africa, Asia, Europe, North America, Oceania, South America, and global (subsampled) datasets.

We run all of of our experiments using Colab with an A100 GPU that has 80 GB RAM.

Some of the code for path-patching functions was adapted and borrowed from an AI Safety course created by McDougall.

REFERENCES

Michael I Barton, Stuart A MacGowan, Mikhail A Kutuzov, Omer Dushek, Geoffrey John Barton, and P Anton van der Merwe. Effects of common mutations in the sars-cov-2 spike rbd and its ligand, the human ace2 receptor on binding affinity and kinetics. *eLife*, 10:e70658, aug 2021. ISSN 2050-084X. doi: 10.7554/eLife.70658. URL https://doi.org/10.7554/eLife.70658.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 2632–2652, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/0b408293619f725fd30162af057e531a-Paper-Conference.pdf.

Nicholas G Davies, Sam Abbott, Rosanna C Barnard, Christopher I Jarvis, Adam J Kucharski, James D Munday, Carl A B Pearson, Timothy W Russell, Damien C Tully, Alex D Washburne, Tom Wenseleers, Amy Gimma, William Waites, Kerry L M Wong, Kevin van Zandvoort, Justin D Silverman, Karla Diaz-Ordaz, Ruth Keogh, Rosalind M Eggo, Sebastian Funk, Mark Jit, Katherine E Atkins, and W John Edmunds. Estimated transmissibility and impact of sars-cov-2 lineage b.1.1.7 in england. *Science (New York, N.Y.)*, 372(6538), 2021. doi: 10.1126/science.abg3055.

Daniele Focosi, Federica Novazzi, Angelo Genoni, Francesco Dentali, Daniela Dalla Gasperina, Andreina Baj, and Fabrizio Maggi. Emergence of sars-cov-2 spike protein escape mutation q493r after treatment for covid-19. *Emerging infectious diseases*, 27(10):2728–2731, 2021. doi: 10.3201/eid2710.211538.

Daniele Focosi, Pietro Giorgio Spezia, Federico Gueli, and Fabrizio Maggi. The era of the flips: How spike mutations l455f and f456l (and a475v) are shaping sars-cov-2 evolution. *Viruses*, 16 (1), 2023. doi: 10.3390/v16010003.

Ramona Groenheit, Ilias Galanis, Klara Sondén, Maike Sperk, Elin Movert, Philip Bacchus, Tatiana Efimova, Lina Petersson, Marie Rapp, Viktoria Sahlén, Sandra Söderholm, Karin Valentin Asin, Sarah Zanetti, Maria Lind Karlberg, Andreas Bråve, Jonas Klingström, and Kim Blom. Rapid emergence of omicron sublineages expressing spike protein r346t. *The Lancet Regional Health – Europe*, 24, Jan 2023. ISSN 2666-7762. doi: 10.1016/j.lanepe.2022.100564. URL https://doi.org/10.1016/j.lanepe.2022.100564.

James Hadfield, Colin Megill, Sidney M Bell, John Huddleston, Barney Potter, Charlton Callender, Pavel Sagulenko, Trevor Bedford, and Richard A Neher. Nextstrain: real-time tracking of pathogen evolution. *Bioinformatics*, 34(23):4121–4123, May 2018. ISSN 1367-4811. doi: 10.1093/bioinformatics/bty407. URL http://dx.doi.org/10.1093/bioinformatics/bty407.

Brian Hie, Ellen D Zhong, Bonnie Berger, and Bryan Bryson. Learning the language of viral evolution and escape. *Science*, 371(6526):284–288, 2021.

Jumpei Ito, Adam Strange, Wei Liu, Gustav Joas, Spyros Lytras, Keita Matsuno, Naganori Nao, Hirofumi Sawa, Keita Mizuma, Isshu Kojima, Jingshu Li, Tomoya Tsubo, Shinya Tanaka, Masumi Tsuda, Lei Wang, Yoshikata Oda, Zannatul Ferdous, Kenji Shishido, Takasuke Fukuhara, Tomokazu Tamura, Rigel Suzuki, Saori Suzuki, Shuhei Tsujino, Hayato Ito, Yu Kaku, Naoko Misawa, Arnon Plianchaisuk, Ziyi Guo, Alfredo A. Hinay, Kaoru Usui, Wilaiporn Saikruang, Keiya Uriu, Yusuke Kosugi, Shigeru Fujita, Jarel Elgin M.Tolentino, Luo Chen, Lin Pan, Wenye Li, Mai Suganami, Mika Chiba, Ryo Yoshimura, Kyoko Yasuda, Keiko Iida, Naomi Ohsumi, Shiho Tanaka, Kaho Okumura, Kazuhisa Yoshimura, Kenji Sadamas, Mami Nagashima, Hiroyuki Asakura, Isao Yoshida, So Nakagawa, Akifumi Takaori-Kondo, Kotaro Shirakawa, Kayoko Nagata, Ryosuke Nomura, Yoshihito Horisawa, Yusuke Tashiro, Yugo Kawai, Kazuo Takayama, Rina Hashimoto, Sayaka Deguchi, Yukio Watanabe, Yoshitaka Nakata, Hiroki Futatsusako, Ayaka Sakamoto, Naoko Yasuhara, Takao Hashiguchi, Tateki Suzuki, Kanako Kimura, Jiei Sasaki, Yukari Nakajima, Hisano Yajima, Takashi Irie, Ryoko Kawabata, Kaori Sasaki-Tabata, Terumasa Ikeda, Hesham Nasse, Ryo Shimizu, MST Monira Begum, Michael Jonathan, Yuka Mugita, Sharee Leong, Otowa Takahashi, Kimiko Ichihara, Takamasa Ueno, Chihiro Motozono, Mako Toyoda, Akatsuki Saito, Maya Shofa, Yuki Shibatani, Tomoko Nishiuchi, Jiri Zahradni, Prokopios Andrikopoulos, Miguel Padilla-Blanco, Aditi Konar, and Kei Sato. A protein language model for exploring viral fitness landscapes. *Nature Communications*, 16(1), May 2025. ISSN 2041-1723. doi: 10.1038/s41467-025-59422-w. URL http://dx.doi.org/10.1038/s41467-025-59422-w.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, Aug 2021. ISSN 1476-4687. doi: 10.1038/s41586-021-03819-2. URL https://doi.org/10.1038/s41586-021-03819-2.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

Maria Jose Lista, Helena Winstone, Harry D. Wilson, Adam Dyer, Suzanne Pickering, Rui Pedro Galao, Giuditta De Lorenzo, Vanessa M. Cowton, Wilhelm Furnon, Nicolas Suarez, Richard Orton, Massimo Palmarini, Arvind H. Patel, Luke Snell, Gaia Nebbia, Chad Swanson, and Stuart J. D. Neil. The p681h mutation in the spike glycoprotein of the alpha variant of sars-cov-2 escapes ifitm restriction and is necessary for type i interferon resistance. *Journal of Virology*, 96(23): e01250–22, 2022. doi: 10.1128/jvi.01250-22. URL https://journals.asm.org/doi/abs/10.1128/jvi.01250-22.

Zhuoming Liu, Laura A. VanBlargan, Louis-Marie Bloyet, Paul W. Rothlauf, Rita E. Chen, Spencer Stumpf, Haiyan Zhao, John M. Errico, Elitza S. Theel, Mariel J. Liebeskind, Brynn Alford, William J. Buchser, Ali H. Ellebedy, Daved H. Fremont, Michael S. Diamond, and Sean P.J. Whelan. Identification of sars-cov-2 spike mutations that attenuate monoclonal and serum antibody neutralization. *Cell Host & Microbe*, 29(3):477–488.e4, Mar 2021. ISSN 1931-3128. doi: 10.1016/j.chom.2021.01.014. URL https://doi.org/10.1016/j.chom.2021.01.014.

Samuel Marks, Can Rager, Eric Michaud, Yonatan Belinkov, David Bau, and Aaron Mueller. Sparse feature circuits: Discovering and editing interpretable causal graphs in language models. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 23888–23923, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/3ba4d47a83e498c2b1a0868cba20f6de-Paper-Conference.pdf.

Frederick A Matsen, Will Dumm, Kevin Sung, Mackenzie M Johnson, David Rich, Tyler Starr, Yun S Song, Julia Fukuyama, and Hugh K Haddox. Separating selection from mutation in antibody language models. March 2026. doi: 10.7554/elife.109644.2. URL http://dx.doi.org/10.7554/eLife.109644.2.

Callum McDougall. Arena transformer interpretability. https://arena-chapter1-transformer-interp.streamlit.app/. [Accessed 25-09-2026].

Jatin Nainani, Bryn Marie Reimer, Connor Watts, David Jensen, and Anna G. Green. Mechanistic evidence that motif-gated domain recognition drives contact prediction in protein language models. In *Mechanistic Interpretability Workshop at NeurIPS 2025*, 2025. URL https://openreview.net/forum?id=cvCgSb9xeQ.

Fritz Obermeyer, Martin Jankowiak, Nikolaos Barkas, Stephen F Schaffner, Jesse D Pyle, Leonid Yurkovetskiy, Matteo Bosso, Daniel J Park, Mehrtash Babadi, Bronwyn L MacInnis, Jeremy Luban, Pardis C Sabeti, and Jacob E Lemieux. Analysis of 6.4 million sars-cov-2 genomes identifies mutations associated with fitness. *Science (New York, N.Y.)*, 376(6599):1327–1332, 2022. doi: 10.1126/science.abm1208.

Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C Lawrence Zitnick, Jerry Ma, and Rob Fergus. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118, 2021.

Valeria Ruscio, Umberto Nanni, and Fabrizio Silvestri. What are you sinking? a geometric approach on attention sink. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 123482–123521. Curran Associates, Inc., 2025. doi: 10.52202/085713-4116. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/b2c4b7d34b3d96b9dc12f7bce424b7ae-Paper-Conference.pdf.

C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27 (3):379–423, 1948. doi: 10.1002/j.1538-7305.1948.tb01338.x.

Sristy Shikha, Mukesh Kumar Jogi, Ruchika Jha, Rana Amit Kumar, Tathagat Sah, Pushpendra Singh, Ritu Sagar, Anuj Kumar, Robin Marwal, Kalaiarasan Ponnusamy, Subhash Mohan Agarwal, R. Suresh Kumar, Nazneen Arif, Mausumi Bharadwaj, Shalini Singh, and Pramod Kumar. Genome sequencing of sars-cov-2 omicron variants in delhi reveals alterations in immunogenic regions in spike glycoprotein. *Frontiers in Immunology*, Volume 14 - 2023, 2023. ISSN 1664-3224. doi: 10.3389/fimmu.2023.1209513. URL https://www.frontiersin.org/journals/immunology/articles/10.3389/fimmu.2023.1209513.

Elana Simon and James Zou. Interplm: Discovering interpretable features in protein language models via sparse autoencoders. *arXiv preprint arXiv:2412.12101*, 2024.

Tomokazu Tamura, Takashi Irie, Sayaka Deguchi, Hisano Yajima, Masumi Tsuda, Hesham Nasser, Keita Mizuma, Arnon Plianchaisuk, Saori Suzuki, Keiya Uriu, Mst Monira Begum, Ryo Shimizu, Michael Jonathan, Rigel Suzuki, Takashi Kondo, Hayato Ito, Akifumi Kamiyama, Kumiko Yoshimatsu, Maya Shofa, Rina Hashimoto, Yuki Anraku, Kanako Terakado Kimura, Shunsuke Kita, Jiei Sasaki, Kaori Sasaki-Tabata, Katsumi Maenaka, Naganori Nao, Lei Wang, Yoshitaka Oda, Hirofumi Sawa, Ryoko Kawabata, Yukio Watanabe, Ayaka Sakamoto, Naoko Yasuhara, Tateki Suzuki, Yukari Nakajima, Zannatul Ferdous, Kenji Shishido, Yuka Mugita, Otowa Takahashi, Kimiko Ichihara, Yu Kaku, Naoko Misawa, Ziyi Guo, Alfredo Hinay, Yusuke Kosugi, Shigeru Fujita, Jarel M. Tolentino, Luo Chen, Lin Pan, Mai Suganami, Mika Chiba, Ryo Yoshimura, Kyoko Yasuda, Keiko Iida, Naomi Ohsumi, Adam P. Strange, Yuki Shibatani, Tomoko Nishiuchi, Shiho Tanaka, Olivia Putri, Gustav Joas, Yoonjin Kim, Daichi Yamasoba, Kazuhisa Yoshimura, Kenji Sadamasu, Mami Nagashima, Hiroyuki Asakura, Isao Yoshida, So Nakagawa, Akifumi Takaori-Kondo, Kotaro Shirakawa, Kayoko Nagata, Ryosuke Nomura, Yoshihito Horisawa, Yusuke Tashiro, Yugo Kawai, Takamasa Ueno, Chihiro Motozono, Mako Toyoda, Terumasa Ikeda, Akatsuki Saito, Keita Matsuno, Jumpei Ito, Shinya Tanaka, Kei Sato, Takao Hashiguchi,

Kazuo Takayama, Takasuke Fukuhara, and The Genotype to Phenotype Japan (G2P-Japan) Consortium. Virological characteristics of the sars-cov-2 omicron xbb.1.5 variant. *Nature Communications*, 15(1):1176, Feb 2024. ISSN 2041-1723. doi: 10.1038/s41467-024-45274-3. URL https://doi.org/10.1038/s41467-024-45274-3.

Darin Tsui, Kunal Talreja, Daniel Saeedi, and Amirali Aghazadeh. Protein circuit tracing via cross-layer transcoders, 2026. URL https://arxiv.org/abs/2602.12026.

Robert Verkuil, Ori Kabeli, Yilun Du, Basile I. M. Wicky, Lukas F. Milles, Justas Dauparas, David Baker, Sergey Ovchinnikov, Tom Sercu, and Alexander Rives. Language models generalize beyond natural proteins. *bioRxiv*, 2022. doi: 10.1101/2022.12.21.521521. URL https://www.biorxiv.org/content/early/2022/12/22/2022.12.21.521521.

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small, 2022. URL https://arxiv.org/abs/2211.00593.

Qian Wang, Sho Iketani, Zhiteng Li, Liyuan Liu, Yicheng Guo, Yiming Huang, Anthony D Bowen, Michael Liu, Maple Wang, Jian Yu, Riccardo Valdez, Adam S Lauring, Zizhang Sheng, Harris H Wang, Aubree Gordon, Lihong Liu, and David D Ho. Alarming antibody evasion properties of rising sars-cov-2 bq and xbb subvariants. *Cell*, 186(2):279–286.e8, 2023. doi: 10.1016/j.cell.2022.12.018.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 21875–21895, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/5e5fd18f863cbe6d8ae392a93fd271c9-Paper-Conference.pdf.

Yijia Xiao, Edward Sun, Yiqiao Jin, Qifan Wang, and Wei Wang. ProteinGPT: Multimodal LLM for protein property prediction and structure understanding. In *ICLR 2025 Workshop on Machine Learning for Genomics Explorations*, 2025. URL https://openreview.net/forum?id=qWlqfjGVWX.

Zayd Muhammad Kawakibi Zuhri, Erland Hilman Fuadi, and Alham Fikri Aji. Softpick: No attention sink, no massive activations with rectified softmax. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 18179–18213, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.905. URL https://aclanthology.org/2026.findings-acl.905/.

A Appendix

A.1 Model schematics and "fitness" logit

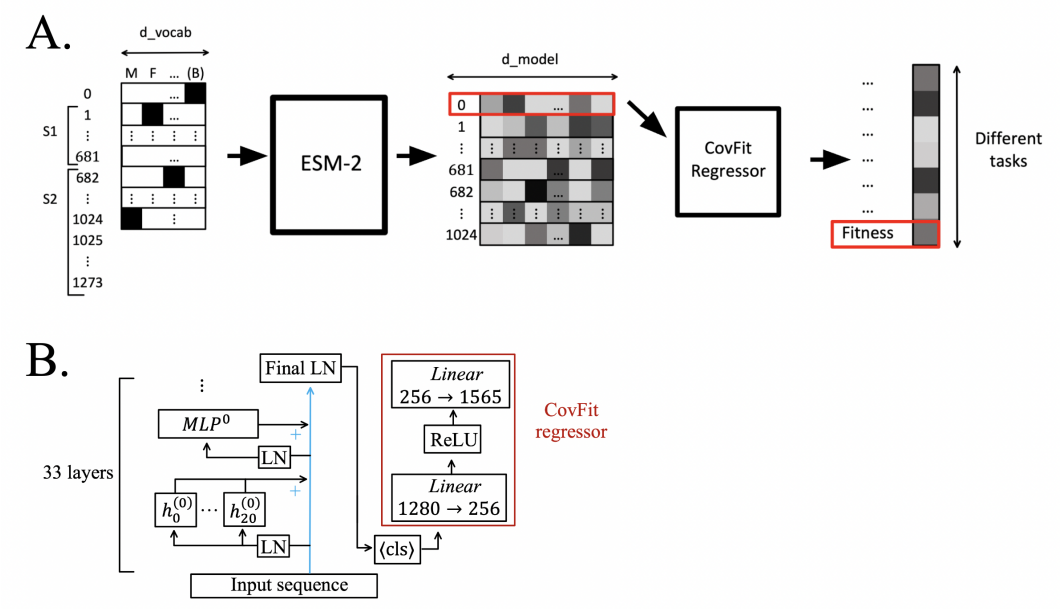

Figure 6: A. The pipeline of how a tokenized spike protein sequence turns into a fitness prediction: Tokenized sequences are passed through a fine-tuned ESM-2 model. From there the ⟨cls⟩-token is passed into the CovFit-regressor to generate predictions for a variety of biological tasks. We choose fitness (specifically the predicted fitness in USA) as our logit of interest. B. The model architecture. $\text{ESM}_{\text{coronaviridae}}$ is a BERT-style transformer model with 33 layers, with each layer having 20 attention heads, and uses RoPE embeddings. The red component is the CovFit-regressor, a 2-layer neural network which takes as input the ⟨cls⟩-token and and returns predictions for 1565 different biological tasts. The full architecture in B is the architecture of CovFit.

We also explain exactly what the "fitness" logit that CovFit predicts refers to, as this may be unclear to both biological and computational audiences. As described in Ito et al. (2025), fitness refers to the relative effective reproduction number between variants of SARS-CoV-2. For a single variant v, its effective reproduction number $R_{t,v}$ at time t is the expected number of new infections arising from a single individual infected with the virus v at time t. Then, for any two variant v_1, v_2, the relative effective reproduction number is defined:

$$\frac{R_{t,v_1}}{R_{t,v_2}}$$

and Ito et al. (2025) make the assumption that the relative effective reproduction number between variants stays constant with respect to time.

The "ground-truth" values for fitness were then assigned to each sequence in CovFit's training data by choosing a reference sequence and defining the fitness of all other sequences to be their relative effective reproduction number with respect to the reference sequence. These relative effective reproduction numbers were estimated using a multinomial logistic regression model (which is a common practice genomic surveillance, see Obermeyer et al. (2022); Davies et al. (2021)) The chosen reference sequence was the major genotype of the BA.5 lineage (equivalent to BA.5.2.1 in the PANGO lineage), and Ito et al. (2025) provide a justification of why this specific sequence was chosen in their methods section.

Some readers may be confused why CovFit was introduced when "fitness" as is defined by Ito et al. (2025) can be inferred using the aforementioned multinomial logistic regression model. CovFit is a generative model which was created to model the distribution of SARS-CoV-2 sequences. In

contrast to the multinomial logistic model, which requires many samples of sequence data before providing an estimate, CovFit is able to provide estimates of fitness from a single spike protein sequence.

A.2 CovFit-regressor is Linear model

For this experiment, we split all of our (unique) sequences up into a 60:20:20 train:valid:test split. For each of our splits, we cached the ⟨cls⟩-tokens and the predicted fitness outputs. Then, on our training split, we fit a linear regression to predict the fitness output from the ⟨cls⟩-token embedding. We plot the fitted linear regression's predictions for fitness against CovFit-regressor's predictions for fitness on each of our splits.

The correlations for all splits are almost exactly 1, which demonstrates that the CovFit-regressor is essentially taking a linear combination of the elements of the ⟨cls⟩-token embedding to produce the final fitness output. Therefore, it plays no role in extracting information from the sequence in contrast to the attention heads and MLP blocks of the transformer component of CovFit.

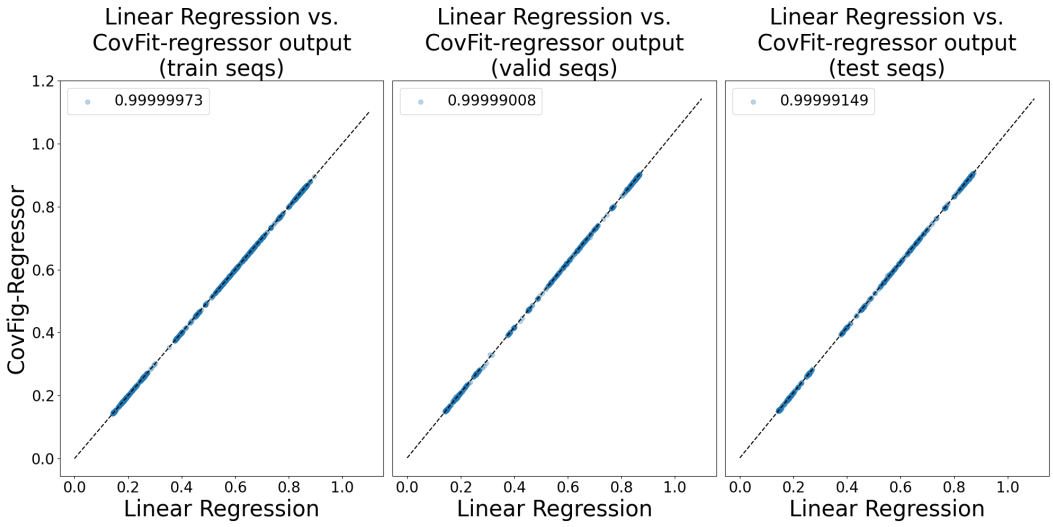

Figure 7: Given the ⟨cls⟩-token embedding for each sequence, we create a 60:20:20 train:valid:test split and fit a linear regression to predict output of CovFit-regressor on the train ⟨cls⟩-token embeddings. The correlations for each split are almost exactly 1.

Table 3: Number of sequences per mutation dataset

Mutations	# sequences
E484A	1804
F456L	267
G339H	2139
K417N	2081
L455F	3547
P681H	1131
Q493R	2361
R346T	113
S486P	86
V445P	2938

A.3 "WILD-TYPE", "MUTATED", AND "MASKED" DATASET CURATION PIPELINE

For any mutation aa_1 Site aa_2, we create our 3 datasets as follows: We first choose "wild-type" sequences which have aa_1 at the mutated site from SARS-CoV-2 lineages at or before the mutation appeared. We then create identical sequences and change aa_1 to aa_2 at the mutated site and these are our "mutated" sequences. For "masked" sequences we do a similar procedure but replace aa_1 with $\langle \text{mask} \rangle$.

We synthetically create mutated sequences in order to have a larger set of sequences with which to perform our analysis. We believe that, although we are synthetically creating mutated sequences, since we are choosing the wild-type sequence to be from the appropriate lineages (and not, for examples, choosing a sequence from after the mutation occurred to be included in our "wild-type" dataset), our datasets are still biologically meaningful.

We provide the number of sequences in each of our mutations datasets in the table below (note that, for each mutation, each of the "wild-type," "mutated," and "masked" datasets contain the same number of sequences, and that is the number that appears in the table entries):

A.4 DISTRIBUTION OF FITNESS DIFFERENCES

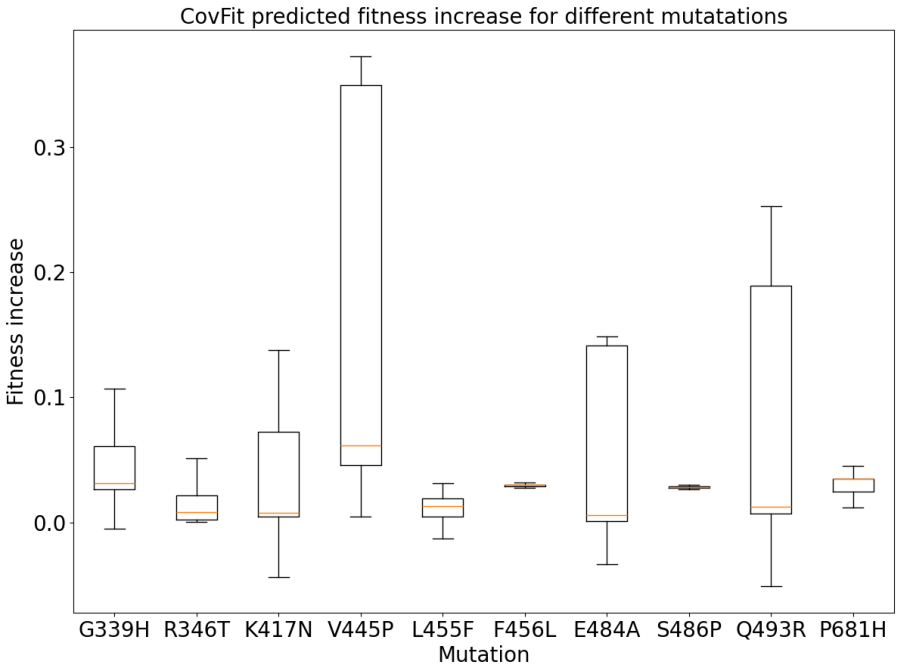

Figure 8: Distribution of predicted fitness difference between sequences in each of the "mutated" and "wild-type" datasets

A.5 ACTIVATION PATCHING RESULTS FOR MUTATIONS

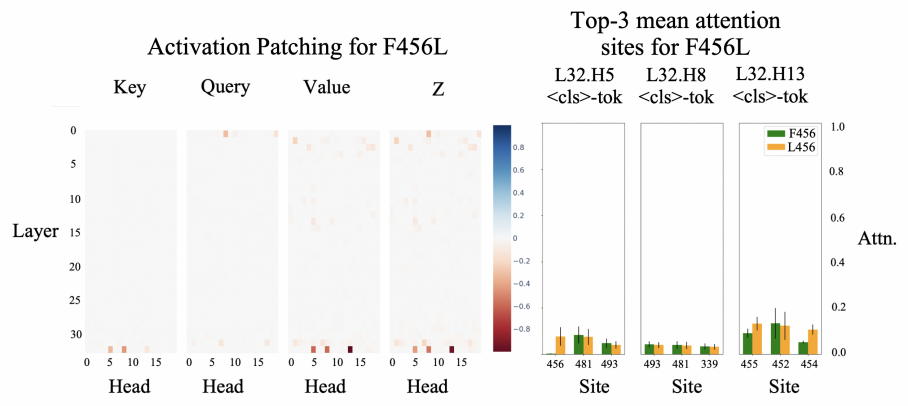

Figure 9: Activation patching results for F456L

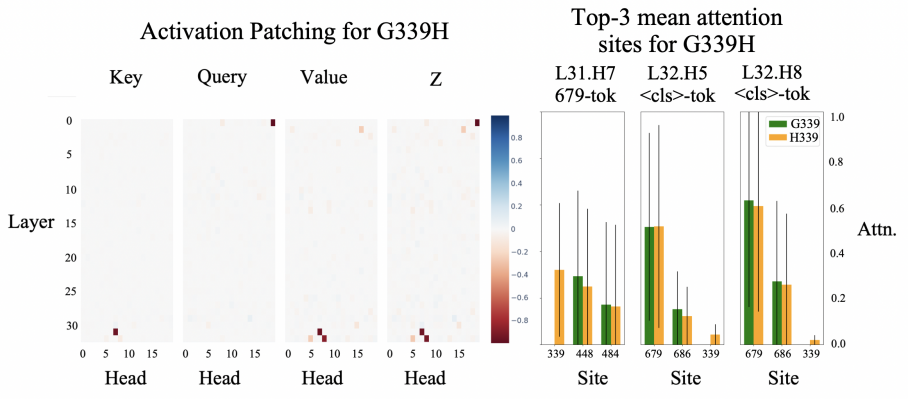

Figure 10: Activation patching results for G339H

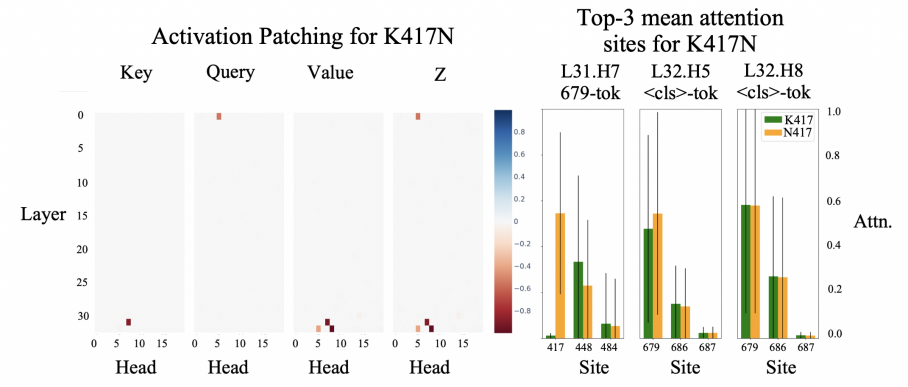

Figure 11: Activation patching results for K417N

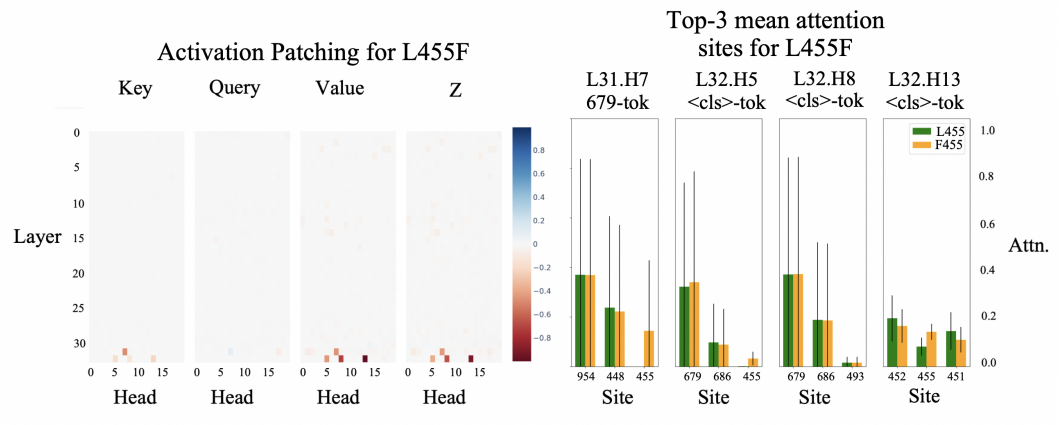

Figure 12: Activation patching results for L455F

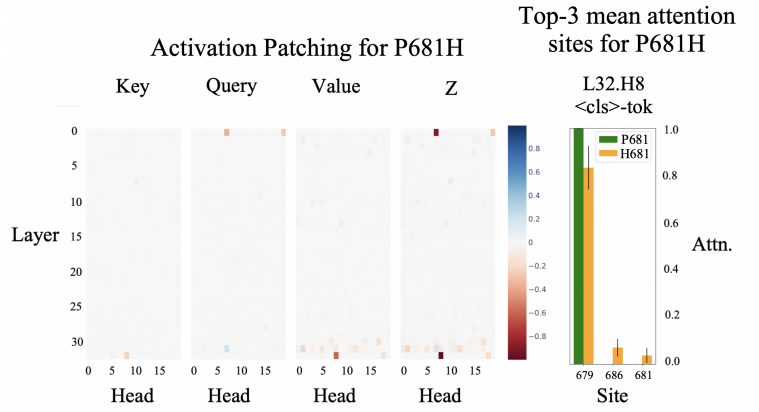

Figure 13: Activation patching results for P681H

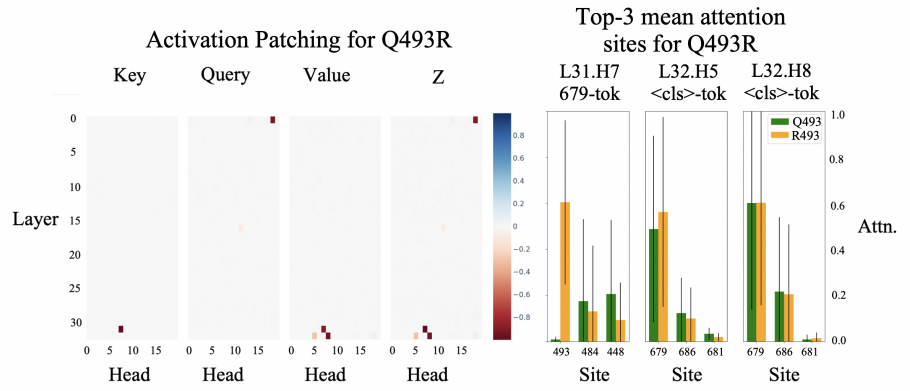

Figure 14: Activation patching results for Q493R

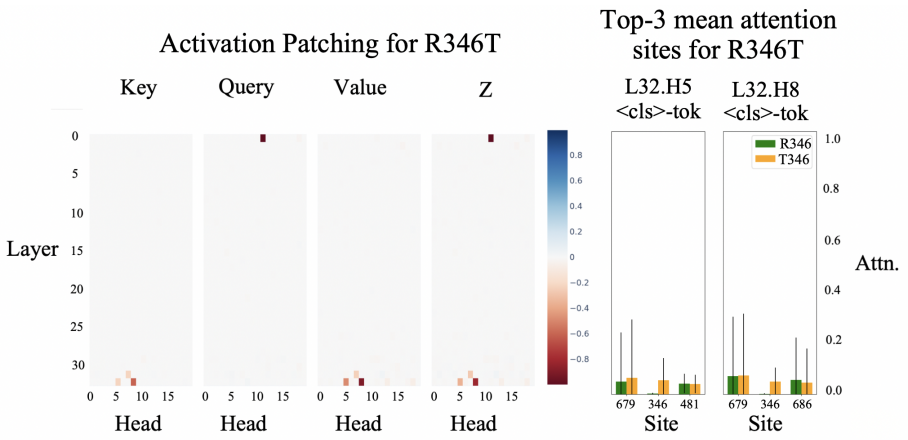

Figure 15: Activation patching results for R346T

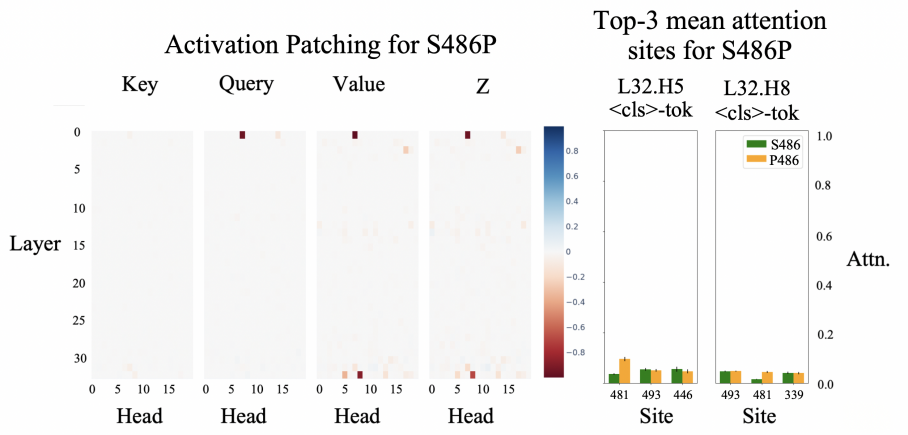

Figure 16: Activation patching results for S486P

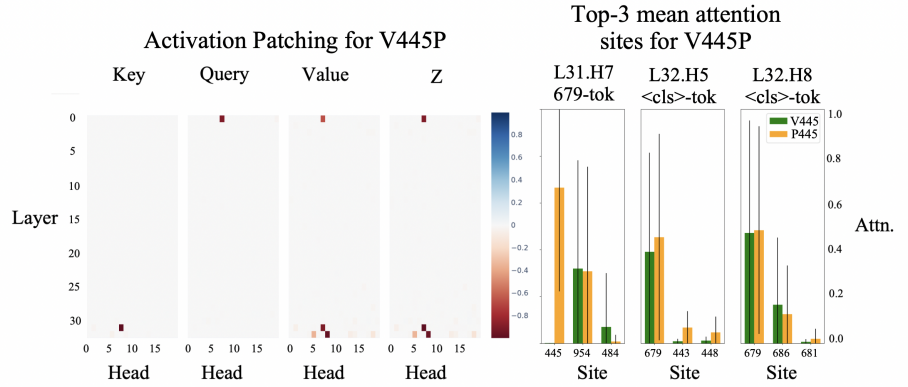

Figure 17: Activation patching results for V445

A.6 PATH-PATCHING TO SHOW L31.H7 COMPOSES WITH LAYER 32 HEADS

We absolute path patch to MLP 31 and find that L31.H7 has a direct causal impact on MLP 31 (for the mutations where L31.H7 has a large downstream impact on predicted fitness when perturbed).

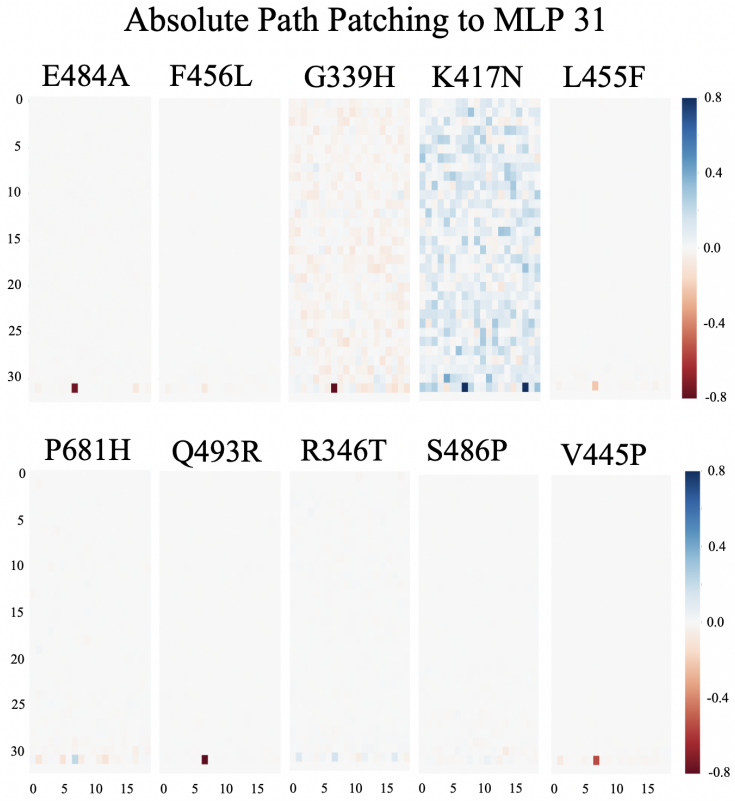

Figure 18: Absolute path patching to MLP 31 for each mutation. L31.H7 can can be seen to have an effect on MLP 31 when it's perturbed for mutations E484A, G339H, K417N, L455F, Q493R, and V445P

Since MLP 31 directly affects the layer 32 attention heads, this shows that L31.H7 indirectly composes with the layer 32 attention heads, mediated by MLP 31.

A.7 Full Breakdown of Attention Head roles

Below, we provide a categorization of each attention head included in our L455F circuit attention heads. For heads that exclusively look a certain number of tokens ahead/behind, we compute a "mover" score which is computed by taking the average attention score on the off-diagonal corresponding to the number of tokens that the head looks ahead/behind. We find that most early layer attention heads can be categorized in this way due to their high "mover" scores. We have the following notation: "N-succ head" := an attn. head that always looks N tokens ahead, "N-prev head" := an attn. head that always looks N tokens previous, "curr tok head" := an attn. head that always looks at the current token, "compositional head" := if attn. head shows compositional behavior at key sites (i.e. if the attn. head has an increase in attention score at site 455 or site on 455F sequences vs. 455L sequences). In the main manuscript, we identified heads that strongly affect the ⟨cls⟩-mover heads as "aa455 heads;" those heads are also labeled here with the "(aa455)" prefix. For any heads that do not fall into these categories, we provide a qualitative description of their behavior at site 455 and site 679. We do not provide mover scores for aa455 and ⟨cls⟩-mover heads.

Table 4: Categorization of heads involved in L455F circuit

Attn. head	Description	Mover score (if applicable)
0.*	Layer 0 heads (crucial for model to function)	N/A
1.15	Primarly 2-prev head, also 1-prev and 3-prev head	2-prev: 0.52 1-prev: 0.20 3-prev: 0.12
2.4	1-succ head	1-succ: 0.94
2.17	2-prev head	2-prev: 0.89
12.18	partially 1-succ head	1-succ: 0.52
13.5	primarily 2-prev head	2-prev: 0.68
13.9	3-succ head	3-succ: 0.82
14.6	Primarily 3-prev head	3-prev: 0.68
31.1	(aa455) 2-succ. head at site 679, examining many toks at site 455	N/A
31.2	(aa455) comp. head	N/A
31.3	(aa455) 1-succ head	1-succ: 0.85
31.7	(aa455) comp. head	N/A
31.8	(aa455) comp. head,	N/A
31.9	(aa455) partially 3-prev head	3-prev: 0.52
31.12	(aa455) comp. head	N/A
31.13	(aa455) 1-prev head	1-prev: 0.84
31.14	(aa455) primarily 1-succ head	1-succ: 0.68
31.15	(aa455) partially 1-prev head	1-prev: 0.46
31.17	(aa455) Attends to end token at site 679 and attends to ⟨cls⟩-token at site 455	N/A
32.5, 32.8, 32.13	⟨cls⟩-mover heads	N/A

A.8 ABSOLUTE PATH-PATCHING

Observe that the formula for obtaining the sequence embedding from the transformer is:

$$\text{seq. embed } = LN\left(x^{(0)} + \sum_{\ell=\text{layers}}\left(\left(\sum_{\text{attn. heads } h_i} h_i^{(\ell)}(LN(x^{\ell}))\right) + \text{MLP}^{\ell}(LN(x^{\ell}_{\text{attn.}}))\right)\right) \quad (1)$$

where h_i^{ℓ} denotes attention head i at layer ℓ, $x^{(\ell)}$ denotes the transformer block output at layer ℓ, $x^{\ell}_{attn.}$ denotes the output of the layer ℓ attention block, and LN denotes a layernorm. Note that $\langle$cls$\rangle$-token $=$ seq. embedding$[0, :]$ and fitness prediction $=$ CovFit-regressor($\langle$cls$\rangle$-token), i.e. the $\langle$cls$\rangle$-token embedding is the first row of the sequence embedding and this embedding is passed into the CovFit-regressor to get the final fitness output.

Given our dataset of "mutated" (clean) and "wild-type" (counterfactual) sequences, we obtain the causal impact of a specific component $comp$ on the final fitness output by replacing the corresponding term for $comp$ in the above equation (when run on clean sequences) with its output on the counterfactual sequences. We take the $\langle$cls$\rangle$-token and pass it into the CovFit-regressor as normal and then compare the difference in predicted fitness using the following metric:

$$\text{patch score}(comp) = \frac{\text{patched-fitness} - \text{counterfactual-fitness}}{\text{counterfactual-fitness} - \text{clean-fitness}}$$

where "patched-fitness" is the new predicted fitness after patching $comp$ and "counterfactual-fitness", "clean-fitness" are the respective (mean) fitness predictions on the counterfactual and clean datasets. A score of 0 means that patching to $comp$ has no effect on the fitness difference while a score of -1 means that patching to $comp$ completely removes the mutation effect.

To measure the causal impact of one component $comp_1$ on anoter component $comp_2$, we employ a procedure similar to Wang et al. (2022). First, observe that the output of $comp_2$ can come from taking a subset of the terms in equation (1). To absolute path patch from $comp_1$ to $comp_2$, in the summation formula for the output of $comp_2$, we replace the output of $comp_1$ to its output on the counterfactual dataset i.e. the "wild-type" sequences. This is equivalent to only patching the output of $comp_1$ to $comp_2$ and freezing all intermediate components between the two components. Then, we recompute all components downstream of $comp_2$ and record the difference in predicted fitness. We use the same metric patch score$(\cdot)$ as described above to obtain a score for the causal impact of $comp_1$ on $comp_2$.