Thrifty wide-context models of B cell receptor somatic hypermutation

Kevin Sung[1], **Mackenzie M. Johnson**[1], **Will Dumm**[1], **Noah Simon**[2], **Hugh Haddox**[1], **Julia Fukuyama**[3], **Frederick A Matsen IV**[4,5,6*]

***For correspondence:** matsen@fredhutch.org (FAM)

[1]Computational Biology Program, Fred Hutchinson Cancer Center, Seattle, Washington, USA; [2]Department of Biostatistics, University of Washington, Seattle, Washington, USA; [3]Department of Statistics, Indiana University, Bloomington, Indiana, USA; [4]Howard Hughes Medical Institute, Seattle, Washington, USA; [5]Department of Genome Sciences, University of Washington, Seattle, Washington, USA ; [6]Department of Statistics, University of Washington, Seattle, Washington, USA

Abstract Somatic hypermutation (SHM) is the diversity-generating process in antibody affinity maturation. Probabilistic models of SHM are needed for analyzing rare mutations, for understanding the selective forces guiding affinity maturation, and for understanding the underlying biochemical process. High throughput data offers the potential to develop and fit models of SHM on relevant data sets. In this paper we model SHM using modern frameworks. We are motivated by recent work suggesting the importance of a wider context for SHM, however, assigning an independent rate to each k-mer leads to an exponential proliferation of parameters. Thus, using convolutions on 3-mer embeddings, we develop "thrifty" models of SHM of various sizes; these can have fewer free parameters than a 5-mer model and yet have a significantly wider context. These offer a slight performance improvement over a 5-mer model, and other modern model elaborations worsen performance. We also find that a per-site effect is not necessary to explain SHM patterns given nucleotide context. Also, the two current methods for fitting an SHM model — on out-of-frame sequence data and on synonymous mutations — produce significantly different results, and augmenting out-of-frame data with synonymous mutations does not aid out-of-sample performance.

Introduction

Antibodies are an essential component of the adaptive immune response. They are secreted by B cells, and when displayed on the surface of B cells are called B cell receptors. When stimulated by antigen binding, B cells undergo a process called "affinity maturation" which involves mutation and selection. The mutation process happens at a very high rate relative to the rate of normal somatic mutation and is called somatic hypermutation (SHM). As such, it is an essential part of the adaptive immune response. It is generated by a complex collection of interacting pathways of DNA damage and error-prone repair, which have been elucidated through decades of research (***Wagner and Neuberger, 1996***; ***Teng and Papavasiliou, 2007***; ***Methot and Di Noia, 2017***; ***Pilzecker and Jacobs, 2019***). These pathways lead to a very non-uniform distribution of mutations.

Furthermore, the mutation biases are predictable from local sequence context. Many papers have investigated predictors of mutation rates from molecular sequence, including early work establishing the biases (***Dunn-Walters et al., 1998***; ***Rogozin and Kolchanov, 1992***; ***Rogozin and Diaz, 2004***), to parametric models estimating the mutability based on local sequence "motif", or se-

quence neighborhood around a focal base (***Yaari et al., 2013***; ***Elhanati et al., 2015***; ***Cui et al., 2016***; ***Feng et al., 2019***; ***Fisher et al., 2025***).

Such models are important when predicting the probability of amino acid changes in affinity maturation, e.g. for understanding the prospects of selecting such mutations in for reverse vaccinology (***Wiehe et al., 2018***; ***Martin Beem et al., 2023***) or in computing a model of natural selection on antibodies (***McCoy et al., 2015***; ***Hoehn et al., 2017***, ***2019***).

The most popular models for somatic hypermutation are the S5F 5-mer model and its variants (***Yaari et al., 2013***; ***Cui et al., 2016***). They have shown their worth for over a decade now, including tasks such as predicting the probability of mutations to mature broadly neutralizing antibodies against HIV (***Wiehe et al., 2018***, ***2022***). However, biological considerations suggest that a wider context should be considered.

Indeed, the consensus view of SHM requires processes such as patch removal around an AID-induced lesion (***Pilzecker and Jacobs, 2019***) and error-prone repair. Thus, for example, the presence of an AID hotspot several bases away may influence the probability of a mutation at a focal base. More recently, mesoscale-level sequence effects on AID deamination potentially deriving from local DNA sequence flexibility have been discovered (***Wang et al., 2023***). In addition, other work has found that position in the sequence can influence SHM (***Cohen et al., 2011***; ***Zhou and Kleinstein, 2020***; ***Spisak et al., 2020***).

This begs the question of how one could use more complex models to predict somatic hypermutation. 7-mer models, which have 3 flanking bases on either side of the focal base, have been used (***Elhanati et al., 2015***; ***Marcou et al., 2018***). However, one cannot simply increase the size of the k-mer model indefinitely because the number of parameters grows exponentially with the size of the k-mer. More recent models of somatic hypermutation include position-specific terms (***Spisak et al., 2020***) and context models of size up to 21 parameterized by a convolutional neural network (***Tang et al., 2022***). In other contexts, models based on the transformer architecture (***Vaswani et al., 2017***) have shown great success, raising the question of if such an architecture could be used here.

In this paper we develop new models using modern frameworks, and provide a comprehensive evaluation of the models. We especially focus on the development of parameter-efficient convolutional neural networks of various sizes, which we call "thrifty" models. These models have wide nucleotide context yet can have fewer parameters than a 5-mer model, while providing slightly better performance on metrics in train and test time. On the other hand, we find that elaborations such as a per-site rate and transformer only harm out-of-sample performance. We also find a clear difference between training models to predict well on out-of-frame data, compared to training models to predict well on synonymous mutations. To make these models useful for the community, we have released an open-source Python package https://github.com/matsengrp/netam with pretrained models and a simple API. Our analysis for this paper is reproducible via https://github.com/matsengrp/netam-experiments-1.

Results

Overview of data preparation and objective

We will begin with an overview of our models and data. Full details are provided in the Methods.

Our objective in this project is to predict the probability of observed somatic hypermutation in a child sequence relative to a parent sequence. We follow previous work (***Spisak et al., 2020***) in overall goal and data setup. Specifically, we predict mutations in BCR sequences that are out-of-frame, i.e. such that the sequence cannot code for a productive receptor. Because the data is out-of-frame, this means that the sequences under consideration are less likely to have undergone selective pressure in the germinal centers, and instead provides more information about the SHM process. We also provide more relevant parent sequences and predict finer-scale events by using phylogenetic reconstruction and ancestral sequence inference on sequences clustered into

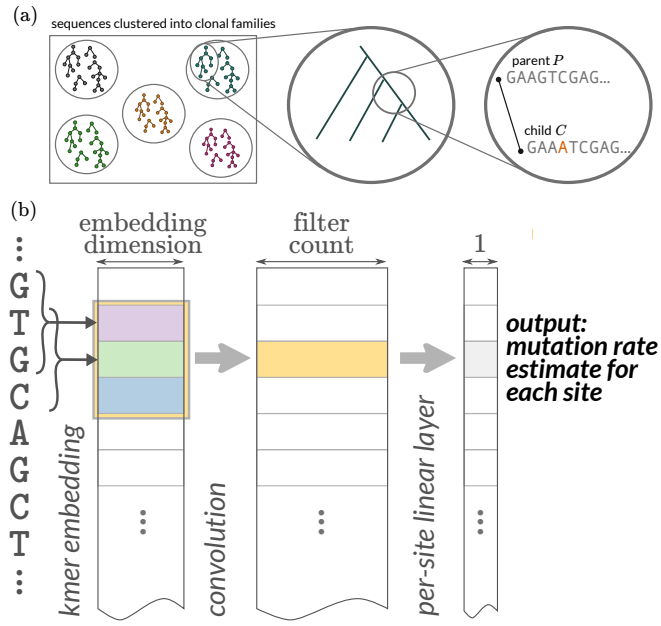

Figure 1. (a) Overview of data processing and objective. Out-of-frame sequences are clustered into clonal families. Trees are built on clonal families and then ancestral sequences are reconstructed using a simple sequence model. The prediction task is to predict the location and identity of mutations of child sequences given parent sequences. (b) Strategy for "thrifty" CNNs with relatively few parameters. We use a trainable embedding of each 3-mer into a space; downstream convolutions happen on sequential collections of these embeddings. The "width" of the k-mer model is determined by the size of the convolutional kernel, which in this cartoon is 3. This would give us effectively a 5-mer model because the 3-mer model adds one base on either side of a convolution of length 3. For the sake of simplicity, the probability distribution of the new base conditioned on there being a substitution (which we call the conditional substitution probability or CSP) is not shown. The CSP output can emerge in several ways (***Figure 1—figure Supplement 1***).

Figure 1—figure supplement 1. Strategies for estimating both per-site rate and CSPs.

clonal families (***Figure 1***a). We split the tree with ancestral sequences into pairs of parent and child sequences, which we call parent-child pairs. We also experiment with using synonymous mutation data by masking mutations from the loss function that are not synonymous (below; details in Methods).

In all models, the distribution of mutations at a particular site is assumed to be independent of mutations at all other sites (but not independent of context). We follow many authors starting from (***Yaari et al., 2013***) in estimating a per-site rate, as well as a per-site probability distribution among the non-identical bases describing the base selected in the event of a mutation. We will call this the conditional substitution probability (CSP). For each site i, we assume that the mutation process is an Exponential waiting time process with rate λ_i. Once the mutation occurs, we assume that the base is selected according to a categorical distribution with probabilities $\mathbf{p}_i$. Similar assumptions have been made previously (***Rosset, 2007***; ***Levinstein Hallak et al., 2018***; ***Spisak et al., 2020***; ***Levinstein Hallak and Rosset, 2022***). To accommodate evolutionary time in our model, we include *offsets* in our exponential model—if t is a branch length parameter for a sequence pair we use parameter $\tilde{\lambda} = t\lambda$ for model inference so that the model is able to learn λ irrespective of evolutionary time on a particular branch. This parameter t is frequently the normalized mutation count (as in ***Spisak et al., 2020***) but can be optimized as part of a joint optimization.

We used two data sets, that we will call the `briney` and `tang` data sets. The `briney` data (***Briney et al., 2019***) consists of samples from 9 individuals, but 2 of these samples resulted in many more sequences than the rest. Thus, we will use a test-train split in which these 2 samples form the training data and the other 7 samples form the testing data. We acknowledge the important work the ***Spisak et al.*** (***2020***) team did in processing the `briney` data. The `tang` data (***Vergani et al., 2017***;

Table 1. Selected model shapes and dropout probabilities. The release name of the model is the name of the trained model released in the GitHub repository. The paper name is the name of the model used in this manuscript, which describes more about its architecture. "Kernel": the size of the convolutional kernel used in the model. "Embed": the size of the embedding used for each 3-mer. Because there is one additional base on either side of a 3-mer, a model with kernel size 9 is effectively an 11-mer model, and a model with kernel size 11 is effectively a 13-mer model. The "Medium" and "Large" labels in the paper name designate the settings for Kernel, Embed, Filters, and Dropout.

release name: paper name	Kernel	Embed	Filters	Dropout	Params
ThriftyHumV0.2-20: CNN Joined Large	11	7	19	0.3	2,057
5mer	-	-	-	-	3,077
Spisak	-	-	-	-	3,576
ThriftyHumV0.2-45: CNN Indep Medium	9	7	16	0.2	4,539
ThriftyHumV0.2-59: CNN Indep Large	11	7	19	0.3	5,931

Tang et al., 2020) will be a further test set. Details on data processing appear in the Methods. In the Methods, we describe our attempts to find additional data sets.

Models

We use the following strategy to combine the predictive power of local-context models without having the parameter penalty (***Figure 1***b). Each 3-mer is mapped into a embedding space of a fixed dimension, and these embedding locations are trainable parameters of the model. The idea is that the embedding abstracts some SHM-relevant characteristics of that 3-mer. Each sequence is then represented as a matrix with (sequence length) rows and (embedding dimension) columns. We then apply convolutional filters to these matrices, with taller convolutional filters effectively increasing the context of the model. For example, a kernel size of 11 gives effectively a 13-mer model (because of the additional base on either side of the 3-mer). We then apply a simple linear layer to the result of this step in order to get a mutation rate estimate for each site.

As described above, this class of models predicts both the per-site rate of SHM as well as the probability of alternate bases after mutation (called the CSP as above). We make these two model outputs in three ways (***Figure 1—figure Supplement 1***): they can share everything except for the final layer ("joined" model), or they can share the embedding layer ("hybrid" model), or they can be estimated separately ("independent" model). A key difference with a full k-mer model is that when we increase the size of the kernel, the number of parameters increases linearly, not exponentially. In this way, the thriftiest well-performing model is effectively a 13-mer model with fewer parameters than a 5-mer model, however one can scale these models among a variety of dimensions (***Table 1***).

We implemented our models in PyTorch (***Paszke et al., 2019***). Because of the small size of these models, they are fast to train and use. The hyperparameters for the models (Table 1) were selected with a run of Optuna (***Akiba et al., 2019***) early in the project and then fixed. Further optimization was not pursued because of the limited performance differences between the existing models.

Thrifty CNNs give a modest performance improvement

In order to evaluate our proposed methods and compare to previous work, we first characterized the models in terms of predictive performance using AUROC, AUPRC, R-precision, and substitution accuracy. AUROC, the area under the ROC curve, can be interpreted as the probability that the model correctly identifies sites that mutate as having higher mutability than those that do not. In fact, if one randomly selects a positive-negative pair, the AUROC is the probability that the positive example is assigned a higher probability than the negative example. However, this measure is sensitive to class imbalance, and we are in the imbalanced setting here because mutations are relatively rare. AUPRC, the area under the precision-recall curve, provides an alternative that is less

sensitive to class imbalance effects (***Saito and Rehmsmeier, 2015***; ***Ozenne et al., 2015***) because the precision is the fraction of positive predictions that are true positives. R-precision gives a sense of how accurate the model is among sites that are most mutable. Specifically, if a given pair of parent and child sequences had R mutations, R-precision is the precision of the predictions of mutability at the R sites that are ranked as being most mutable. To evaluate performance at predicting per-base substitution probabilities (given a mutation occurred), we report substitution accuracy: how frequently is the predicted-most-likely base the one to which a site mutates?

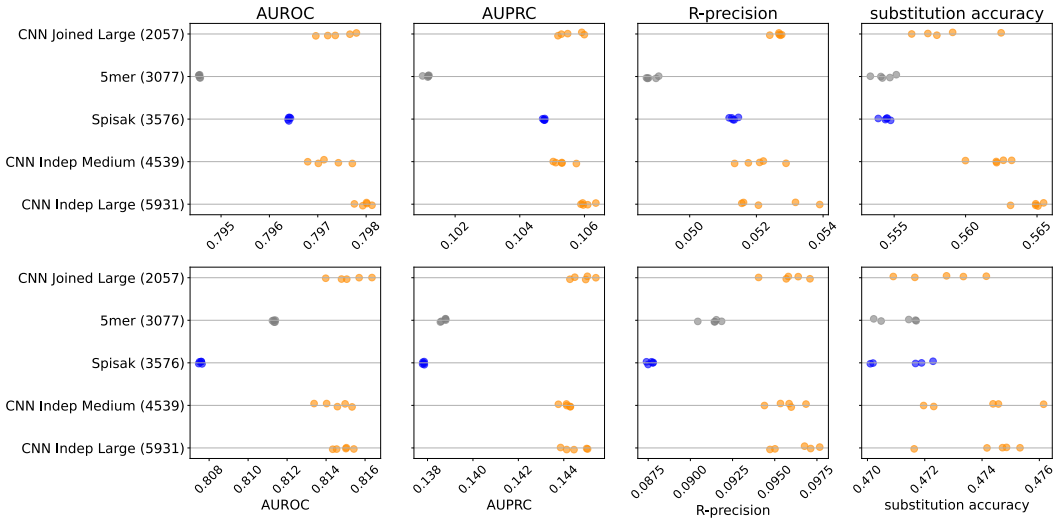

Figure 2. Model predictive performance on held-out samples: held-out individuals from the `briney` data (upper row) and on a separate sequencing experiment (`tang` data, lower row). Note that the two rows use different x-axis scales. Integer in parentheses indicates the number of parameters of the model. Each model has multiple points, each corresponding to an independent model training. This is a subset of the models for clarity; see ***Figure 2—figure Supplement 1*** for all models.

Figure 2—figure supplement 1. Performance results for all the models.

Figure 2—figure supplement 2. Agreement between the original "shmoof" model of ***Spisak et al.*** (***2020***) and our "reshmoof" reimplementation.

Figure 2—figure supplement 3. Performance comparison including S5F model with original coefficients.

We found that the thrifty CNN models gave a modest performance improvement for these predictive metrics compared to existing models (***Figure 2***, ***Figure 2—figure Supplement 1***). Specifically, we compared to a 5-mer model trained in exactly the same way, as well as a reimplementation of the model of ***Spisak et al.*** (***2020***). We confirmed that our reimplementation infers very similar parameters to the previous implementation, although we add a slight regularization to avoid some aspects of the original model fit that appear to be artifacts (***Figure 2—figure Supplement 2***). Because the ***Spisak et al.*** (***2020***) model fits a per-site rate, and the `briney` data does not have full sequence coverage, we limited all evaluation to a region well covered by the `briney` data: positions 80 to 319, inclusive.

We were surprised to find that all metrics except substitution accuracy were better on data from a distinct sequencing experiment (`tang` data) than on held-out samples from the same sequencing experiment (`briney` data). We attribute this to there being a difference in sequencing error between the two experiments. The `briney` data allowed sequences with only a single UMI representative (***Spisak et al., 2020***) while the `tang` data required at least two sequences per UMI (***Tang et al., 2020***). If our model is successfully learning substitution probabilities due to SHM rather than the sequencing error, we would expect our model to under-estimate the number of substitutions at positions with a low probability of substitution in the `briney` data (which we expect to have more sequencing error) but not in the `tang` data (which we expect to have less sequencing error). This

is in fact what we see in our characterization of model fit below. We were also surprised to find that the per-site rate did not seem to help the 5-mer model on held-out data, despite the results of ***Spisak et al.*** (***2020***); see Discussion.

We did not see a substantial performance improvement by increasing the number of parameters of the CNNs. Recall that our models differ in terms of how much they share between predicting the position of mutations and predicting their base identity (***Figure 1—figure Supplement 1***). Although the "Large Indep" model that has the most flexibility may do a slightly better job with held-out samples from the same experiment in terms of substitution accuracy, it doesn't appear any better when considering data from another experiment.

We also compared our work to a previous deep neural network model of SHM (***Tang et al., 2022***), which was trained on the `tang` data set. A DeepSHM model consists of a pair of CNN models, one for estimating mutation frequencies and another for CSPs; this is akin to our "independent" model configuration. Each of these CNNs has over 250,000 parameters, so in total is about 100 times larger than the largest CNN model we trained. These CNNs are also slow to evaluate: because they make predictions one k-mer at a time, one must iterate over the sequence and obtain predictions for every site. Because the DeepSHM model cannot handle ambiguous nucleotides, we had to remove these from the evaluation. The authors found the best performance is achieved with 15-mers, which is comparable with the 11-mers and 13-mers in our thrifty models. We evaluated the DeepSHM 15-mer model on the subset of the `briney` data and found it performs better than S5F but comparable to our models (Table 2). Specifically, our models performed comparably when trained on the `tang` data only, but when trained on the combined `tang` data and `briney` two largest repertoires, our models performed slightly better on the `briney` held-out repertoires.

model	training data	AUROC	AUPRC	R-prec	sub. acc.
S5F	*	0.775	0.0698	0.0290	0.514
CNN Joined Large	`tang`	0.787	0.0850	0.0406	0.510
CNN Indep Medium	`tang`	0.786	0.0848	0.0407	0.528
CNN Indep Large	`tang`	0.786	0.0852	0.0406	0.524
DeepSHM	`tang`†	0.786	0.0876	0.0421	0.537
CNN Joined Large	`tang+briney`	0.793	0.0923	0.0439	0.551
CNN Indep Medium	`tang+briney`	0.793	0.0919	0.0441	0.560
CNN Indep Large	`tang+briney`	0.794	0.0926	0.0450	0.562

Table 2. Performance evaluation on held-out `briney` data for S5F, DeepSHM, and thrifty models. The * on S5F indicates that this model was trained using synonymous mutations on a distinct data set to those considered here. The † on `tang`† is to signify that this is the `tang` data but with a different preprocessing scheme (***Tang et al., 2022***).

We next characterized models in terms of out-of-sample model fit. For each site in the parent of each parent-child pair (PCP) of sequences, we computed the probability of a nucleotide substitution at that site in the corresponding child. We then compared the sum of those probabilities to the actual number of substitutions that were observed at each site in the PCPs (Figure 3). If the observed and expected counts match, then the model is doing a good job, on average, of predicting site-specific probabilities of substitution. We assessed matching using an "overlap" metric, which quantifies the size of the intersection of the histograms divided by the average area of the histograms. We also assessed model log likelihood. These assessments were performed after branch length optimization to maximize likelihood.

We found that, as with the predictive metrics, all models gave similar performance, the differences of which were smaller than the differences between data sets. The overlap metric was better on the 5mer model for the held-out `briney` data, but was better for the CNN models for the `tang` data. The log likelihood was slightly better for the CNN model for all held-out data sets

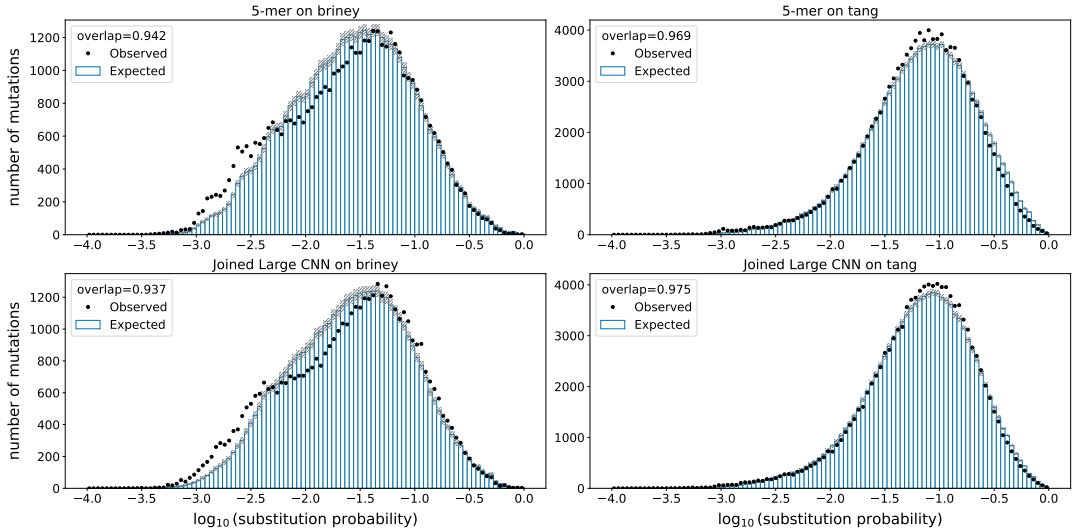

Figure 3. Model fit on held-out samples: held-out individuals from the `briney` data (upper row) and on a separate sequencing experiment (`tang` data, lower row). Observed mutations in held-out data are placed in bins according to their probability of mutation. For every bin, the points show the observed number of mutations in that bin, while the bars show the expected number of mutations in that bin. The overlap metric is the area of the intersection of the observed and expected divided by the average area between the two.

Figure 3—figure supplement 1. Comparison of held-out log likelihoods between the models.

(***Figure 3—figure Supplement 1***).

Further model elaborations did not improve out of sample performance

We tried adding a per-site rate to our CNN models, as well as other elaborations such as a transformer model directly on the amino acid embeddings, and a transformer combined with a CNN. We also tried adding a positional encoding to the input for the CNN model, which does not require additional parameters, but rather perturbs the input embeddings in a way that indicates their position in the sequence (***Vaswani et al., 2017***). None of these elaborations improved performance on held-out data. All of these experiments can be found in notebooks in the GitHub repository associated with this paper.

We also found that jointly optimizing branch lengths along with model parameters did not improve out-of-sample performance.

Out of frame evolution and synonymous mutations give different results

We conclude from the above that the richness of these models is limited by data volume, but unfortunately we were not able to find additional data sets with many out-of-frame sequences (see end of *Methods*). This raised the question of if we can use synonymous mutations of productive sequences to augment our data set. To do so, we trained models using a combination of the original training using the `briney` data but also with the `tang` data where the loss function was restricted to 4-fold synonymous sites.

We found that adding these synonymous mutations to the training set reduced performance on the held-out out-of-frame data (***Figure 4***). Furthermore, when we train in the usual way using the `briney` data and evaluate on synonymous data, we do significantly worse than the S5F model, which was trained on synonymous mutations (***Figure 4—figure Supplement 1***, ***Figure 4—figure Supplement 2***). Our conclusion is that these two sources of data capture the effect of different processes.

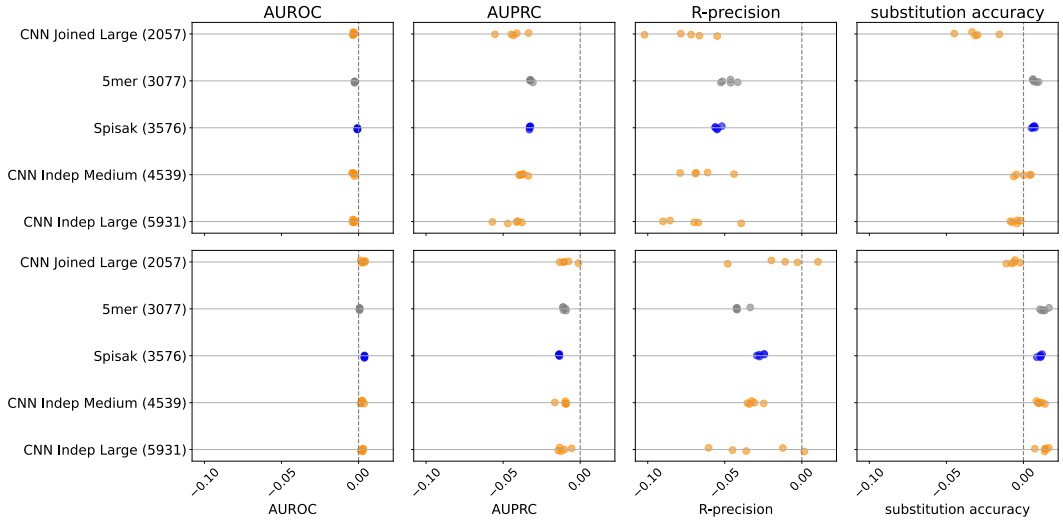

Figure 4. The relative change in performance for each statistic, namely the statistic for the model trained with out of frame (OOF) data and synonymous data, minus the statistic for the model trained with OOF data only, divided by the statistic trained with OOF data only. Thus, adding in synonymous mutations to the training set does not help predict on held-out out-of-frame data. Results shown for `briney` data (upper row) and `tang` data (lower row).

Figure 4—figure supplement 1. S5F performs better for mutation position prediction on synonymous mutations in the `jaffe` data than any model trained on out of frame data.

Figure 4—figure supplement 2. S5F performs better for mutation position prediction on synonymous mutations in the `tang` data than any model trained on out of frame data.

Discussion

Models of somatic hypermutation are important to understand affinity maturation of B cells. They can also provide insight into the mechanism of SHM itself. Here we have developed a collection of models and worked to learn what they can teach us about SHM.

Specifically, by using an overlapping 3-mer embedding approach, we can parameterize wide-context models with relatively few parameters. For example, we can parameterize a 13-mer model with fewer parameters than a 5-mer model with better out-of-sample performance. A similar approach of embedded k-mers has been successful previously for genome regulatory element function prediction (***Ji et al., 2021***).

Our first main result is that these models are better than previous models using a 5-mer context, but only slightly so. This is interesting given that more distant effects are well motivated biologically, both in terms of the consensus model of somatic hypermutation involving DNA damage, stripping, and repair (***Pilzecker and Jacobs, 2019***), as well as more recent results showing that the mesoscale environment of the BCR is important for SHM (***Wang et al., 2023***). Although another model formulation may be able to pick up these features, the present approach should be able to pick up features such as a nearby AID hotspot.

We found that adding a per-site rate to our models did not help predict on out-of-sample data. This is in contrast to recent work of ***Spisak et al.*** (***2020***) suggesting that a per-site rate was helpful to predict SHM. We suspect that this contrast may be because of the means of evaluating this model. The ***Spisak et al.*** (***2020***) paper quantifies an improved model fit over a 5mer model by calculating a Pearson r^2 for each region comparing the model prediction to the aggregated mutation count per site for sites in that region. The ***Spisak et al.*** (***2020***) model is itself parameterized in a per-site way, and so it is not surprising that the model has excellent fit according to this metric (Figure 4J of ***Spisak et al., 2020***). Although they did a data-splitting exercise, this data split was on the level of parent-child pairs (not specified in the original paper; clarified via personal communication), which

means that many of the parent sequences in the training set were very similar to parent sequences in the test set. Furthermore, the baseline comparison provided in Figure 4J of ***Spisak et al.*** (***2020***) is actually not to a separately trained 5mer model, but to the 5mer component γ_w of the joint model (typo in the original paper; clarified via personal communication).

We tried other means of adding per-site rates to the model, such as positional encoding and a transformer component, but these did not help. Although it is quite possible that sites evolve differently according to their absolute position in the sequence, our work shows that this is not necessary as part of a predictive model. Due to commonalities between neighborhoods of sites in sequences, it is difficult or perhaps impossible to disentangle the effect of a per-site rate from the effect of the motif model given a sufficiently rich motif model.

From a biological perspective, our findings indicate that while per-site rate (***Cohen et al., 2011***; ***Zhou and Kleinstein, 2020***; ***Spisak et al., 2020***) and mesoscale-level (***Wang et al., 2023***) effects may exist in SHM, their impact appears insufficient to significantly enhance statistical modeling performance. We refrain from making further biological conclusions, and suggest that more definitive conclusions require an explicitly mechanistic model (***Fisher et al., 2025***).

Our work also contrasts the two main ways of training a neutral model: one is to use synonymous mutations (***Yaari et al., 2012***), and the other is to use out-of-frame sequences (***Spisak et al., 2020***) or some other sequences that one can assume are evolving neutrally (***Cui et al., 2016***). Here we find evidence that these two methods give different results, and that models trained on one task do not do well on the other. This may be from synonymous mutations being under selection due to codon usage effects, or could be from only a subset of motifs being possible to estimate from synonymous data (***Yaari et al., 2013***). Another possible explanation is that the spatial correlation of mutations (***Spisak et al., 2020***) leads to correlation in mutations at synonymous and non-synonymous sites: as an extreme example, a four-fold synonymous site could be next to a site under strong purifying selection, and if these two sites only mutate together this would effectively lead to purifying selection on the synonymous site. One could also imagine that the out-of-frame sequences are under selection such as having an insertion-deletion mutation that throws a sequence under selection out of frame, but this is mitigated by the ancestral sequence reconstruction and the removal of the naïve sequence from the tree. From a model-fitting perspective, the contrast between these two objectives is disappointing, because productive sequences are much more available than out-of-frame sequences.

Overall, we have presented and tested a variety of new models with test-train splits, and found a slight improvement using parameter sparse or "thrifty" CNNs. One interesting aspect of these models is that they allow for a wider k-mer context without having a parameter explosion. It is possible that the conclusions would differ if we had considerably more data, though despite our best efforts (see Materials and Methods) we were unable to find a large additional volume of out-of-frame data. If and when additional data becomes available, our reproducible analysis can be used to evaluate these models on that data.

Materials and Methods

Data sets and data processing

Our primary dataset is the one introduced by Spisak et al. (***Spisak et al., 2020***) in their recent work on modeling somatic hypermutation (SHM). The data consists of human, out-of-frame IgH sequences sampled from several individuals (***Briney et al., 2019***), aligned using pRESTO (***Vander Heiden et al., 2014***). The resulting data comes to us as a collection of trees, one for each clonal family with at least six observed sequences in an individual, complete with (ancestral and observed) sequences and branch lengths, as well as a collection of metadata annotating, among other things, the identity and position of the V gene in the sequence. Starting from the clonal family clustering and multiple sequence alignments from (***Spisak et al., 2020***), we remove sequences containing gaps to avoid ambiguity of site positions due to insertions or deletions. Resulting clonal families

with less than six observed sequences are discarded. Phylogenetic inference in each clonal family is then redone with IQ-TREE (***Minh et al., 2020***). We apply the K80 (***Kimura, 1980***) substitution model and use the germline sequence as an outgroup, following the method of (***Spisak et al., 2020***), and additionally allow for mutation rate heterogeneity among sites with a 4-category FreeRate model (***Yang, 1995***; ***Soubrier et al., 2012***). The edges of the inferred phylogenetic trees are taken as parent-child pairs, except for the edge containing the naïve sequence outgroup.

Additionally, we use the `tang` data set of human IgH sequences previously used for modeling SHM by (***Tang et al., 2022***). This data set, originally generated by (***Tang et al., 2020***; ***Vergani et al., 2017***), includes full-length BCR repertoires from 21 individuals. We obtained pre-processed samples directly from the authors (see ***Tang et al., 2020***, for data preprocessing details). We extracted the IgH sequences from marginal zone (MZ), memory (M), and plasma (PC) B cells and completed germline and clonal family inferences with `partis` (***Ralph and Matsen, 2016a***,b, ***2019***). We keep clonal families of two or more sequences that consist of only out-of-frame sequences. In `partis`, a sequence is considered out-of-frame if either of the conserved codons for cysteine or tryptophan bounding the CDR3 is out-of-frame with the germline V gene. We then perform phylogenetic inference and ancestral sequence reconstruction for each clonal family using IQ-TREE as described above. The first site of the sequence is set to align with the start of the germline V gene; if necessary, nucleotides before the start of the V gene are truncated or sites at the 5′ end with missing reads are padded with 'N'.

A small number of the edges have very long branch lengths, some of which seem to correspond to improper alignments. Following the lead of the authors of (***Spisak et al., 2020***), we filter our data to only include edges with fewer than 10 mutations, which results a modest $\sim 6.4\%$ reduction in the available data.

The `jaffe` data set (***Jaffe et al., 2022***) consists of paired heavy chain and light chain, full-length sequences of productive antibodies from four donors. We process the data with `partis` and IQ-TREE, similar to the `tang` data set. The `jaffe` synonymous data set are parent-child pairs from `jaffe` where we mask sites in the child sequence that are not 4-fold degenerate in the parent context. For each parent-child pair, we check each codon context in the parent sequence for 4-fold degenerate sites. For these sites, the corresponding nucleotide in the child sequence is preserved while all other sites are masked.

name	type	sequencing method	read/cell	samples	CFs	PCPs	median mutations
`briney`	out-of-frame	bulk	∼1	9	3,903	52,915	2
`tang`	out-of-frame	single cell	∼40	21	3,209	9,984	7
`tang`	productive	single cell	∼40	21	157,863	820,837	1
`jaffe`	productive	single cell	∼5000	4	67,304	282,732	1

Table 3. Data used in this paper. `briney` data is from (***Briney et al., 2019***) after processing done by (***Spisak et al., 2020***). `tang` data is from ***Vergani et al.*** (***2017***); ***Tang et al.*** (***2020***) and was sequenced using the methods of ***Vergani et al.*** (***2017***). Out-of-frame sequences from `briney` and `tang` are used. For productive sequences from `tang`, only 4-fold synonymous sites are used. `jaffe` data is from ***Jaffe et al.*** (***2022***) sequenced using 10X, where only 4-fold synonymous sites of productive sequences are used. The "read/cell" column is the sequencing depth listed as average number of reads per cell. "Samples" is the number of individual samples in the dataset; in these datasets, each sample is from a distinct individual. "CFs" is the number of clonal families in the dataset. "PCPs" is the number of parent-child pairs in the dataset. "Median mutations" is the median number of mutations per PCP in the dataset.

Models

In all the models we propose, we model SHM using a two-part model. First we assume that the occurrence of point mutations follows a per-site exponential waiting time that is dependent entirely

on the parent sequence. Furthermore, we assume that all mutations occur simultaneously, so that we can ignore how the order of point mutations along a branch affects likelihood computations. The other output of the model is the prediction of base identity. We interpret the normalized version of these rates as conditional probabilities given that the mutation has occurred.

For the "thrifty" models, we have an embedding of each 3-mer along the sequence which is then the input for a convolutional layer (***Figure 1***b). The output of the convolutional layer is then the input for the mutation rate estimate, which is shown in (***Figure 1***b) as well as the CSP (***Figure 1—figure Supplement 1***).

All models are implemented in PyTorch (***Paszke et al., 2019***) and can be found in the GitHub repository associated with this paper.

Model training

Our loss functions are the likelihood of child mutation locations using offset, as well as a categorical cross-entropy loss for the base identity. We sum these log losses together using a weight of 0.01 on the cross-entropy loss to approximately even out the contributions of these two sources of loss.

Models were trained for 100 epochs, with the Poisson offset simply being the normalized count of mutations in the child sequence. We also tried a more sophisticated approach of joint optimization of branch length and model parameters.

Model evaluation

AUROC

The Receiver Operating Characteristic (ROC) curve is a method of visualizing the trade-off between true positive rate (TPR=$\frac{\text{TP}}{\text{TP+FN}}$) and false positive rate (FPR=$\frac{\text{FP}}{\text{TN+FP}}$) as we vary the cutoff value we use for separating positive and negative predictions. Notice that the denominators in each statistic depend only on the true labels, so the trade-off parallels the one between true positives and false positives. Given a random classifier, the TPR is expected to be equal to FPR, meaning that the ROC would be a diagonal line from $(0, 0)$ to $(1, 1)$.

In order to reduce the ROC curve to a single quantity for performance assessment, one can compute the area under the ROC curve (AUROC) which, when compared to 0.5, gives a sense of how well the classifier is doing when compared with a random classifier. ***Hanley and McNeil*** (***1982***) show that AUROC is equivalent to probability that the classifier correctly ranks a randomly-chosen pair of points, one from the negative and one from the positive class.

AUPRC

Class imbalance, where we have many more negative examples than positive examples, can muddle performance evaluation of a classifier. Specifically, a large number of correctly-identified negative examples can obscure the relatively-poor performance of a classifier on a class that is under-represented in the data. Another approach is to instead consider precision and recall (***Saito and Rehmsmeier, 2015***). Analogous to the ROC curve, the principle here is to track the precision ($\frac{\text{TP}}{\text{TP+FP}}$) and recall ($\frac{\text{TP}}{\text{TP+FN}}$) as we vary the cutoff parameter and plot the resulting points. Similar to before, we can distill the information of this curve down to a single number in the unit interval by computing the area under the precision-recall curve (AUPRC). The precision of a classifier that uniformly assigns sites to the positive and negative classes will be

$$\rho = \frac{\text{\# mutated sites}}{\text{\# sites}}$$

in expectation. Thus, if we exclude pathologically bad classifiers, ρ forms a base-line minimum value of the AUPRC. In contrast with the AUROC, neither precision nor recall are affected by the addition or removal of true negative (i.e. non-mutated sites that were predicted not to mutate) examples, and in fact the relationship is driven by the tradeoff between false positives and false negatives.

R-precision

In order to characterize the ability of our classifiers to correctly identify sites that will mutate we consider another metric: R-precision. This is easiest to explain by first introducing top-k precision, in which we take the k "hottest" (that is, predicted to mutate with the highest probability) sites according to our classifier and compute the precision on those sites. We can refine top-k precision above to R-precision, which is the top-k precision when k is set equal to the expected or observed number of mutations. This value can be interpreted in the following way: an R-precision of 0.1 means that if the classifier is told to predict the correct sites to mutate given their count, 10% of them will have actually mutated. As with AUPRC, a random classifier will have an R-precision of ρ.

Substitution accuracy

As described above, when evaluating performance at predicting per-base substitution probabilities (given a mutation occurred), we report accuracy: how frequently is the predicted-most-likely base the one to which a site mutates?

Software

Our work is released in an open-source Python package https://github.com/matsengrp/netam with a simple API that makes it easy to train and evaluate models. We release trained models and their weights. Our analysis for this paper is reproducible via https://github.com/matsengrp/netam-experiments-1, which includes notebooks that reproduce the figures and tables in this paper. We used the following software: PyTorch (***Paszke et al., 2019***), pandas (***Wes McKinney, 2010***), matplotlib (***Hunter, 2007***), seaborn (***Waskom, 2021***), snakemake (***Mölder et al., 2021***), pytest (***Krekel et al., 2004***), biopython (***Cock et al., 2009***).

Attempts to find additional full-length, out-of-frame sequences

While the primary dataset used here and originally by ***Spisak et al.*** (***2020***) provides a large number of out-of-frame human IgH sequences, the sequencing methods used resulted in minimal coverage at the start of the V gene and thus limited information for that region (***Briney et al., 2019***). Alternatively, the Tang dataset provides relatively high coverage along the full IgH sequence, but is limited in the amount of unique sequences sampled. We sought to supplement this data with additional full-length, out-of-frame human IgH sequences. Full-length IgH data is generally limited by design in common sequencing approaches. Even productive IgH data show low coverage at the start of the V gene, as evidenced by over 40% of the sequences in the Observed Antibody Space (OAS) database lacking sequence data for the first 15 amino acids (***Olsen et al., 2022b***; ***Kovaltsuk et al., 2018***; ***Olsen et al., 2022a***). We focus our efforts on datasets from recent studies focused on full-length antibody sequencing (***Soto et al., 2019***; ***Ford et al., 2023***; ***Rodriguez et al., 2023***). For all datasets, we implemented the `partis`-IQ-TREE pipeline as previously described on pre-processed data and extracted parent-child pairs for all clonal families of size 2+. Overall, out-of-frame sequences make up a relatively small proportion of sequence data (with this proportion varying by study/sequencing protocols) and none of the datasets considered had enough depth to extract a meaningful amount of out-of-frame sequence data for our purposes. The details of these efforts are described below.

From ***Ford et al.*** (***2023***), we obtained pre-processed data for 10 IgG FLAIRR-seq samples from the authors. Using consensus sequences from UMIs that were observed more than once (DUPCOUNT>1), we recovered 7,938 out-of-frame sequences across all samples. These sequences belonged to 226 clonal families of size 2+ and 2,633 singletons. This amounted to only 722 parent-child pairs, of which only 324 contained a mutation event. In attempts to extract more out-of-frame data, we additionally ran our pipeline on the pre-processed data including consensus sequences that were only observed once (and thus included no UMI error correction). From the 10 samples, we were able to recover 52,785 out-of-frame sequences, some of which were not UMI error corrected. This resulted in 2,415 clonal families of size 2+ and 22,983 singletons, but still only gave

us 6,296 parent-child pairs with mutation events (9,114 in total). We note that this sequencing method provided a higher proportion of out-of-frame sequences (9% for fully pre-processed data) than other methods we considered.

From ***Rodriguez et al.*** (***2023***), we obtained pre-processed 5' RACE AIRR-seq sequencing data for 51 IgG samples from the authors. We ran our pipeline on the pre-processed data and recovered only 4,337 out-of-frame sequences in total. These sequences belonged to 188 clonal families of size 2+ and 1,925 singletons. Of the 889 parent-child pairs from non-singletons, only 215 contained a mutation event. While the proportion of out-of-frame sequences was on-par with the primary dataset, sequence depth per sample was shallow and thus out-of-frame data was limited.

From ***Soto et al.*** (***2019***), we obtained pre-processed data for all 3 HIP donors from the authors. We ran our pipeline on a large subset of the data (sampling the first 1 million sequences for each donor IgH fasta file) to assess its potential for our purposes. From the 3 million sequences processed, we extracted 2,686 out-of-frame sequences in total. These sequences corresponded to 11 clonal families of size 2+ and 2,618 singletons. We obtained 102 parent-child pairs from non-singletons, of which only 57 contained a mutation event. The relatively low recovery of out-of-frame sequences in this subset of the data suggested that processing the full dataset would not yield a meaningful amount of parent-child pairs for this study. We additionally observed that all of these sequences had no coverage at the start of the V gene, missing the first 12-60 bases.

Acknowledgements

Thank you to the authors of (***Spisak et al., 2020***) for providing the data and answering questions about their work. We are also grateful to the authors of (***Tang et al., 2022***) for providing the data and answering questions about their work, as well as to the lab of Corey Watson for sharing data from (***Ford et al., 2023***) and (***Rodriguez et al., 2023***). Additionally, we thank Luke Myers and the authors of (***Soto et al., 2019***) for sharing their pre-processed data.

This work supported by NIH grant R01-AI146028. Scientific Computing Infrastructure at Fred Hutch funded by ORIP grant S10OD028685. Frederick Matsen is an investigator of the Howard Hughes Medical Institute.

This work was partially completed at the Kavli Institute for Theoretical Physics (KITP) at the University of California, Santa Barbara, and thus was supported by grant no. NSF PHY-2309135 to the Kavli Institute for Theoretical Physics (KITP) and the Gordon and Betty Moore Foundation Grant No. 2919.02.

References

Akiba T, Sano S, Yanase T, Ohta T, Koyama M. Optuna: A Next-generation Hyperparameter Optimization Framework. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; 2019. .

Briney B, Inderbitzin A, Joyce C, Burton DR. Commonality despite exceptional diversity in the baseline human antibody repertoire. Nature. 2019; 566(7744):393–397.

Cock PJA, Antao T, Chang JT, Chapman BA, Cox CJ, Dalke A, Friedberg I, Hamelryck T, Kauff F, Wilczynski B, de Hoon MJL. Biopython: freely available Python tools for computational molecular biology and bioinformatics. Bioinformatics. 2009 03; 25(11):1422–1423. https://doi.org/10.1093/bioinformatics/btp163, doi: 10.1093/bioinformatics/btp163.

Cohen RM, Kleinstein SH, Louzoun Y. Somatic hypermutation targeting is influenced by location within the immunoglobulin V region. Mol Immunol. 2011 Jul; 48(12-13):1477–1483. http://dx.doi.org/10.1016/j.molimm.2011.04.002, doi: 10.1016/j.molimm.2011.04.002.

Cui A, Di Niro R, Vander Heiden JA, Briggs AW, Adams K, Gilbert T, O'Connor KC, Vigneault F, Shlomchik MJ, Kleinstein SH. A Model of Somatic Hypermutation Targeting in Mice Based on High-Throughput Ig Sequencing Data. J Immunol. 2016 Nov; 197(9):3566–3574. http://dx.doi.org/10.4049/jimmunol.1502263, doi: 10.4049/jimmunol.1502263.

Dunn-Walters DK, Dogan A, Boursier L, MacDonald CM, Spencer J. Base-specific sequences that bias somatic hypermutation deduced by analysis of out-of-frame human IgVH genes. J Immunol. 1998 Mar; 160(5):2360–2364. https://www.ncbi.nlm.nih.gov/pubmed/9498777.

Elhanati Y, Sethna Z, Marcou Q, Callan CG Jr, Mora T, Walczak AM. Inferring processes underlying B-cell repertoire diversity. Philos Trans R Soc Lond B Biol Sci. 2015 Sep; 370(1676). http://dx.doi.org/10.1098/rstb.2014.0243, doi: 10.1098/rstb.2014.0243.

Feng J, Shaw DA, Minin VN, Simon N, Matsen FA IV. Survival analysis of DNA mutation motifs with penalized proportional hazards. Ann Appl Stat. 2019 Jun; 13(2):1268–1294. https://projecteuclid.org/euclid.aoas/1560758446, doi: 10.1214/18-AOAS1233.

Fisher T, Sung K, Simon N, Fukuyama J, Matsen FA IV. Inferring mechanistic parameters of somatic hypermutation using neural networks and approximate Bayesian computation. Ann Appl Stat. 2025 Mar; 19(1):720–743. https://projecteuclid.org/journals/annals-of-applied-statistics/volume-19/issue-1/Inferring-mechanistic-parameters-of-somatic-hypermutation-using-neural-networks-and/10.1214/24-AOAS1985.short, doi: 10.1214/24-aoas1985.

Ford EE, Tieri D, Rodriguez OL, Francoeur NJ, Soto J, Kos JT, Peres A, Gibson WS, Silver CA, Deikus G, Hudson E, Woolley CR, Beckmann N, Charney A, Mitchell TC, Yaari G, Sebra RP, Watson CT, Smith ML. FLAIRR-seq: A method for single-molecule resolution of near full-length antibody H chain repertoires. J Immunol. 2023 May; 210(10):1607–1619. http://dx.doi.org/10.4049/jimmunol.2200825, doi: 10.4049/jimmunol.2200825.

Hanley JA, McNeil BJ. The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology. 1982; 143(1):29–36.

Hoehn KB, Lunter G, Pybus OG. A Phylogenetic Codon Substitution Model for Antibody Lineages. Genetics. 2017 May; 206(1):417–427. http://dx.doi.org/10.1534/genetics.116.196303, doi: 10.1534/genetics.116.196303.

Hoehn KB, Vander Heiden JA, Zhou JQ, Lunter G, Pybus OG, Kleinstein SH. Repertoire-wide phylogenetic models of B cell molecular evolution reveal evolutionary signatures of aging and vaccination. Proc Natl Acad Sci U S A. 2019 Oct; http://dx.doi.org/10.1073/pnas.1906020116, doi: 10.1073/pnas.1906020116.

Hunter JD. Matplotlib: A 2D graphics environment. Computing in Science & Engineering. 2007; 9(3):90–95. doi: 10.1109/MCSE.2007.55.

Jaffe DB, Shahi P, Adams BA, Chrisman AM, Finnegan PM, Raman N, Royall AE, Tsai F, Vollbrecht T, Reyes DS, Hepler NL, McDonnell WJ. Functional antibodies exhibit light chain coherence. Nature. 2022 Nov; 611(7935):352–357. http://dx.doi.org/10.1038/s41586-022-05371-z, doi: 10.1038/s41586-022-05371-z.

Ji Y, Zhou Z, Liu H, Davuluri RV. DNABERT: pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome. Bioinformatics. 2021 Aug; 37(15):2112–2120. https://academic.oup.com/bioinformatics/article-pdf/37/15/2112/57195892/btab083.pdf, doi: 10.1093/bioinformatics/btab083.

Kimura M. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. Journal of Molecular Evolution. 1980 06; 16:111–120. https://doi.org/10.1007/BF01731581, doi: 10.1007/BF01731581.

Kovaltsuk A, Leem J, Kelm S, Snowden J, Deane CM, Krawczyk K. Observed Antibody Space: A resource for data mining next-generation sequencing of antibody repertoires. J Immunol. 2018 Oct; 201(8):2502–2509. http://dx.doi.org/10.4049/jimmunol.1800708, doi: 10.4049/jimmunol.1800708.

Krekel H, Oliveira B, Pfannschmidt R, Bruynooghe F, Laugher B, Bruhin F, pytest 8.1.1; 2004. https://github.com/pytest-dev/pytest.

Levinstein Hallak K, Rosset S. Statistical modeling of SARS-CoV-2 substitution processes: predicting the next variant. Commun Biol. 2022 Mar; 5(1):285. http://dx.doi.org/10.1038/s42003-022-03198-y, doi: 10.1038/s42003-022-03198-y.

Levinstein Hallak K, Tzur S, Rosset S. Big data analysis of human mitochondrial DNA substitution models: a regression approach. BMC Genomics. 2018 Oct; 19(1):759. http://dx.doi.org/10.1186/s12864-018-5123-x, doi: 10.1186/s12864-018-5123-x.

Marcou Q, Mora T, Walczak AM. High-throughput immune repertoire analysis with IGoR. Nat Commun. 2018 Feb; 9(1):561. http://dx.doi.org/10.1038/s41467-018-02832-w, doi: 10.1038/s41467-018-02832-w.

Martin Beem JS, Venkatayogi S, Haynes BF, Wiehe K. ARMADiLLO: a web server for analyzing antibody mutation probabilities. Nucleic Acids Res. 2023 Jul; 51(W1):W51–W56. https://academic.oup.com/nar/advance-article-abstract/doi/10.1093/nar/gkad398/7187705, doi: 10.1093/nar/gkad398.

McCoy CO, Bedford T, Minin VN, Bradley P, Robins H, Matsen FA IV. Quantifying evolutionary constraints on B-cell affinity maturation. Philos Trans R Soc Lond B Biol Sci. 2015 Sep; 370(1676). http://dx.doi.org/10.1098/rstb.2014.0244, doi: 10.1098/rstb.2014.0244.

Wes McKinney. Data Structures for Statistical Computing in Python. In: Stéfan van der Walt, Jarrod Millman, editors. *Proceedings of the 9th Python in Science Conference*; 2010. p. 56 – 61. doi: 10.25080/Majora-92bf1922-00a.

Methot SP, Di Noia JM. Chapter Two - Molecular Mechanisms of Somatic Hypermutation and Class Switch Recombination. In: Frederick W Alt, editor. *Advances in Immunology*, vol. 133 Academic Press; 2017.p. 37–87. http://www.sciencedirect.com/science/article/pii/S0065277616300530.

Minh BQ, Schmidt HA, Chernomor O, Schrempf D, Woodhams MD, von Haeseler A, Lanfear R. IQ-TREE 2: New Models and Efficient Methods for Phylogenetic Inference in the Genomic Era. Molecular Biology and Evolution. 2020 02; 37(5):1530–1534. https://doi.org/10.1093/molbev/msaa015, doi: 10.1093/molbev/msaa015.

Mölder F, Jablonski K, Letcher B, Hall M, Tomkins-Tinch C, Sochat V, Forster J, Lee S, Twardziok S, Kanitz A, Wilm A, Holtgrewe M, Rahmann S, Nahnsen S, Köster J. Sustainable data analysis with Snakemake [version 2; peer review: 2 approved]. F1000Research. 2021; 10(33). doi: 10.12688/f1000research.29032.2.

Olsen TH, Boyles F, Deane CM. Observed Antibody Space: A diverse database of cleaned, annotated, and translated unpaired and paired antibody sequences. Protein Sci. 2022 Jan; 31(1):141–146. http://dx.doi.org/10.1002/pro.4205, doi: 10.1002/pro.4205.

Olsen TH, Moal IH, Deane CM. AbLang: an antibody language model for completing antibody sequences. Bioinform Adv. 2022 Jun; 2(1):vbac046. http://dx.doi.org/10.1093/bioadv/vbac046, doi: 10.1093/bioadv/vbac046.

Ozenne B, Subtil F, Maucort-Boulch D. The precision–recall curve overcame the optimism of the receiver operating characteristic curve in rare diseases. J Clin Epidemiol. 2015 Aug; 68(8):855–859. http://dx.doi.org/10.1016/j.jclinepi.2015.02.010, doi: 10.1016/j.jclinepi.2015.02.010.

Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Köpf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J, et al. PyTorch: An imperative style, high-performance deep learning library. arXiv. 2019 Dec; https://papers.nips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.

Pilzecker B, Jacobs H. Mutating for Good: DNA Damage Responses During Somatic Hypermutation. Front Immunol. 2019 Mar; 10:438. http://dx.doi.org/10.3389/fimmu.2019.00438, doi: 10.3389/fimmu.2019.00438.

Ralph DK, Matsen FA IV. Consistency of VDJ rearrangement and substitution parameters enables accurate B cell receptor sequence annotation. PLoS Comput Biol. 2016 Jan; 12(1):e1004409. http://dx.doi.org/10.1371/journal.pcbi.1004409, doi: 10.1371/journal.pcbi.1004409.

Ralph DK, Matsen FA IV. Likelihood-based inference of B cell clonal families. PLoS Comput Biol. 2016 Oct; 12(10):e1005086. http://dx.doi.org/10.1371/journal.pcbi.1005086, doi: 10.1371/journal.pcbi.1005086.

Ralph DK, Matsen FA IV. Per-sample immunoglobulin germline inference from B cell receptor deep sequencing data. PLoS Comput Biol. 2019 Jul; 15(7):e1007133. http://dx.doi.org/0.1371/journal.pcbi.1007133, doi: 0.1371/journal.pcbi.1007133.

Rodriguez OL, Safonova Y, Silver CA, Shields K, Gibson WS, Kos JT, Tieri D, Ke H, Jackson KJL, Boyd SD, Smith ML, Marasco WA, Watson CT. Genetic variation in the immunoglobulin heavy chain locus shapes the human antibody repertoire. Nature Comm. 2023 Jul; 14(1):4419. http://dx.doi.org/10.1038/s41467-023-40070-x, doi: 10.1038/s41467-023-40070-x.

Rogozin IB, Kolchanov NA. Somatic hypermutagenesis in immunoglobulin genes. II. Influence of neighbouring base sequences on mutagenesis. Biochim Biophys Acta. 1992 Nov; 1171(1):11–18. http://www.ncbi.nlm.nih.gov/pubmed/1420357.

Rogozin IB, Diaz M. Cutting edge: DGYW/WRCH is a better predictor of mutability at G:C bases in Ig hypermutation than the widely accepted RGYW/WRCY motif and probably reflects a two-step activation-induced cytidine deaminase-triggered process. J Immunol. 2004 Mar; 172(6):3382–3384. https://www.ncbi.nlm.nih.gov/pubmed/15004135.

Rosset S. Efficient inference on known phylogenetic trees using Poisson regression. Bioinformatics. 2007 Jan; 23(2):e142–7. http://dx.doi.org/10.1093/bioinformatics/btl306, doi: 10.1093/bioinformatics/btl306.

Saito T, Rehmsmeier M. The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. PLOS ONE. 2015; 10(3):e0118432. https://app.dimensions.ai/details/publication/pub.1012273932, doi: 10.1371/journal.pone.0118432, https://journals.plos.org/plosone/article/file?id=10.1371/journal.pone.0118432.

Soto C, Bombardi RG, Branchizio A, Kose N, Matta P, Sevy AM, Sinkovits RS, Gilchuk P, Finn JA, Crowe Jr JE. High frequency of shared clonotypes in human B cell receptor repertoires. Nature. 2019 02; 566(7744):398–402. https://doi.org/10.1038/s41586-019-0934-8, doi: 10.1038/s41586-019-0934-8.

Soubrier J, Steel M, Lee MSY, Der Sarkissian C, Guindon S, Ho SYW, Cooper A. The Influence of Rate Heterogeneity among Sites on the Time Dependence of Molecular Rates. Molecular Biology and Evolution. 2012 07; 29(11):3345–3358. https://doi.org/10.1093/molbev/mss140, doi: 10.1093/molbev/mss140.

Spisak N, Walczak AM, Mora T. Learning the heterogeneous hypermutation landscape of immunoglobulins from high-throughput repertoire data. Nucleic Acids Res. 2020 Nov; 48(19):10702–10712. http://dx.doi.org/10.1093/nar/gkaa825, doi: 10.1093/nar/gkaa825.

Tang C, Bagnara D, Chiorazzi N, Scharff MD, MacCarthy T. AID overlapping and Polη hotspots are key features of evolutionary variation within the human antibody heavy chain (IGHV) genes. Front Immunol. 2020 Apr; 11:788. http://dx.doi.org/10.3389/fimmu.2020.00788, doi: 10.3389/fimmu.2020.00788.

Tang C, Krantsevich A, MacCarthy T. Deep learning model of somatic hypermutation reveals importance of sequence context beyond hotspot targeting. iScience. 2022 Jan; 25(1):103668. http://dx.doi.org/10.1016/j.isci.2021.103668, doi: 10.1016/j.isci.2021.103668.

Teng G, Papavasiliou FN. Immunoglobulin somatic hypermutation. Annu Rev Genet. 2007; 41:107–120. http://dx.doi.org/10.1146/annurev.genet.41.110306.130340, doi: 10.1146/annurev.genet.41.110306.130340.

Vander Heiden JA, Yaari G, Uduman M, Stern JNH, O'Connor KC, Hafler DA, Vigneault F, Kleinstein SH. pRESTO: a toolkit for processing high-throughput sequencing raw reads of lymphocyte receptor repertoires. Bioinformatics. 2014 03; 30(13):1930–1932. https://doi.org/10.1093/bioinformatics/btu138, doi: 10.1093/bioinformatics/btu138.

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Lu, Polosukhin I. Attention is All you Need. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R, editors. *Advances in Neural Information Processing Systems*, vol. 30 Curran Associates, Inc.; 2017. https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Vergani S, Korsunsky I, Mazzarello AN, Ferrer G, Chiorazzi N, Bagnara D. Novel method for high-throughput full-length IGHV-D-J sequencing of the immune repertoire from bulk B-cells with single-cell resolution. Front Immunol. 2017 Sep; 8:1157. http://dx.doi.org/10.3389/fimmu.2017.01157, doi: 10.3389/fimmu.2017.01157.

Wagner SD, Neuberger MS. Somatic hypermutation of immunoglobulin genes. Annu Rev Immunol. 1996; 14:441–457. http://dx.doi.org/10.1146/annurev.immunol.14.1.441, doi: 10.1146/annurev.immunol.14.1.441.

Wang Y, Zhang S, Yang X, Hwang JK, Zhan C, Lian C, Wang C, Gui T, Wang B, Xie X, Dai P, Zhang L, Tian Y, Zhang H, Han C, Cai Y, Hao Q, Ye X, Liu X, Liu J, et al. Mesoscale DNA feature in antibody-coding sequence facilitates somatic hypermutation. Cell. 2023 Apr; http://dx.doi.org/10.1016/j.cell.2023.03.030, doi: 10.1016/j.cell.2023.03.030.

Waskom ML. seaborn: statistical data visualization. Journal of Open Source Software. 2021; 6(60):3021. https://doi.org/10.21105/joss.03021, doi: 10.21105/joss.03021.

Wiehe K, Bradley T, Ryan Meyerhoff R, Hart C, Williams WB, Easterhoff D, Faison WJ, Kepler TB, Saunders KO, Munir Alam S, Bonsignori M, Haynes BF. Functional Relevance of Improbable Antibody Mutations for HIV Broadly Neutralizing Antibody Development. Cell Host Microbe. 2018 May; 0(0). http://www.cell.com/article/S1931312818302191/abstract, doi: 10.1016/j.chom.2018.04.018.

Wiehe K, Saunders KO, Stalls V, Cain DW, Venkatayogi S, Martin Beem JS, Berry M, Evangelous T, Henderson R, Hora B, Xia SM, Jiang C, Newman A, Bowman C, Lu X, Bryan ME, Bal J, Sanzone A, Chen H, Eaton A, et al. Mutation-Guided Vaccine Design: A Strategy for Developing Boosting Immunogens for HIV Broadly Neutralizing Antibody Induction. bioRxiv. 2022 Nov; p. 2022.11.11.516143. https://www.biorxiv.org/content/10.1101/2022.11.11.516143v1, doi: 10.1101/2022.11.11.516143.

Yaari G, Uduman M, Kleinstein SH. Quantifying selection in high-throughput Immunoglobulin sequencing data sets. Nucleic Acids Res. 2012 May; 40(17):e134. http://dx.doi.org/10.1093/nar/gks457, doi: 10.1093/nar/gks457.

Yaari G, Vander Heiden JA, Uduman M, Gadala-Maria D, Gupta N, Stern JNH, O'Connor KC, Hafler DA, Laserson U, Vigneault F, Kleinstein SH. Models of somatic hypermutation targeting and substitution based on synonymous mutations from high-throughput immunoglobulin sequencing data. Front Immunol. 2013 Nov; 4:358. http://dx.doi.org/10.3389/fimmu.2013.00358, doi: 10.3389/fimmu.2013.00358.

Yang Z. A space-time process model for the evolution of DNA sequences. Genetics. 1995 02; 139(2):993–1005. https://doi.org/10.1093/genetics/139.2.993, doi: 10.1093/genetics/139.2.993.

Zhou JQ, Kleinstein SH. Position-Dependent Differential Targeting of Somatic Hypermutation. J Immunol. 2020 Nov; http://dx.doi.org/10.4049/jimmunol.2000496, doi: 10.4049/jimmunol.2000496.

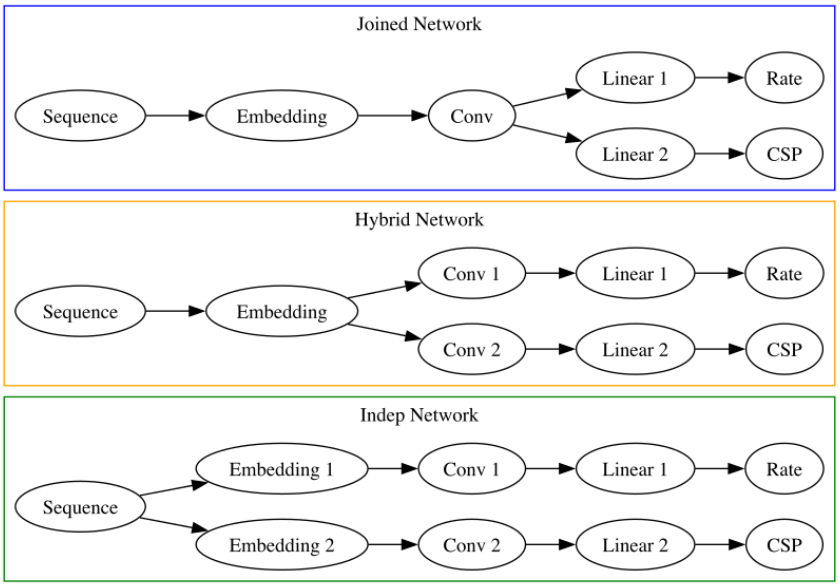

Figure 1—figure supplement 1. Our three strategies for estimating the per-site rate and CSPs. In the joined version, the two outputs come directly out of the convolutional layer. In the hybrid version, the two outputs share the embedding layer. In the independent version, the two outputs are estimated separately.

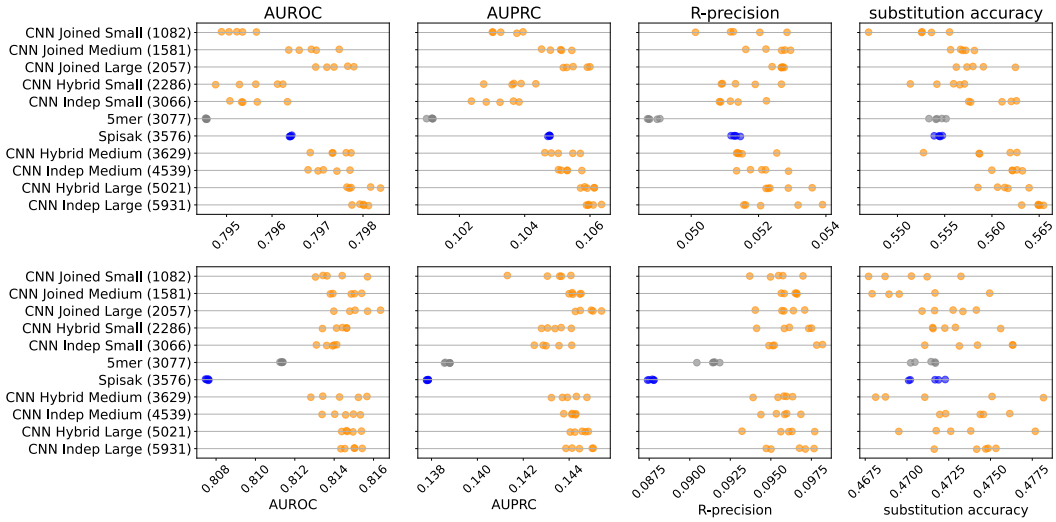

Figure 2—figure supplement 1. Performance results for all the models: held-out individuals from the `briney` data (upper row) and the `tang` data (lower row). The "Small" thrifty models have settings (7, 6, 14, 0.1) for hyperparameters (Kernel, Embed, Filters, Dropout), as described in Table 1.

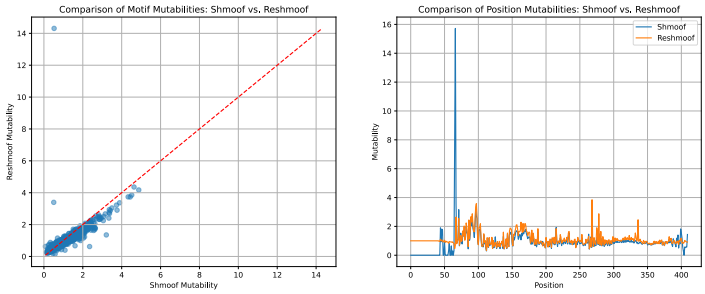

Figure 2—figure supplement 2. There is good agreement between the originally inferred shmoof coefficients and our re-implementation, both in the motif mutability terms and the per-position mutabilities. The primary exception is that we infer more reasonable values when sequencing coverage is weak or absent and avoid an extreme value at site 67. These are due to a slight regularization to the per-position mutabilities.This analysis can be reproduced using the `reshmoof.ipynb` notebook.

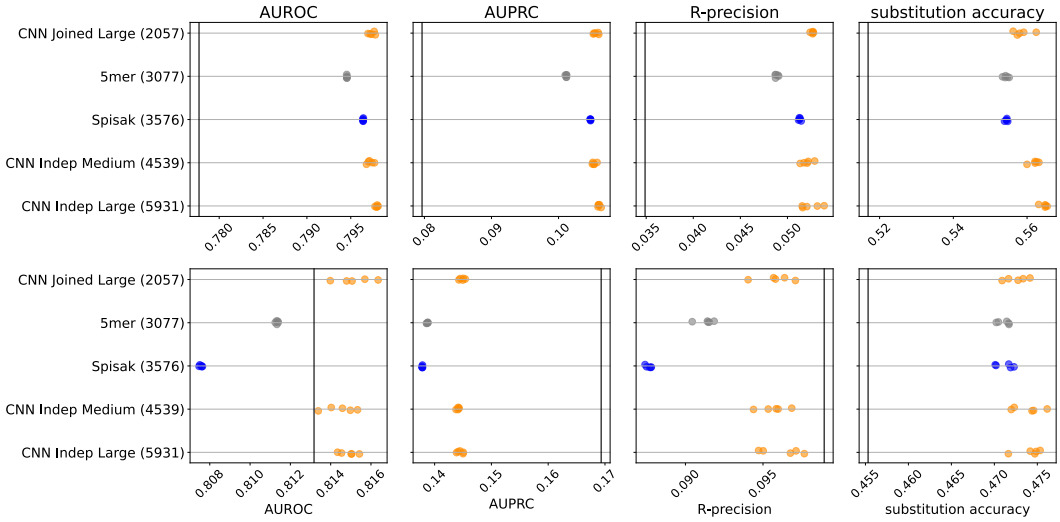

Figure 2—figure supplement 3. Performance plot including the original S5F model (vertical black lines) for held-out individuals from the `briney` data (upper row) and on a separate sequencing experiment (`tang` data, lower row).

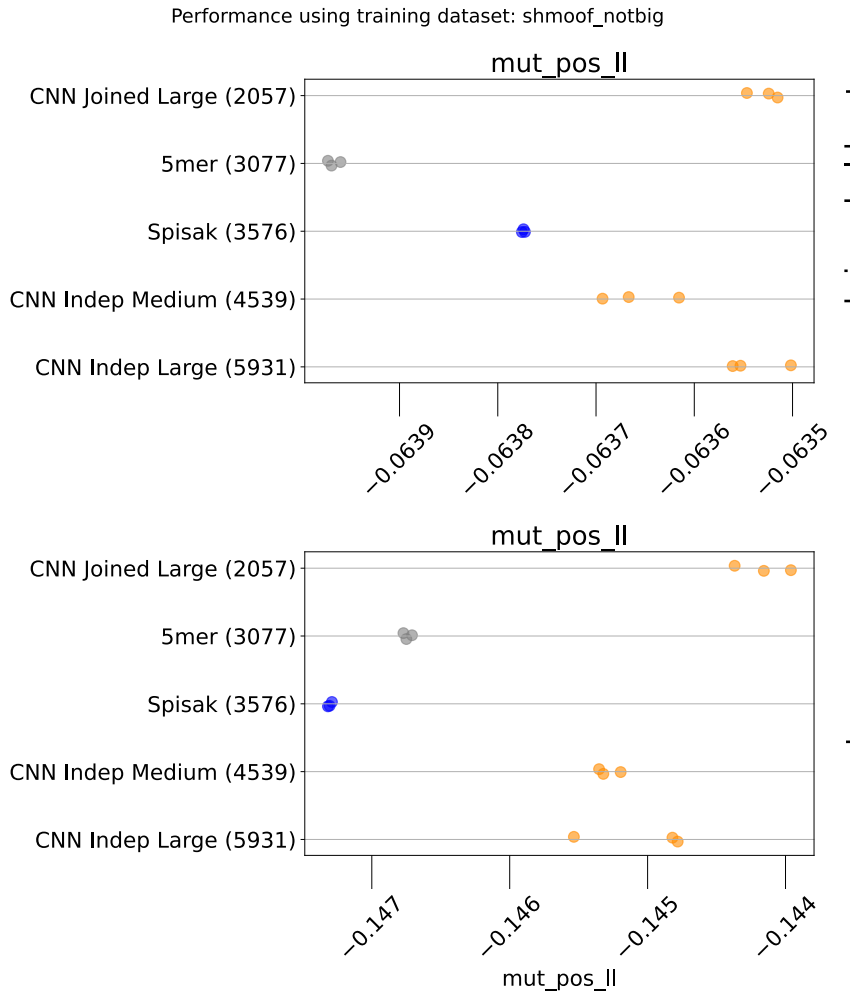

Figure 3—figure supplement 1. Comparison of held-out log likelihoods between the models.

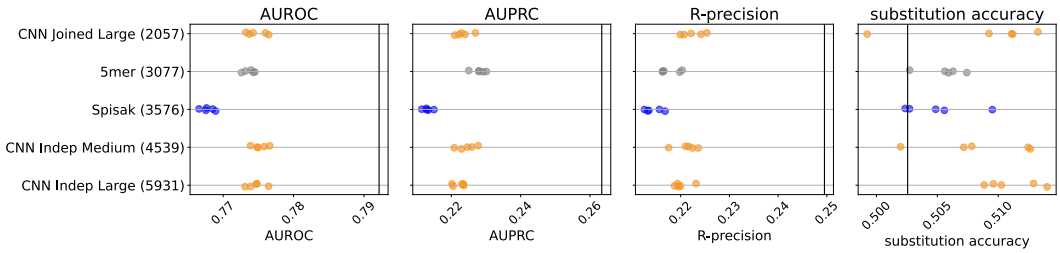

Figure 4—figure supplement 1. S5F performs better for mutation position prediction on synonymous mutations in the `jaffe` data than any model trained on out of frame data. Performance plot is as before, but black vertical bar indicates the performance of the S5F model, and all models evaluated on synonymous mutations in the `jaffe` data.

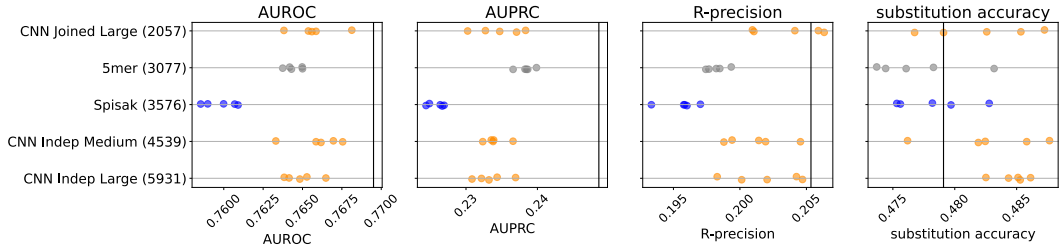

Figure 4—figure supplement 2. S5F performs better for mutation position prediction on synonymous mutations in the `tang` data than any model trained on out of frame data. Performance plot is as before, but black vertical bar indicates the performance of the S5F model, and all models evaluated on synonymous mutations in the `tang` data.