PLOS Computational Biology

RESEARCH ARTICLE

Popformer: Learning general signatures of positive selection with a self-supervised transformer

Leon Zong[1,2], **Sorelle A. Friedler**[2], **Sara Mathieson**[1,2*]

1 Department of Biology, University of Pennsylvania, Philadelphia, Pennsylvania, United States of America, **2** Department of Computer Science, Haverford College, Haverford, Pennsylvania, United States of America

* smathi@sas.upenn.edu

OPEN ACCESS

Citation: Zong L, Friedler SA, Mathieson S (2026) Popformer: Learning general signatures of positive selection with a self-supervised transformer. PLoS Comput Biol 22(7): e1014492. https://doi.org/10.1371/journal.pcbi.1014492

Editor: Minyu Feng, Southwest University, CHINA

Received: March 6, 2026

Accepted: June 24, 2026

Published: July 15, 2026

Data availability statement: We have made all resources for this work publicly available, including checkpoints for the pre-trained Popformer-base model, the fine-tuned selection classification model, simulated datasets, and all code used for training and evaluation. All

Abstract

Understanding natural selection can help shed light on the genetics underpinning adaptive evolution. The widespread availability of large-scale human genetic variation data has led to the development of data-driven methods for detecting signatures of selection, many of which are based on deep learning. However, these methods often fail to generalize well to the diversity of selection signatures across a broad range of evolutionary scenarios. We propose a novel transformer-based model, Popformer, for learning encodings of general patterns of genetic variation. Popformer includes site-wise and haplotype-wise attention, allowing us to capture variation among both genetic positions and individuals. It additionally learns relative positional embeddings for inter-SNP distances. The model is pre-trained with an analog of the masked language modeling objective across a range of real human genomic data, similar to the task of genetic imputation. Using dimensionality reduction, we show that the pre-trained model learns meaningful embeddings of genomic windows that correspond with population structure. We also show that the model can accurately perform genotype imputation. We further fine-tune the model on selection classification and demonstrate that our model is more accurate than other selection classification methods, on selection simulations of both well-specified and mis-specified demographic models. Using a novel real data validation approach, we apply Popformer to human data from the 1000 Genomes Project and reveal its ability to generalize from simulated data to the diversity of real data. Overall, our method provides a new direction for population genetic inference methods, with future fine-tuning applications including the inference of recombination rates, introgression, and local ancestry.

resources can be found at https://popformer.leonzong.com.

Funding: This work is funded by National Institutes of Health (NIH, https://www.nih.gov/) grant R15HG011528 to SM and SF. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NIH. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

Author summary

Natural selection lies at the heart of adaptation and evolution. Identifying genomic signals of natural selection can lead to insights about evolutionary pressures acting on populations, and provide an understanding of how populations differentiate. Detecting these signals using only present-day genomic data is difficult, and methods developed for this task often use specialized simulations, trading generalization for power. We show here that adapting deep learning techniques from modern language and image processing can be used to identify signals of selection with both high power and robustness to misspecification. By pre-training our transformer model, Popformer, on real data, we induce learned model representations that are representative of the variation in real data. We fine-tune our model on simulations inferred from European, Asian, and African populations, and compare with other machine learning methods and summary statistics. Our results suggest that both our model architecture and our training regime are useful for strong performance in diverse simulation tests and in recovering known selection signatures in real data.

Introduction

Modern population genetics broadly aims to understand patterns of variation in and between populations at a genome-wide level. A major goal of the field is to identify regions of the genome that are or previously were under natural selection. A beneficial mutation can rapidly rise in frequency in a population in a process known as a selective sweep. While this process leaves behind distinct patterns of variation in the population due to linked selection, identifying these sweeps is challenging [1,2]. There are a number of confounding evolutionary effects, including those of demographic events and other stochastic forces like background selection and varying mutation and recombination rates [3–5]. Traditionally, theoretically motivated summary statistics are used to identify genetic regions which have undergone a selective sweep. These summary statistics identify a particular signature of selection; for example, reduced diversity (π, Tajima's D), long stretches of identical DNA (iHS), or are a composite of several of these statistics (SweepFinder, CMS) [6–10]. While these methods are widely used, they are often underpowered and unreliable when confounding evolutionary forces produce the same genetic signatures [4,11–13]. A new cohort of machine-learning based methods, many of which utilize convolutional neural networks (CNNs), have made progress in detecting selective sweeps with increased power and in the face of confounding evolution [14–21].

These new sweep detection methods have arisen in part due to the development of improved population genetics simulation programs including SLiM, discoal, and msms, which have the ability to model selective events [22–24]. With selection as a simulatable parameter, the methods are able to use a supervised training regime. Machine learning detection methods have proven to be powerful at detecting

selection in well-specified matched simulations, and many have even demonstrated robustness to simulation misspecification [25,26]. While success in the simulated regime is an important step, the ultimate goal of these methods is to discover signatures of selection in real genomic data. It is yet unclear to what extent models trained on simulations can generalize to real data. Simulations model theories of evolutionary processes, which are ultimately simplifications of the complex nature of real evolution.

Here, we introduce Popformer, a transformer-based model which learns general signatures of variation from real population genetic data. The model operates on sets of variants in the form of haplotype matrices. By training on real variants, the model is able to learn common patterns of variation that are derived from real evolutionary processes. The model is pre-trained with a version of the self-supervised masked language modeling objective, in which random positions in the input matrices are masked and the model learns to recover the value at the masked positions [27]. We demonstrate that the pre-trained model can perform a limited form of genotype imputation on par with widely used methods. We additionally show that pre-training produces model embeddings which are good representations of genetic variation.

The pre-trained model could further be trained (fine-tuned) for region-level tasks such as predicting signatures of selection, recombination rate, or mutation rate. In this work, our downstream focus is primarily selection detection. We introduce new, diverse simulated selection datasets for training and testing supervised selection detection methods. On these datasets, we demonstrate that our method can detect selection with increased power compared to both traditional summary statistics as well as recent deep learning methods trained on the same data. We find that Popformer can generalize better than other methods at detecting selection in out-of-distribution simulated test scenarios. Additionally, we use reported lists of real selected sites [28,29] as validation, and find that our model predicts enriched scores for these reported sites over baseline. As such, a trained Popformer model can be used out-of-the-box to perform selection detection on a wide range of populations without the need for per-dataset simulations and training.

Our transformer architecture provides several benefits over the CNN architectures used by previous neural network methods. First, the transformer is invariant to the ordering of the haplotypes, which can heavily influence the output of the CNN methods [20,30]. The attention mechanism used in the transformer also allows for potential dependencies between SNPs at longer ranges than can be captured by local convolutional layers. Finally, the architecture of the model additionally lends itself to SNP-level and haplotype-level prediction tasks, such as local ancestry inference or introgression detection. While we do not explore this in this work, we hope to facilitate these directions by releasing checkpoints for the pre-trained Popformer-base model, fine-tuned selection classification model, simulated datasets, along with all code used for training and evaluation. All resources can be found at https://popformer.leonzong.com.

Results

Overview of Popformer method

Popformer is an encoder-only transformer model which takes haplotype matrices as input and outputs a vector representation (embedding) of each position of the matrix (further details can be found in the Methods). The encoder uses an implementation of axial attention: where attention is computed both along the variants for each haplotype, and along the haplotypes for each variant (Fig 1) [31,32]. This allows the model to dynamically weight particular haplotype variants based on the full population and window context. Distances between variants are included as a learned positional attention bias, allowing the model to learn patterns of variant density.

During pre-training, the embedding representation is learned by training the model to unmask random positions in matrices derived from real human populations in the 1000 Genomes dataset [33]. We refer to this model as Popformer-base. We perform this pre-training primarily to learn the patterns of genetic variation in real data - population structure, patterns of LD, etc. The aim of the pre-training is to produce high-quality representations of this real variation that will enable better generalization performance on the downstream task of selection detection. To this end, we simulate a set of diverse genomic regions under human-inferred demographies, including both neutrally evolving regions and those

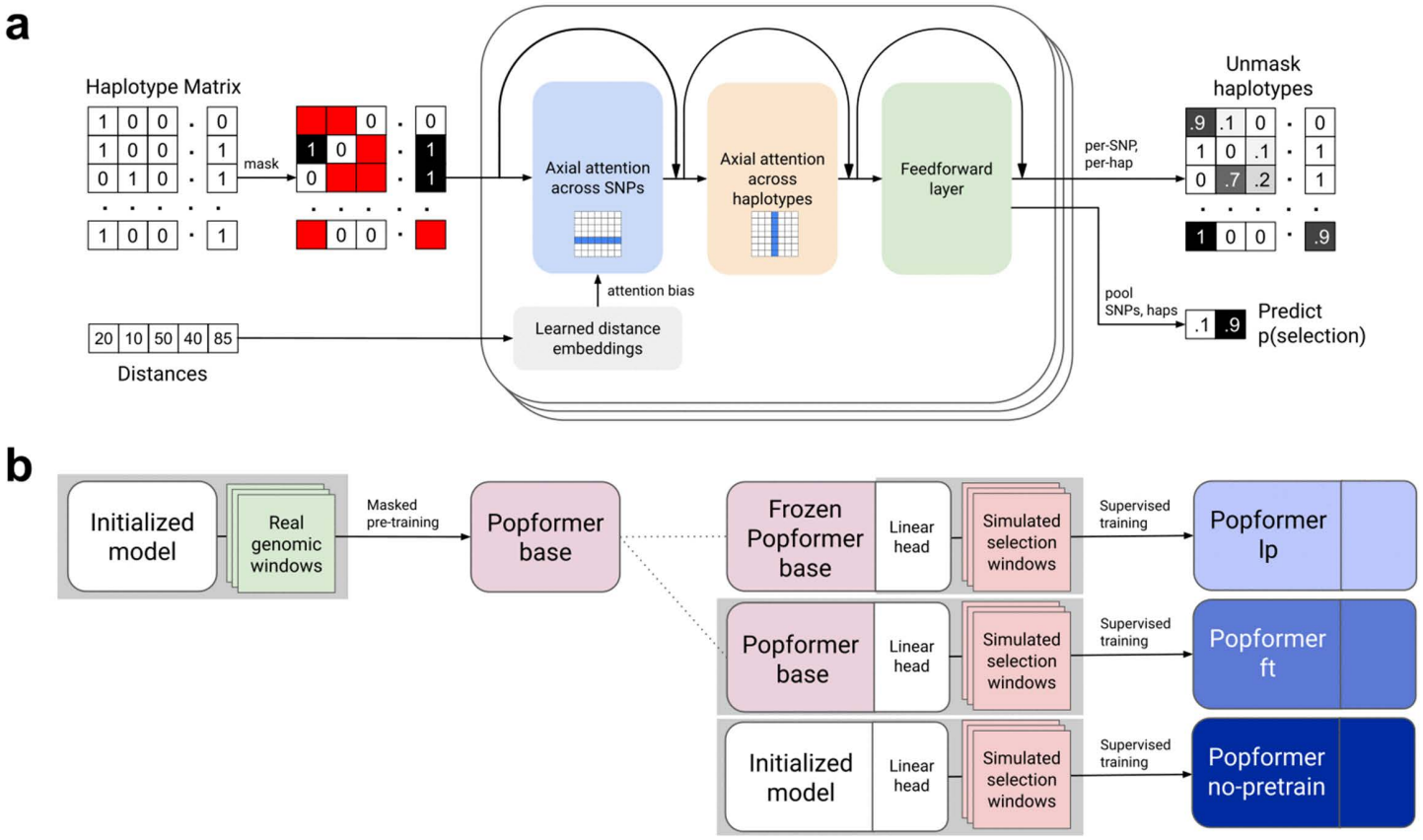

Fig 1. Overview of the architecture and training strategies for Popformer. a, overview of the architecture and pre-training strategy of Popformer. Haplotype matrices and distance vectors are input to the model. The model consists of multiple blocks of axial attention. Each block consecutively applies SNP-wise attention across every haplotype, haplotype-wise attention for every SNP, and a fully-connected feed-forward layer. Normalized residual connections are applied after each layer. Pre-training is self-supervised, using real genomic windows. Random tokens are replaced with a mask token, and the model is tasked with unmasking the token. **b**, Popformer-base is first trained using self-supervised unmasking. Using it, we evaluate three different selection training recipes. For Popformer-lp, the pre-trained base model is frozen and the head is trained on the static embeddings. For the Popformer-ft, gradients from training also flow through the base model. For Popformer-no-pretrain, we ablate the pre-training step, starting from a newly initialized model. The gray background indicates where gradients are computed.

https://doi.org/10.1371/journal.pcbi.1014492.g001

evolving under various selection strengths. We use these simulations to fine-tune a simple linear model (the classification head) on top of Popformer-base under several different training recipes. We evaluate:

1. A linear probe (Popformer-lp), where the base model (encoder) is frozen and only the linear head is allowed to update.
2. Full fine-tuning (Popformer-ft), where both the base model and the linear head are allowed to update.
3. A pre-training ablation (Popformer-no-pretrain), where a newly initialized encoder is used in place of the pre-trained base model. Here, both the encoder and the head are allowed to update.

Once trained, these models are verified on held-out test simulations, where the ground truth is known. Additionally, they can be used to perform windowed inference along real genomes.

Learned population embeddings capture informative genomic variation

Popformer-base is trained with an analogue of the masked language modeling task (see Methods) [27]. In this modeling scheme, the input sequences to be masked are haplotype matrices (S1 Fig). In order to ensure that our model accurately learns complex dependencies between SNPs, we first test the actual performance of our method on its training objective by computing the unmasking accuracy on a test set of 75%-uniformly-masked samples from 1000 Genomes populations. On this test, Popformer-base outperforms two simple baselines and performs quite well. The nearest neighbor baseline simply chooses the allele using the most similar windowed haplotype to the masked haplotype, and the allele frequency baseline chooses the most frequent unmasked allele (Table 1).

The task of unmasking haplotypes is also conceptually similar to the task of genetic imputation, whereby a 'target' panel of individuals has missing variant data imputed using a 'reference' panel. Our masked training differs from genotype imputation primarily in the nature of the missing data - in our masked training, there is not a set of complete haplotypes used as the 'reference'. Instead, all entries in the haplotype matrix were masked uniformly at random (see Methods and S1 Fig). To compare to methods specifically designed for genotype imputation, we formulated an imputation-like test to further validate our pre-training objective performance. 64 haplotypes are used as an unmasked reference panel, and 64 other haplotypes are used as a target panel, where a percentage of SNPs are randomly masked in the whole target panel (S2 Fig). This test setup is identical to classic genotype imputation, allowing us to compare to a widely used imputation method, IMPUTE5 [34].

We calculated the squared correlation R^2 using the posterior imputed allele probabilities for IMPUTE5 and Popformer by comparing the imputed allele at each site to the true allele (Fig 2) [35]. We additionally calculated error rate using the maximum posterior imputed allele. Comparing R^2, Popformer-base performs as well as IMPUTE5 [34], and continues to outperform both baselines. We find a similar result for the error rate, where our method imputes exact haplotypes comparably to IMPUTE5. A breakdown of the results by binned minor allele frequency demonstrates that all methods perform worse at very low allele frequency, with IMPUTE5 performing the best (S3 Fig). Of course, IMPUTE5 is designed for biobank-scale panels, and is therefore orders of magnitude more efficient than our method (S4 Fig). These results are quite promising given that our self-supervised training data has no special examples with a reference-vs-target masking pattern, and demonstrate that the pre-training strategy effectively learns patterns of genomic variation.

While trained on unmasking, Popformer is primarily designed for applications in downstream tasks. In order to achieve good performance on these tasks, it is important that the model's pre-training objective has produced an informative learned embedding. To inspect these representations, we first examined whether the embeddings adequately represent real genomic variation between populations at least at a continent-level scale. Individual embeddings were calculated for each individual in the 1000 Genomes dataset by concatenating mean-pooled window-level embeddings for sliding windows across chromosome 1 [33].

We performed a principal component analysis (PCA) on the these mean-pooled embeddings and used the first two components to visualize them (Fig 3a). Given that even raw allele frequency patterns reflect continental-level population

Table 1. Denoising accuracy for Popformer-base and two naive baselines.

Method	Accuracy
column frequency baseline	90.3 ± 2.96
nearest neighbor baseline	88.3 ± 2.49
popformer-base	$\mathbf{95.8 \pm 1.17}$

Mean and standard deviation of denoising accuracy for 100 test windows. 75% of positions are uniformly masked from haplotype matrices of size 128 SNPs by 128 haplotypes, as in S1 Fig.

https://doi.org/10.1371/journal.pcbi.1014492.t001

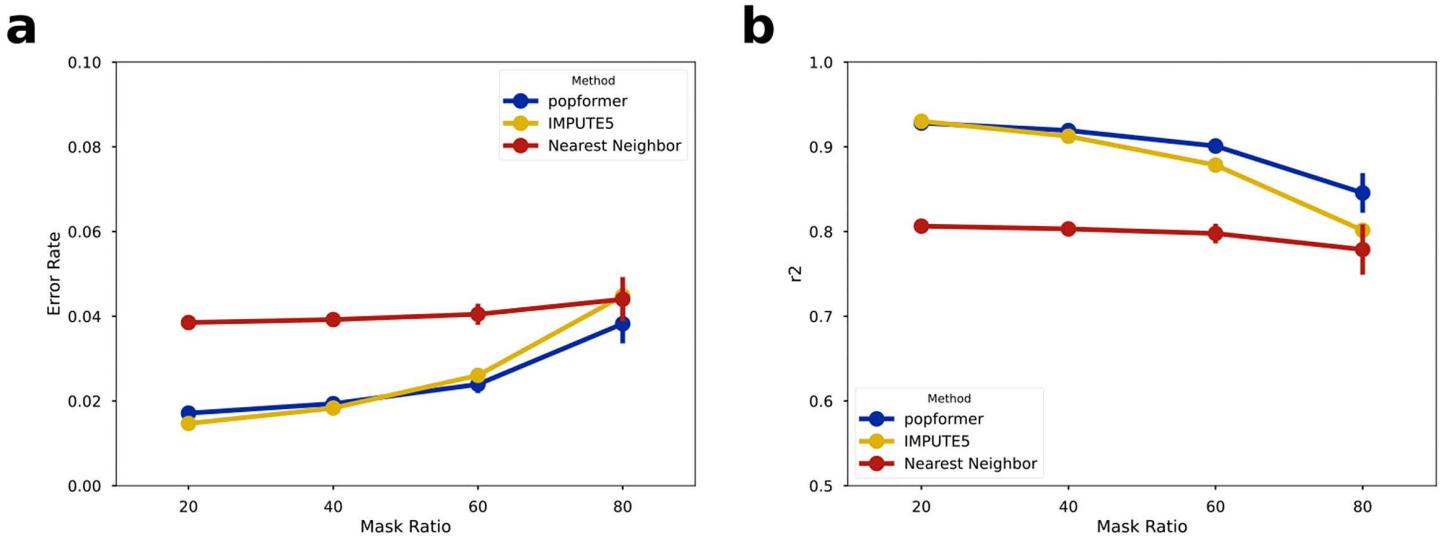

Fig 2. Imputation results for Popformer and other baselines. a, imputation error rate (lower is better) and **b**, squared correlation R^2 (higher is better) at four different imputation levels (20%, 40%, 60%, and 80%). We compare performance for Popformer, IMPUTE5, and the nearest neighbor baseline. The column frequency baseline is omitted, performing significantly worse than the other methods.

https://doi.org/10.1371/journal.pcbi.1014492.g002

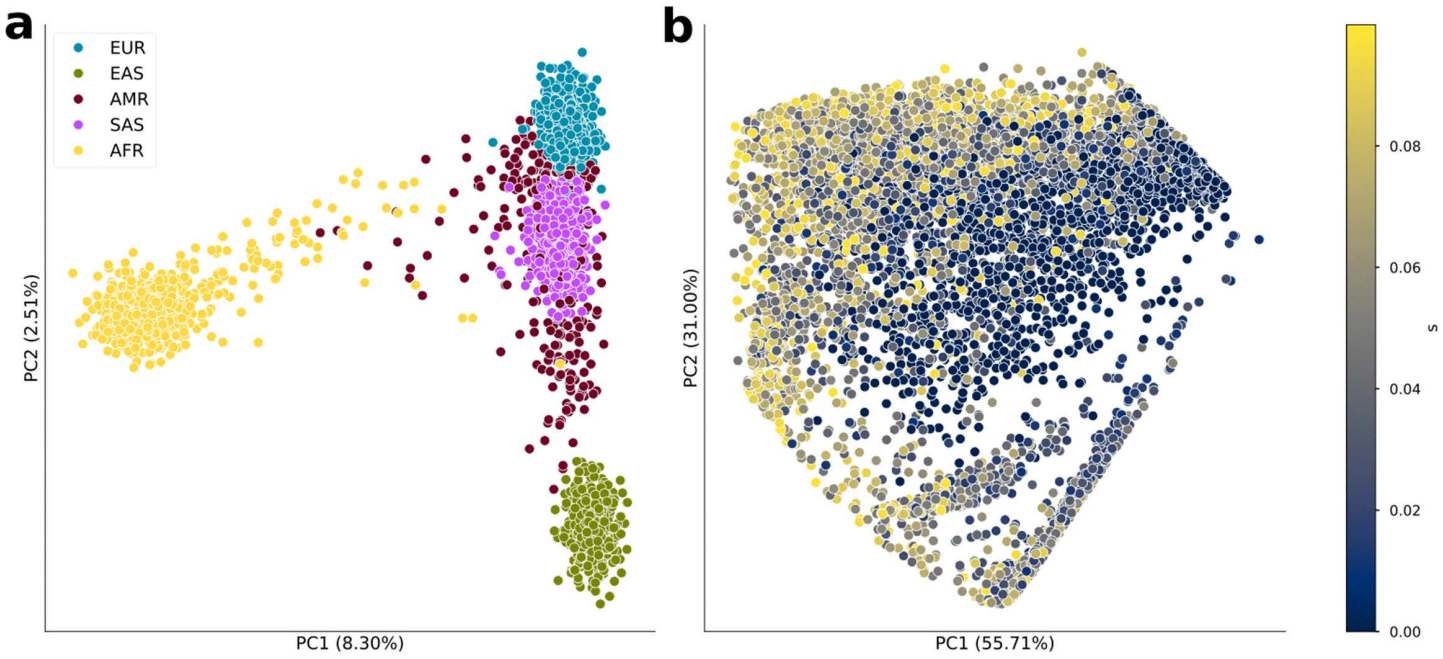

Fig 3. Visualization of Popformer's embeddings for real and simulated data. a, PCA reduction of region-averaged learned embeddings for chromosome 1 for all 1000 Genomes populations, with colors indicating superpopulations (continent-level variation). Points represent chromosome 1 embeddings for diploid individuals. Population codes: EUR (Europe), EAS (East Asia), AMR (America), SAS (South Asia), AFR (Africa). **b**, PCA reduction of window-level learned embeddings for simulated selection windows at a range of selection strengths. Points represent individual haplotype matrices.

https://doi.org/10.1371/journal.pcbi.1014492.g003

structure, we expect that Popformer's learned embeddings should also represent this structure. Indeed, Popformer-base's embeddings match commonly reported results in which dimensionality-reduced genomic variation differentiates continental populations [36,37].

Secondly, we aimed to identify whether the pre-training objective aligned at all with the task of selection detection. To do this, we calculated window-level embeddings for our selection simulations, which span a selection coefficient of 0-0.1 (Fig 3b). We observe that there is a clear gradient between low and high selection strengths. While not definitive, this indicates that the model's pre-training task is partially aligned with the selection classification objective.

Trained Popformer models can accurately detect selection

In the standard application of a supervised selection model, the model is trained on simulations that are designed to match the downstream real data. In our case, we trained all of the supervised models on a balanced mix of human European-inferred (CEU), East Asian-inferred (CHB), and African-inferred (YRI) demographic simulations (for details on the simulations, other models, and training see Methods). In this study, we only aimed to detect positive selection, but training data encompassing negative or balancing selection could also be used. Each model classifies these simulated haplotype windows as either containing a selected-for mutation or not. We additionally include shouldering windows from selection simulations as negatives, aiming to improve localization power.

We report AUC curves on a held-out test set of CEU, CHB, and YRI simulations, and find that our method, in all training configurations, outperforms other models (fine-tuned, CEU: AUC = 0.93) (Fig 4a). We additionally stratify the metrics by selection strength and by whether windows are shoulders (to a window containing a selected allele) and find that stronger selection is easier to detect, except for when the window is in a shoulder region of a stronger selected allele. These shoulders of strong selection are likely a strong confounding signal (S5 Fig).

While these results demonstrate the effectiveness of our model's architecture at classifying selection from in-distribution simulations, the true benefits of the pre-training strategy are in the method's ability to generalize to out-of-distribution (OOD) data. There are several types of distribution shifts that may occur in the simulation-based inference pipeline. First, the demographic history used to simulate the training data may not match the real data. Second, the real data may have various effects (genotyping error, phasing error, population structure) that are not typically simulated.

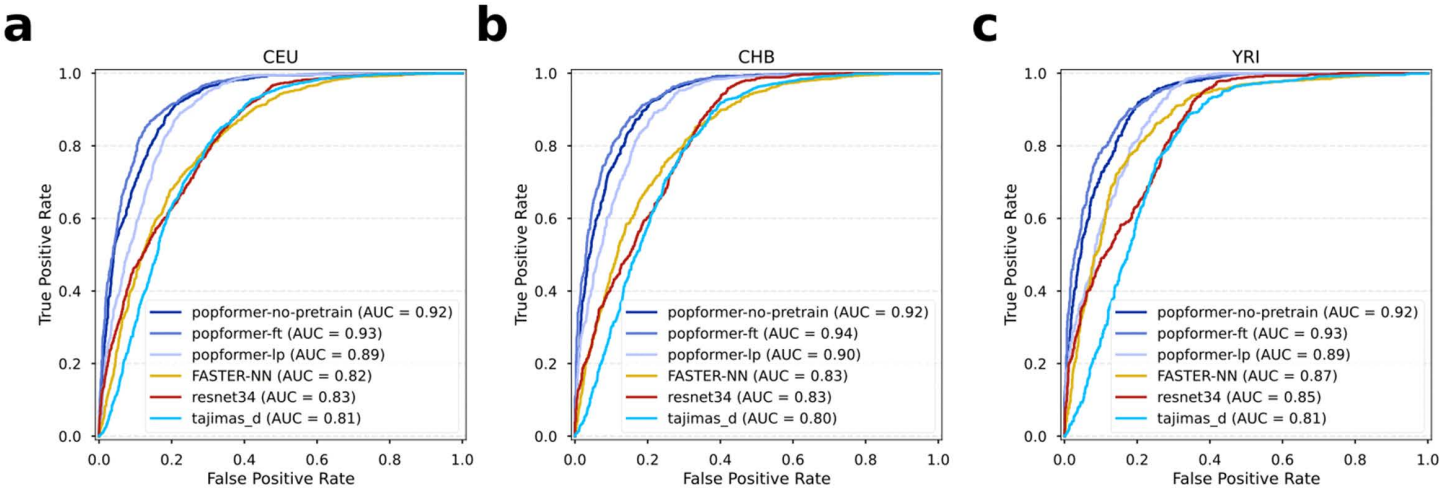

Fig 4. Selection detection results on simulated human demographies. a, ROC curves for our Popformer models, FASTER-NN CNN model, resnet34 CNN, and a summary-statistic (Tajima's D) score, evaluated on a held-out test set of CEU-inferred demographic simulations. **b**, for a test set of CHB-inferred demographies. **c**, for a test set of YRI-inferred demographies.

https://doi.org/10.1371/journal.pcbi.1014492.g004

We performed several experiments to test the mis-specified demographic history case. In order to isolate the effects of the pre-training, we evaluated all methods on a set of simulations with varying bottleneck strengths and a set of simulations with different constant population sizes. Instead of using Popformer-base (pre-trained on real data) as the pre-trained model, we pre-train new base Popformer models. Then, we train on the selection task. These experiments therefore parallel the real-data pipeline, but with entirely simulated data. For the bottleneck experiment, we pre-train a base Popformer model on a 10% bottleneck, train all models (including fine-tuning and linear probe variants) on no bottleneck, and evaluate on varying strengths of bottlenecks (Fig 5a). For the constant population size experiment, we pre-train on N = 10000, train all models on N = 100000, and evaluate on varying constant population sizes (Fig 5b).

We find the bottleneck example to be especially informative. Since bottlenecks can leave similar signatures to natural selection in the genome, we expect that regions in stronger bottlenecks are harder to classify. We see this decrease in performance for all models. The linear probe (with the pre-trained model from data from a 10% bottleneck), though, generalizes better both to the matching 10% bottleneck data and to more extreme bottlenecks. This result also follows in the varying constant sized simulations experiment, with the linear probe model generalizing nearly as well as Tajima's D. This finding strongly indicates that the pre-training assists in generalization ability.

The fine-tuned model only clearly outperforms the pre-training ablation in the bottleneck experiment, and does not outperform the linear probe model. We hypothesize that allowing the encoder to update, as in the fine-tuned model, likely leads to some 'forgetting' of the pre-training demography. In the varying constant population size experiment, the order of results is not clear. For example, the orders of the Popformer fine-tuned and pre-training ablations are inconsistent. These inconclusive results may be indicative of instability in training (only one model was trained for each configuration) or that the models are 'overfit' to a particular feature of the simulations (e.g., if lower diversity is the only signal needed in the training dataset, the model may not learn to use haplotype structure).

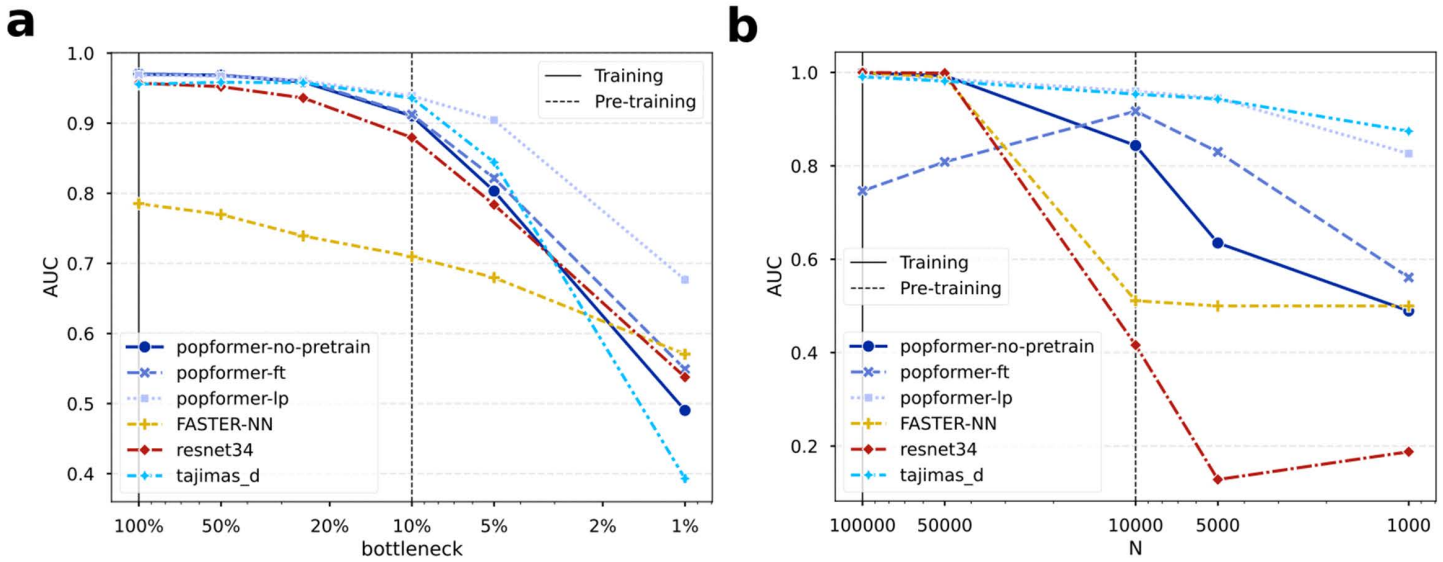

Fig 5. Selection detection results for models trained on varying demographic scenarios. Solid vertical lines indicate training demographic configuration and dashed lines indicate pre-training demographic configuration (for popformer-ft and popformer-lp). For all results, bootstrapping with N = 100 resamples was applied. The bootstrapped mean and 1SD error bars for the area under the ROC curve (AUC) are shown. In most cases, the error bars are smaller than the points. **a**, varying strengths of bottlenecks, models trained on no bottleneck. **b**, varying constant population sizes, models trained on N = 100000.

https://doi.org/10.1371/journal.pcbi.1014492.g005

Overall, Popformer models outperform the other methods we tested on in-distribution simulations, and pre-trained models demonstrate improved generalization on out-of-distribution tests. These results show that the pre-training of our models is beneficial in the case that the distribution shift is related to mis-specified demography. We additionally hypothesized that pre-training might also be beneficial in a limited-labeled-data regime, and find this to be true at very low numbers of training examples (S6 Fig). The linear probe and fine-tuned Popformer models outperform the other deep learning methods when only 10 simulated windows are used to train the models. In practice, though, more simulated data should be used to train the models.

Validation on known selected regions

Though we display promising results on simulated tests, ultimately, real evolutionary effects encompass a wider range of variation then we can simulate. As mentioned above, real evolution encompasses both categories of distribution shifts: we are both 1) unable to know the true demography (to create ideal simulations) and 2) unable to capture the range of variation caused by other effects (technical, biases, etc). Despite this, methods are still validated on real data by inferring selection on a list of known or potentially novel sites, and examining whether predictions match the literature. We follow this tradition and analyze the commonly used LCT/MCM6 region (S7 Fig), which has a strong signature of positive selection in European populations [38].

However, this approach is prone to bias, in the form of cherry-picking a set of regions that a method identifies well, and does not provide any information on the false positive rate of a method. This approach is also uninformative on how powerful a model is, as any signals which are already well known are likely strong. We propose that selection detection methods, which are potentially most well suited for discovery (hypothesis generation), should focus heavily on controlling the false positive rate. The major difficulty with this is the fact that it is impossible to know the full ground truth for selection in the human genome.

Here, we introduce two approaches to validating selection detection methods on real data which can be used to compare methods on both power and false positive rate. First, new ancient DNA data and methodology has allowed for selection inference using real allele frequency trajectories [39]. Using these results, we gather a list of putatively neutral regions ($X < 2$) from Akbari *et al.* [29] to use as 'true negatives'. By using the regions from Grossman *et al.* [28] as 'true positives', we can generate a pseudo-ROC-curve with the goal of comparing how well methods can differentiate between selected regions and putatively negative regions. Crucially, in this framework, we treat the Grossman *et al.* as a correct, but incomplete list. It is then possible that some methods may predict other regions correctly that are missing from our positive set, thus leading to a misleading comparison.

Along these lines, the composite of multiple signals (CMS) test used by Grossman *et al.* presents a circular validation issue. Since the CMS method uses specific patterns of variation (long haplotypes, higher differentiation, lower diversity), a method which can better recover the Grossman *et al.* list might merely be making predictions which correlate with the CMS method. This circularity especially affects Tajima's D, which indeed correlates directly with some of the CMS signals.

For most populations, which lack sufficient ancient DNA, we can instead compare [28]'s true positives with the fraction of the genome predicted to be under selection, under the assumption that a majority of the genome is neutrally evolving. Both of these approaches can be interpreted as analogous to a standard ROC curve, allowing for simple comparisons. These approaches are also independent of binary prediction threshold, which is important for deep learning methods, whose outputs are rarely calibrated.

Using these approaches, we find that the deep learning methods' strong performances on simulated data do not translate well to performance on real data (Fig 6). Notably, Tajima's D is the method which typically aligns best with the list of selection regions from [28], with the deep learning methods falling behind. This is a worrying result for deep learning methods, which are significantly more computationally expensive relative to the simpler summary statistics. Of the deep learning methods, however, Popformer generally recovers the known regions the best. Crucially, our Popformer method

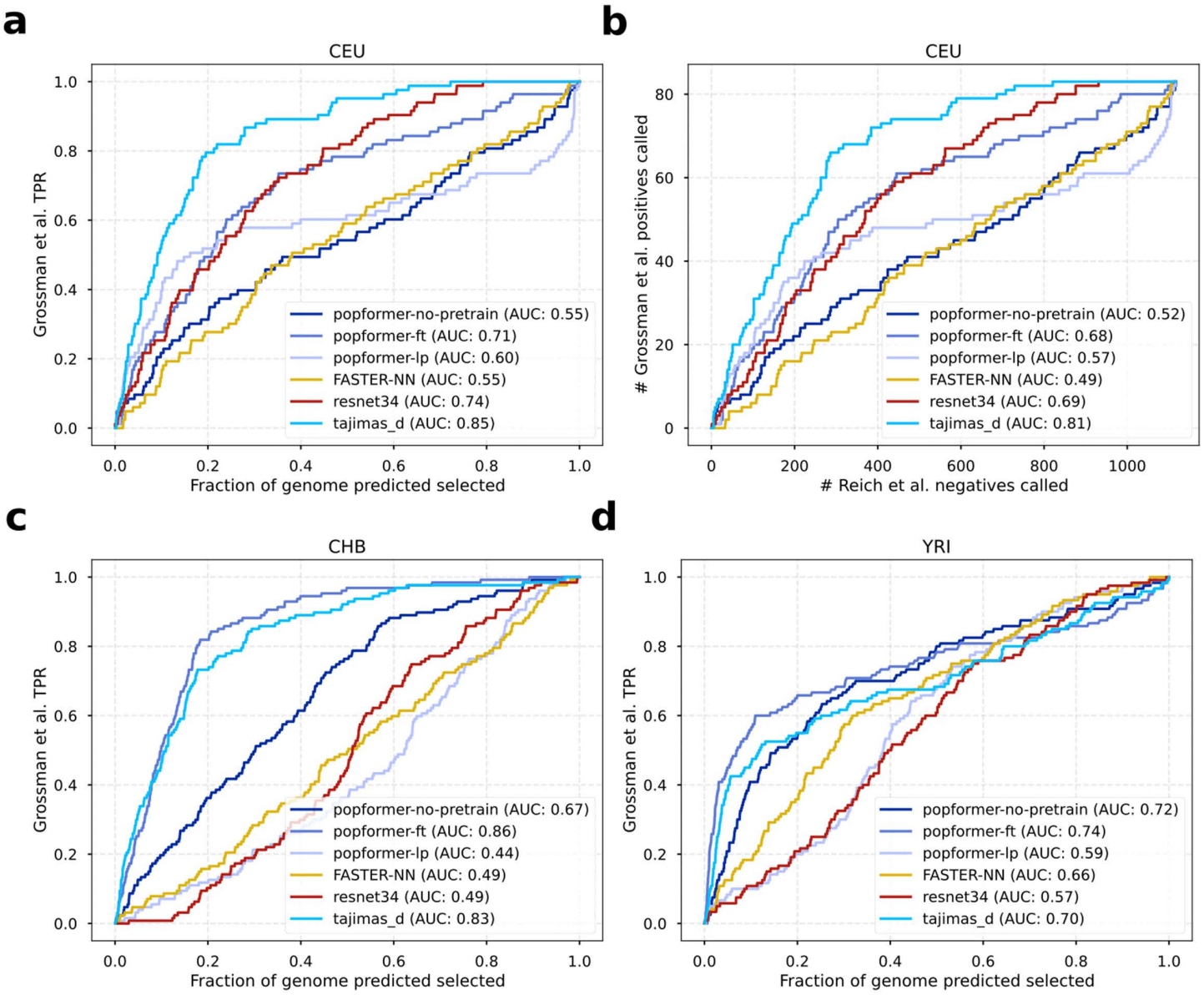

Fig 6. Validation results of selection detection on real data. a, all models evaluated on recovery of Grossman *et al.* [28] known selection regions in the CEU population versus fraction of the genome predicted to be under selection. **b**, models evaluated on recovery of Grossman *et al.* CEU regions versus false discovery of Reich *et al.* negative regions. **c**, same as **a**, but on the CHB population **d**, same as **a**, but on the YRI population.

https://doi.org/10.1371/journal.pcbi.1014492.g006

outperforms all other methods, including Tajima's D, at recovering known positive regions in the East Asian (CHB) and Yoruban (YRI) populations. This result only holds in the fine-tuned Popformer training recipe, not in the pre-training ablation, suggesting that real pre-training does provide an improvement when applying downstream models to real genomes.

The linear probe, though, performs worse than the other Popformer variants, which conflicts with the pattern of results from the varying simulated evaluation. Although speculation, we believe that for the simulated demographic misspecification, updating the encoder likely caused some degree of demographic forgetting and causing reduced performance. For the real data, the simulations match the real data in demography. Instead, they are mismatched due to other

effects. In this case, updating the encoder may have been necessary for the model's ability to generalize across the simulation-real distribution shift without causing any forgetting.

These results are representative of the fact that the selection signatures can manifest differently under different demographic models, and suggest that deep learning methods may be learning specific, not general, signatures. To begin to understand which, we examine correlations between Popformer-ft's model scores and summary statistics, both for simulated (S8 Fig) and real (S9 Fig) datasets. There are some moderately sized correlations, which are expected (Spearman rho: Tajima's D = -0.61, # singletons = 0.56).

We also find evidence to support that using non-reported-positive regions as neutral regions is valid. Using inferred neutral alleles from ancient DNA as true negatives strikingly resembles using the fraction of the genome called (Fig 6a, 6b). Thus, if one is confident that their list of selected regions are true, using the rest of the genome as a neutral baseline in a test for recovery is likely to be informative. These analyses do not take into account potentially undiscovered selected regions in the genome – but, given that the reported regions are likely to be strong signals, detection methods should be able to predict them at higher probability than weaker and neutral signals.

Discussion

Learning signatures of natural selection using supervised machine learning can improve selection detection power over traditional summary-statistic based methods [14,16,17,40]. However, all simulation-based methods suffer when the simulations are poorly matched to the real data. In contrast, our method uses a modern paradigm for self-supervised learning in order to first learn important features of variation in real genomic data before fine-tuning on a particular supervised inference task. In addition, Popformer's transformer-based architecture allows for more complex learned representations of genomic windows, including both relationships between SNPs and between haplotypes. Together, the self-supervised pre-training and the novel architecture enable Popformer to capture informative signals of selection. We demonstrate a higher windowed classification performance against two other deep-learning methods and a summary statistic score method.

Despite using self-supervised pre-training on real data, Popformer is still reliant on simulations to provide a training dataset for selection. Any such supervised method will have inherent data biases due to the necessary nature of selecting prior distributions on demographic and evolutionary simulation parameters. We aimed to mitigate this in our simulations by using a wide prior for our training simulations that encompasses a variety of possible human-like evolutionary histories. However, the main idea behind pre-training on real data is aimed to tackle this problem. Therefore, we tested on several human-like simulation sets and a range of varying demographic simulations and found that Popformer achieves a level of generalization to out-of-distribution data superior to the other tested deep-learning methods. Ultimately, we are most interested in applying Popformer to real genomic datasets. We demonstrated that our method can modestly recover previously reported results of selection in three distinct human populations, but generalizes well to some populations (YRI) despite poorly matched training data (CEU-inferred). We believe that empirical validation of the results of a simulation-based selection method is highly important to assessing its usefulness and propose our validation strategy as one way to achieve this.

We believe that the Popformer architecture and self-supervised training strategy are an important contribution to the field of deep-learning selection inference. There do however remain many avenues for potential developments. Many approaches to unsupervised and self-supervised pre-training have been developed, including autoencoders and contrastive learning [41,42]. As our findings on masked pre-training are not definitive, these approaches might be key to developing higher quality latent representations of population genomics data. Regarding genomic data and its representation, an input scheme where additional information (say, nucleotide variation tokens like '*A>T*' [43] or embeddings from a genomic language model [44]) might allow for a complex transformer architecture like Popformer to better capture patterns of variation. Selection detection additionally depends on context, up to an extent. Efficient linear attention implementations, of

which there are numerous, might allow for longer windows and potentially improved performance [45,46]. Another future direction could be better quantification of uncertainty, which could be accomplished through recent advances in neural posterior estimation (NPE) [47,48]. Finally, the masked haplotype reconstruction task could be repurposed downstream to handle missing input data, given an appropriate fine-tuning setup.

While used here only in the context of selection classification, the idea of learning signatures of variation in real data could easily be applied to other population-level (say, by averaging across windows) and window-level inference tasks in population genetics, including inference of demographic parameters, recombination hotspots, and mutation rates. These examples would be plug-and-play given a set of simulated training data and a pre-trained model, and would likely demonstrate the same robustness to mismatched data generation and assumptions as we found for selection. In addition, we believe that the SNP-level and haplotype-level attention architecture we introduce with Popformer would lend itself well to SNP-level population genetics tasks, including local ancestry inference or archaic introgression detection. Notably, Popformer could feasibly infer selection directly at a per-SNP level rather than a windowed level, allowing for greatly increased resolution. In fact, the pre-training masked language model objective we used might be more well suited to the granularity of these tasks. We expect that continual advancements in simulations and advances in genomic data quality and scale will reveal new applications and benefits of self-supervised deep-learning methods in population genetics.

Materials and methods

Details of the Popformer architecture

Popformer, in a similar manner to other raw-data deep-learning methods, operates on matrices $\mathbf{x} \in \{0, 1\}^{n \times S}$ of n haplotype rows by S single nucleotide polymorphism (SNP) columns derived from genomic windows. Only biallelic SNPs are included. These matrices are computed in the form of binary matrices in which 0 represents the more common (major) allele and 1 represents the less common (minor) allele. We also include a vector $\boldsymbol{d} \in \mathbb{N}^S$ encoding distances between SNPs, which allows the model to incorporate positional information in its predictions. Unlike most CNN methods, the transformer architecture can handle both a variable number of haplotype rows and SNP columns without additional padding. This renders Popformer more flexible when performing inference on populations that have a different number of individuals.

An overview of the architecture can be found in Fig 1. We adopted the model architecture from the axial attention transformer architecture, originally developed for image learning and later for protein sequence learning by MSA Transformer [31,32]. Attention is a widely-used method that allows each element in an input sequence to be dynamically weighted (attended to) based on every other element in the sequence. For example, text tokens in a sentence can each be weighted based on the context from the full sentence. 2D axial attention, originally designed for image methods, extends the standard mechanism of attention, which operates on 1-dimensional sequences, to operate consecutively across multiple dimensions [31]. Column-wise attention (in our case representing structure across haplotypes) is applied for every SNP with complexity $O(Sn^2)$. We use tied row attention, introduced by [32], to share the row-wise self attention maps between every haplotype in the matrix. This forces the model to learn one pattern of SNP dependency, which maps directly between different haplotypes. It also reduces memory usage from $O(S^2n)$ to $O(S^2)$. Following most transformer architectures, we first use an embedding layer to learn representations of our input tokens (major allele, minor allele, mask, start, end, and pad tokens). We include residual connections around the attention layers [49]. In all, each axial attention block consists of a row-attention block, a column attention block, and a feedforward dense network, allowing us to aggregate information about variant and population structure.

In the standard self-attention implementation, a positional embedding of a number of forms can be used to break standard attention's positional-invariance. These forms all assume that 'distances' between consecutive tokens are equal. Unlike protein and image sequences, adjacent columns of SNPs in the input matrices do not represent consecutive positions within the original genome. The distances between variants provide SNP density information otherwise lost in a

pure haplotype matrix. In order to encode the meaningful patterns of inter-SNP distances, we include an altered form of the T5 language model's learned positional embedding approach [50]. Inter-SNP genomic distances (in base pairs) are first binned into a fixed number of linearly-spaced distance buckets. An embedding is learned for each bucket separately for each row-attention head and for each layer. The embedding directly modifies the attention matrix in the form of an added bias. We adapt T5's 32 learned positional embeddings, using 31 linearly-spaced discrete distance buckets from 0 to 50k base pairs, with a final 32nd bucket reserved for any larger distances. We do not include a positional embedding for haplotypes, as the positional-invariance in this dimension allows our method to be insensitive to the ordering of the haplotypes.

Pre-training on real genomes

Our pre-trained model (Popformer-base) is trained with an analogue of the masked language modeling objective. Given masked positions $M \subseteq \{1 \dots n\} \times \{1 \dots S\}$ using which the masked input $\tilde{\mathbf{x}}$ are created, we aim to find model parameters θ which minimize the binary cross-entropy loss:

$$\mathcal{L}(\theta; \mathbf{x}, d, M) = -\sum_{(i,s)\in M} \log p_\theta(x_{i,s} \mid \tilde{\mathbf{x}}, d).$$

The model outputs logits for the major and minor alleles for every position, while the loss is computed only for masked tokens. We do not mask d, the distance vector. In our case, this masking is applied uniformly across all SNPs (S1 Fig). We use a high masking rate (75%) to prevent models from learning simple extrapolation from neighboring haplotypes or alleles, as in [51]. We experimented with additionally masking columns of SNPs, blocking the model from using context from different haplotypes, but find that this degrades downstream model performance. We additionally tried to weight rare variants by oversampling minor alleles (1s) during masking. This resulted in poor performance on validation sets as the model learned from an incorrect distribution of major/minor alleles.

The training data for Popformer's pre-training contained 260,000 haplotype windows, sampled uniformly from all 26 populations of the 1000 Genomes phase 3 release [33]. A random population and start position along the genome are chosen. We find the nearest SNP to the start position, and perform basic quality control by reselecting a new window if less than 50% of the bases in the window are in the 1000 Genomes phase 3 strict accessibility mask. For each window, we take variants within 50,000 base pairs from the start position (only including biallelic SNPs) and take a number of haplotypes uniformly at random between 32 and 128. Thus, haplotypes within a window are guaranteed to be from the same population but their order and number is randomly selected. This pre-training data likely also contains windows with real, unlabeled selective sweeps, which might contribute to the real variation the model learns.

For all Popformer results, we used a model with four hidden layers, with each layer containing four self-attention heads each for row- and column- attention (34M parameters in 16-bit floating-point precision). This model is relatively smaller than similar models used in the vision, protein, and language domains, as we both have a less complex data representation and less training data. Cross-entropy loss was used during both pre-training (at the per-SNP level) and during fine-tuning (at the window level). The Adam optimizer was used with a linear learning rate decay and a warmup period of 0.1 epochs. This pre-trained model (Popformer-base) is used as the base model for the downstream selection task by adding a simple linear classification head. We tested two downstream training strategies: full fine-tuning, in which all the weights of the model (including the transformer encoder) are allowed to update, or training only the linear head (with the encoder frozen, referred to as linear probing). We additionally tested a model with ablated pre-training, in which we trained a randomly initialized encoder and linear head from scratch on the selection detection task. For these downstream selection-trained results, we trained on windows of length 50kbp (resulting in windows with variable numbers of SNPs) and subsampled to 64 random haplotypes. We note that this data configuration falls within the ranges used during pre-training.

In all, the pre-training took 6 hours to complete on a 4xB200 GPU node with 128 GB of memory. Fine-tuning took 1 hour to complete on a single RTX 5000 GPU (32 GB). Inference over 49,000 64-haplotype 50kbp windows (genome length) took 7 minutes to complete on the same RTX 5000 GPU. During pre-training, selection training, and selection inference, we used effective batch sizes of 32, 8, and 1, respectively. We observed no degradation of performance performing training and inference on different batch sizes. The masking strategy also differed: during pre-training we used 75% random masking. For the imputation comparison, we used an unmasked reference panel and random SNP masking for the target panel. For selection, we used no masking.

Selection simulations

We used the `SLiM` simulation framework through the high-level `stdpopsim` library to generate samples for training and testing [22,52,53]. To cover a wide range of demographic parameters, we use demographic parameters randomly sampled from inferred demographies for CEU, CHB, and YRI populations using `pg-gan`, a GAN-based approach to demographic inference [54]. We filter sets of demographic parameters which have growth rate < 0.01 for ease of simulation. A summary of demographies used is in S1 Table. We sample recombination rate uniformly at random from the HapMap recombination map and sample mutation rate randomly in $\mu \sim U(1.29 \times 10^{-8}/2, 1.29 \times 10^{-8} \times 2)$ [55]. We simulated a total of 5,000 neutrally evolving regions and 5,000 regions with a centered beneficial allele, all of length 250kbp. We randomly varied the strength of selection in $s \sim U(0.01, 0.1)$, and chose an onset time such that the allele would roughly reach fixation by the end of the simulation using Equation A17 of [56].

For each neutrally evolving sample, we randomly choose 10 sub-windows, all of which are labeled as neutral. For each selected sample, we randomly choose 10 sub-windows not containing the selected allele to be labeled as neutral (shoulders). We choose 20 sub-windows containing the selected allele to be labeled as positive. In all, this gives us 100,000 total neutral windows and 100,000 total positive windows. These are split first into populations, and for each population into an 80/20 train/test split. In all, these human-inferred simulations took 5 hours on 20 CPUs and storage of raw genotypes took 10GB on disk.

We used the discoal simulation engine to generate the varying simulations of bottlenecks and constant population sizes. We simulated a total of 10,000 neutral regions and 10,000 regions with a centered beneficial allele of length 50kbp. The strength of selection was randomly varied in $s \sim U(0.001, 0.1)$ and the final selected allele frequency was set to 0.95. For the bottlenecks, we used a ancestral population size of 10000, with a bottleneck of varying strengths from between 2,000 and 1,000 generations before the present. population sizes, we simulated populations of varying constant population sizes. For the training data size test, we used simulations of constant $N = 10000$ for both pre-training (unlabeled) and training (labeled).

Selection methods

We found it extremely important to implement fair model comparisons. Most machine learning selection classifiers are not designed to be used out-of-the-box, but rather trained from scratch on a target demographic model. Thus, we designed an evaluation harness for training and evaluating window-based classification methods that facilitates making fair model comparisons as we did in this work. This harness was also designed to be extensible, allowing for additional models to be evaluated with minimal effort (besides the effort to implement and re-train), and is released along with the models at https://popformer.leonzong.com.

We compared our method with two deep-learning methods and Tajima's D. A widely used architecture for selection detection deep-learning models are convolutional neural networks (CNNs). CNNs are naturally suited to the haplotype input data, both due to the input shape and due to the fact that learning convolutional filters aggregates local patterns of diversity in an appropriate way for discovering local signatures of sweeps.

FASTER-NN is a 1D-CNN that takes as input 2 1D vectors - minor allele frequencies and distances [40]. We use the same training recipe as was used in the original FASTER-NN method, except using our training simulations. We additionally compared against a ResNet architecture, which are CNNs commonly used for images. For this model, we adopt the preprocessing steps from [16,57] in which we sort haplotypes based on genetic similarity. This model was trained with a batch size of 16 and learning rate of 1e-4 for around 20 epochs, at which point validation loss stopped improving. For both models, we used a width of 512 SNPs (matching Popformer's input length) and used zero-padding for inputs shorter than this length. The ResNet model had a maximum height of 256 haplotypes, also using zero-padding.

Supporting information

S1 Fig. An example masked window. From top to bottom: 75% uniformly masked input, predicted output, and ground truth labels. These reflect Popformer's self-supervised pre-training process. Loss is computed between the predicted and ground truth for masked positions and used to update the model's weights.
(TIF)

S2 Fig. Example genotype imputation test window. The first 64 haplotypes are in the "reference" panel and are unmasked, and the next 64 haplotypes are the "target" panel and are randomly masked.
(TIF)

S3 Fig. Squared correlation R^2 by minor allele frequency. Points indicate the left edge of the MAF bin. For example, the first point represents the MAF bin (0, 0.05].
(TIF)

S4 Fig. Wall clock time for imputation methods, in seconds. Imputation performed for chromosome 20, 64 reference + 64 target haplotypes.
(TIF)

S5 Fig. Stratified performance of all models. ROC curves for models on held-out CEU-inferred test set, stratified by selection strength (columns) and by shoulders (rows, included shoulder regions or excluded).
(TIF)

S6 Fig. Performance of models at varying training data amounts. Area under the ROC curve (AUC) of all models at varying percentages of training dataset used. All models were trained to convergence.
(TIF)

S7 Fig. Windowed predictions around the LCT/MCM6 allele in the CEU population. The highlighted region represents the selection region reported in [28]. Each prediction line is for a 50kbp window.
(TIF)

S8 Fig. Correlations between Popformer scores and simple summary statistics in simulations. Scatterplots of Popformer-ft scores and Tajima's D, # of singletons, and number of SNPs in window (SNP density), on held-out CEU-inferred test set. Points represent windows and are colored by selection strength.
(TIF)

S9 Fig. Correlations between Popformer scores and simple summary statistics. Scatterplots of Popformer-ft scores and Tajima's D, # of singletons, and number of SNPs in region (SNP density), on all real-CEU-genome windows.
(TIF)

S1 Table. Parameter ranges used for per-population inferred human demographic simulation. 20 total demographic models for each population were inferred, and only those with growth rate < 0.01 are used as possible demographies for our simulations.
(XLSX)

Author contributions

Conceptualization: Leon Zong, Sara Mathieson.

Data curation: Leon Zong.

Formal analysis: Leon Zong.

Funding acquisition: Sara Mathieson.

Methodology: Leon Zong, Sorelle A. Friedler, Sara Mathieson.

Project administration: Sorelle A. Friedler, Sara Mathieson.

Resources: Sorelle A. Friedler, Sara Mathieson.

Software: Leon Zong.

Supervision: Sorelle A. Friedler, Sara Mathieson.

Validation: Leon Zong.

Visualization: Leon Zong.

Writing – original draft: Leon Zong.

Writing – review & editing: Leon Zong, Sorelle A. Friedler, Sara Mathieson.

References

1. Stephan W. Selective sweeps. Genetics. 2019;211(1):5–13. https://doi.org/10.1534/genetics.118.301319
2. Nielsen R, Hellmann I, Hubisz M, Bustamante C, Clark AG. Recent and ongoing selection in the human genome. Nat Rev Genet. 2007;8(11):857–68. https://doi.org/10.1038/nrg2187 PMID: 17943193
3. Dilber E, Terhorst J. Robust detection of natural selection using a probabilistic model of tree imbalance. Genetics. 2022;220(3):iyac009. https://doi.org/10.1093/genetics/iyac009 PMID: 35100408
4. Stajich JE, Hahn MW. Disentangling the effects of demography and selection in human history. Mol Biol Evol. 2005;22(1):63–73. https://doi.org/10.1093/molbev/msh252 PMID: 15356276
5. Charlesworth B. The role of background selection in shaping patterns of molecular evolution and variation: evidence from variability on the Drosophila X chromosome. Genetics. 2012;191(5):233–46. https://doi.org/10.1534/genetics.111.138073
6. Tajima F. Statistical method for testing the neutral mutation hypothesis by DNA polymorphism. Genetics. 1989;123(3):585–95. https://doi.org/10.1093/genetics/123.3.585 PMID: 2513255
7. Voight BF, Kudaravalli S, Wen X, Pritchard JK. A map of recent positive selection in the human genome. PLoS Biol. 2006;4(3):e72. https://doi.org/10.1371/journal.pbio.0040072 PMID: 16494531
8. Nielsen R, Williamson S, Kim Y, Hubisz MJ, Clark AG, Bustamante C. Genomic scans for selective sweeps using SNP data. Genome Res. 2005;15(11):1566–75. https://doi.org/10.1101/gr.4252305 PMID: 16251466
9. DeGiorgio M, Huber CD, Hubisz MJ, Hellmann I, Nielsen R. SweepFinder2: increased sensitivity, robustness and flexibility. Bioinformatics. 2016;32(12):1895–7. https://doi.org/10.1093/bioinformatics/btw051 PMID: 27153702
10. Grossman SR, Shlyakhter I, Karlsson EK, Byrne EH, Morales S, Frieden G, et al. A composite of multiple signals distinguishes causal variants in regions of positive selection. Science. 2010;327(5967):883–6. https://doi.org/10.1126/science.1183863 PMID: 20056855
11. Simonsen KL, Churchill GA, Aquadro CF. Properties of statistical tests of neutrality for DNA polymorphism data. Genetics. 1995;141(1):413–29. https://doi.org/10.1093/genetics/141.1.413 PMID: 8536987
12. Akey JM. Constructing genomic maps of positive selection in humans: where do we go from here?. Genome Res. 2009;19(5):711–22. https://doi.org/10.1101/gr.086652.108 PMID: 19411596

13. Lohmueller KE, Bustamante CD, Clark AG. Detecting directional selection in the presence of recent admixture in African-Americans. Genetics. 2011;187(3):823–35. https://doi.org/10.1534/genetics.110.122739 PMID: 21196524
14. Pybus M, Luisi P, Dall'Olio GM, Uzkudun M, Laayouni H, Bertranpetit J, et al. Hierarchical boosting: a machine-learning framework to detect and classify hard selective sweeps in human populations. Bioinformatics. 2015;31(24):3946–52. https://doi.org/10.1093/bioinformatics/btv493 PMID: 26315912
15. Kern AD, Schrider DR. diploS/HIC: An Updated Approach to Classifying Selective Sweeps. G3 (Bethesda). 2018;8(6):1959–70. https://doi.org/10.1534/g3.118.200262 PMID: 29626082
16. Torada L, Lorenzon L, Beddis A, Isildak U, Pattini L, Mathieson S, et al. ImaGene: a convolutional neural network to quantify natural selection from genomic data. BMC Bioinformatics. 2019;20(Suppl 9):337. https://doi.org/10.1186/s12859-019-2927-x PMID: 31757205
17. Hejase HA, Mo Z, Campagna L, Siepel A. A Deep-Learning Approach for Inference of Selective Sweeps from the Ancestral Recombination Graph. Mol Biol Evol. 2022;39(1):msab332. https://doi.org/10.1093/molbev/msab332 PMID: 34888675
18. Arnab SP, Campelo Dos Santos AL, Fumagalli M, DeGiorgio M. Efficient Detection and Characterization of Targets of Natural Selection Using Transfer Learning. Mol Biol Evol. 2025;42(5):msaf094. https://doi.org/10.1093/molbev/msaf094 PMID: 40341942
19. Arnab SP, Dos Santos ALC, Fumagalli M, DeGiorgio M. Semi-supervised detection of natural selection with positive-unlabeled learning. bioRxiv. 2025.
20. Flagel L, Brandvain Y, Schrider DR. The Unreasonable Effectiveness of Convolutional Neural Networks in Population Genetic Inference. Mol Biol Evol. 2019;36(2):220–38. https://doi.org/10.1093/molbev/msy224 PMID: 30517664
21. Isildak U, Stella A, Fumagalli M. Distinguishing between recent balancing selection and incomplete sweep using deep neural networks. Mol Ecol Resour. 2021;21(8):2706–18. https://doi.org/10.1111/1755-0998.13379 PMID: 33749134
22. Haller BC, Messer PW. SLiM 4: Multispecies Eco-Evolutionary Modeling. The American Naturalist. 2023;201(5):E127-39. https://doi.org/10.1086/723601
23. Kern AD, Schrider DR. Discoal: flexible coalescent simulations with selection. Bioinformatics. 2016;32(24):3839–41. https://doi.org/10.1093/bioinformatics/btw556 PMID: 27559153
24. Ewing G, Hermisson J. MSMS: a coalescent simulation program including recombination, demographic structure and selection at a single locus. Bioinformatics. 2010;26(16):2064–5. https://doi.org/10.1093/bioinformatics/btq322 PMID: 20591904
25. Adrion JR, Galloway JG, Kern AD. Predicting the Landscape of Recombination Using Deep Learning. Mol Biol Evol. 2020;37(6):1790–808. https://doi.org/10.1093/molbev/msaa038 PMID: 32077950
26. Mo Z, Siepel A. Domain-adaptive neural networks improve supervised machine learning based on simulated population genetic data. PLoS Genet. 2023;19(11):e1011032. https://doi.org/10.1371/journal.pgen.1011032 PMID: 37934781
27. Devlin J, Chang MW, Lee K, Google KT, Language AI. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 2019. https://github.com/tensorflow/tensor2tensor
28. Grossman SR, Andersen KG, Shlyakhter I, Tabrizi S, Winnicki S, Yen A, et al. Identifying recent adaptations in large-scale genomic data. Cell. 2013;152(4):703–13. https://doi.org/10.1016/j.cell.2013.01.035 PMID: 23415221
29. Akbari A, Barton AR, Gazal S, Li Z, Kariminejad M, Perry A. Pervasive findings of directional selection realize the promise of ancient DNA to elucidate human adaptation. 2024. https://doi.org/10.1101/2024.09.14.613021
30. Cecil RM, Sugden LA. On convolutional neural networks for selection inference: Revealing the effect of preprocessing on model learning and the capacity to discover novel patterns. PLoS Comput Biol. 2023;19(11):e1010979. https://doi.org/10.1371/journal.pcbi.1010979 PMID: 38011281
31. Ho J, Kalchbrenner N, Weissenborn D, Salimans T. Axial Attention in Multidimensional Transformers. In: 2019. http://arxiv.org/abs/1912.12180
32. Rao R, Liu J, Verkuil R, Meier J, Canny JF, Abbeel P. MSA Transformer. 2021. https://github.com/facebookresearch/
33. 1000 Genomes Project Consortium, Auton A, Brooks LD, Durbin RM, Garrison EP, Kang HM, et al. A global reference for human genetic variation. Nature. 2015;526(7571):68–74. https://doi.org/10.1038/nature15393 PMID: 26432245
34. Rubinacci S, Delaneau O, Marchini J. Genotype imputation using the Positional Burrows Wheeler Transform. PLoS Genet. 2020;16(11):e1009049. https://doi.org/10.1371/journal.pgen.1009049 PMID: 33196638
35. Browning BL, Zhou Y, Browning SR. A One-Penny Imputed Genome from Next-Generation Reference Panels. Am J Hum Genet. 2018;103(3):338–48. https://doi.org/10.1016/j.ajhg.2018.07.015 PMID: 30100085
36. Karczewski KJ, Francioli LC, Tiao G, Cummings BB, Alföldi J, Wang Q, et al. The mutational constraint spectrum quantified from variation in 141,456 humans. Nature. 2020;581(7809):434–43. https://doi.org/10.1038/s41586-020-2308-7 PMID: 32461654
37. Liu X, Koyama S, Tomizuka K, Takata S, Ishikawa Y, Ito S, et al. Decoding triancestral origins, archaic introgression, and natural selection in the Japanese population by whole-genome sequencing. 2024.
38. Anguita-Ruiz A, Aguilera CM, Gil Á. Genetics of lactose intolerance: an updated review and online interactive world maps of phenotype and genotype frequencies. Nutrients. 2020;12(9):2689. https://doi.org/10.3390/nu12092689
39. Dehasque M, Ávila-Arcos MC, Díez-Del-Molino D, Fumagalli M, Guschanski K, Lorenzen ED, et al. Inference of natural selection from ancient DNA. Evol Lett. 2020;4(2):94–108. https://doi.org/10.1002/evl3.165 PMID: 32313686

40. van den Belt S, Alachiotis N. Fast and accurate deep learning scans for signatures of natural selection in genomes using FASTER-NN. Commun Biol. 2025;8(1):58. https://doi.org/10.1038/s42003-025-07480-7 PMID: 39814854
41. Kingma DP, Welling M. An Introduction to Variational Autoencoders. Foundations and Trends® in Machine Learning. 2019;12(4):307–92. https://doi.org/10.1561/2200000056
42. Rethmeier N, Augenstein I. A Primer on Contrastive Pretraining in Language Processing: Methods, Lessons Learned, and Perspectives. ACM Computing Surveys. 2023;55:1–17. https://doi.org/10.1145/3561970
43. Cahyawijaya S, Yu T, Liu Z, Zhou X, Mak TWT, Ip YYN, et al. SNP2Vec: Scalable Self-Supervised Pre-Training for Genome-Wide Association Study. In: Proceedings of the 21st Workshop on Biomedical Language Processing, 2022. 140–54. https://doi.org/10.18653/v1/2022.bionlp-1.14
44. Benegas G, Ye C, Albors C, Li JC, Song YS. Genomic language models: opportunities and challenges. Trends Genet. 2025;41(4):286–302. https://doi.org/10.1016/j.tig.2024.11.013 PMID: 39753409
45. Beltagy I, Peters ME, Cohan A. Longformer: The long-document transformer. 2020. http://arxiv.org/abs/2004.05150
46. Wang S, Li BZ, Khabsa M, Fang H, Ma H. Linformer: Self-Attention with Linear Complexity. In: 2020. http://arxiv.org/abs/2006.04768
47. Blassel L, Boussau B, Lartillot N, Jacob L. Likelihood-free inference of phylogenetic tree posterior distributions. In: 2025. https://doi.org/arXiv:251012976
48. Min J, Ning Y, Pope NS, Baumdicker F, Kern AD. Neural posterior estimation for population genetics. bioRxiv. 2025. 2025–12.
49. He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016. 770–8. https://doi.org/10.1109/cvpr.2016.90
50. Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, et al. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. 2020. http://jmlr.org/papers/v21/20-074.html
51. He K, Chen X, Xie S, Li Y, Dollár P, Girshick R. Masked Autoencoders Are Scalable Vision Learners. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022. 15979–88. https://doi.org/10.1109/CVPR52688.2022.01553
52. Adrion JR, Cole CB, Dukler N, Galloway JG, Gladstein AL, Gower G, et al. A community-maintained standard library of population genetic models. Elife. 2020;9:e54967. https://doi.org/10.7554/eLife.54967 PMID: 32573438
53. Gower G, Pope NS, Rodrigues MF, Tittes S, Tran LN, Alam O, et al. Accessible, Realistic Genome Simulation with Selection Using stdpopsim. Mol Biol Evol. 2025;42(11):msaf236. https://doi.org/10.1093/molbev/msaf236 PMID: 40994024
54. Wang Z, Wang J, Kourakos M, Hoang N, Lee HH, Mathieson I, et al. Automatic inference of demographic parameters using generative adversarial networks. Mol Ecol Resour. 2021;21(8):2689–705. https://doi.org/10.1111/1755-0998.13386 PMID: 33745225
55. International HapMap Consortium, Frazer KA, Ballinger DG, Cox DR, Hinds DA, Stuve LL, et al. A second generation human haplotype map of over 3.1 million SNPs. Nature. 2007;449(7164):851–61. https://doi.org/10.1038/nature06258 PMID: 17943122
56. Hermisson J, Pennings PS. Soft sweeps. Genetics. 2005;169(4):2335–52. https://doi.org/10.1534/genetics.104.036947
57. Whitehouse LS, Ray D, Schrider DR. Tree sequences as a general-purpose tool for population genetic inference. 2024. https://doi.org/10.1101/2024.02.20.581288