PNAS NEXUS

PNAS Nexus, 2026, **5**, pgag046

https://doi.org/10.1093/pnasnexus/pgag046
Advance access publication 5 March 2026
Research Report

Inferring genotype–phenotype maps using attention models

Krishna Rijal [a], Caroline M. Holmes [b], Samantha Petti [c], Gautam Reddy [d], Michael M. Desai [b] and Pankaj Mehta [a,*]

[a]Department of Physics, Boston University, Boston, MA 02215, USA
[b]Department of Organismic and Evolutionary Biology, Harvard University, Cambridge, MA 02138, USA
[c]Department of Mathematics, Tufts University, Medford, MA 02155, USA
[d]Department of Physics, Princeton University, Princeton, NJ 08540, USA
*To whom correspondence should be addressed: Email: pankajm@bu.edu
Edited By Panayiotis Benos

Abstract

Predicting phenotype from genotype is a central challenge in genetics. Traditional approaches in quantitative genetics typically analyze this problem using methods based on linear regression. These methods generally assume that the genetic architecture of complex traits can be parameterized in terms of an additive model, where the effects of loci are independent, plus (in some cases) pairwise epistatic interactions between loci. However, these models struggle to analyze more complex patterns of epistasis or subtle gene–environment interactions. Recent advances in machine learning, particularly attention-based models, offer a promising alternative. Initially developed for natural language processing, attention-based models excel at capturing context-dependent interactions and have shown exceptional performance in predicting protein structure and function. Here, we apply attention-based models to quantitative genetics. We analyze the performance of this attention-based approach in predicting phenotype from genotype using simulated data across a range of models with increasing epistatic complexity, and using experimental data from a recent quantitative trait locus mapping study in budding yeast. We find that our model demonstrates superior out-of-sample predictions in epistatic regimes compared to standard methods. We also explore a more general multienvironment attention-based model to jointly analyze genotype–phenotype maps across multiple environments and show that such architectures can be used for "transfer learning"—predicting phenotypes in novel environments with limited training data.

Significance Statement

Predicting phenotype from genotype is a central challenge in genetics. Here, we introduce a new class of statistical models for carrying out this task (specifically quantitative trait locus (QTL) mapping in model organisms) inspired by machine learning techniques for natural language processing (attention-based models). Our models are especially well-suited for capturing gene–gene interactions (epistasis) and the central role played by the environment in shaping genotype–phenotype maps. We demonstrate the power of this approach using synthetic datasets and experimental data from QTL mapping experiments in budding yeast.

Introduction

Mapping the genetic basis of complex traits is a central goal in quantitative genetics (1–6). This challenge is typically tackled by measuring genotypes and phenotypes of a large number of diverse individuals, and then identifying statistical associations between genetic variants and corresponding phenotypes. Numerous methods have been developed to infer these genotype–phenotype (G–P) maps using data from genome-wide association studies (GWAS) in humans or from controlled crosses in model organisms (eg quantitative trait locus [QTL] mapping) (7–11). These existing methods are typically based on linear regression. Each genotype is represented as a vector of allelic states at a series of genetic loci (see Fig. 1a and details in the next section). The phenotype corresponding to each genotype can then be parameterized based on the linear effect of each locus, plus pairwise and higher-order epistatic interactions (see Fig. 1b). GWAS or QTL mapping studies fit the parameters of this model by minimizing prediction error across individuals (12, 13). The resulting model can then be used to quantify how much of the observed variability in phenotypic traits can be attributed to genetic factors, to identify which genetic loci are most strongly associated with the phenotypes, and to predict the phenotypes of new individuals from their genotype.

Because the number of individuals in even the largest studies is much smaller than the number of possible genotypes, some form of regularization is typically applied to avoid overfitting. Traditional analyses begin with an additive model, where the expected phenotype is determined by the sum of independent

Competing Interest: The authors declare no competing interests.
Received: November 20, 2025. **Accepted:** February 11, 2026

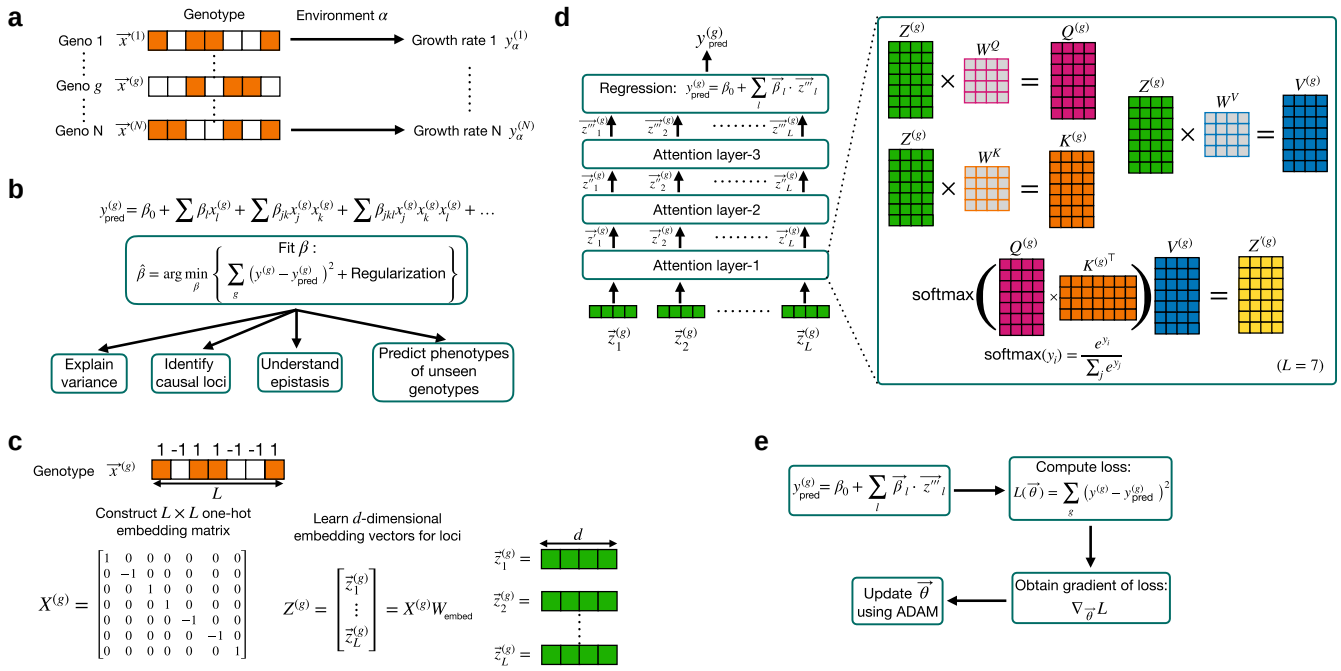

Fig. 1. Schematic illustrating standard regression-based and attention-based methods for G–P mapping. a) Genotype sequences represented as vectors $\mathbf{x}^{(g)}$ for the gth individual, with phenotype $y_\alpha^{(g)}$ in environment α. b) Classical series expansion model, combining linear and higher-order epistasis, fitted by minimizing the loss function with regularization. c) Genotype vectors are converted to one-hot embeddings $X^{(g)}$ and transformed into d-dimensional embeddings $Z^{(g)}$. d) Flowchart illustrating attention-based architecture for a case of $L = 7$ loci and $d = 4$. The embeddings pass through multiple attention layers, followed by prediction. e) The optimization process involves computing the loss $L(\boldsymbol{\theta})$, obtaining the gradient of the loss $\nabla_{\boldsymbol{\theta}} L$, and updating the parameters $\boldsymbol{\theta}$.

effects at some set of causal loci. Effects of potential pairwise epistatic interactions between these loci can then sometimes also be mapped. However, existing methods lack power to infer more complex patterns of epistatic interactions across multiple loci. The potential impact of this epistasis remains controversial. Variance partitioning methods (14, 15) often show that epistasis explains only a small fraction of observed variance in phenotypes, but this may be in part because the effects of epistasis are treated as a perturbation to an initial fit based on additive effects (16). In addition, because biological systems are composed of interconnected and nonlinear networks (17), even sparse epistatic effects can provide key insight into the molecular basis of phenotypes.

Gene–environment interactions are also challenging to model within the context of traditional methods in quantitative genetics (18, 19). These occur when the effect of a genetic variant on a phenotype depends on the environmental context. For instance, individuals with a genetic predisposition to obesity might only exhibit this trait if they are exposed to high-calorie diets (20). Incorporating these interactions in GWAS is challenging because it requires comprehensive environmental data alongside genetic data, and the potential environmental variables and their interactions with genetic factors can be vast and complex. Typically, a linear mixed-model approach is used to study gene–environment interactions by including interaction terms in analyses, though the true interactions may not always be linear (21, 22).

Despite dramatic increases in the scale of GWAS and QTL mapping studies, traditional methods often fail to capture intricate patterns and interactions, prompting the exploration of machine learning (ML) approaches (23). These include convolutional neural networks (24, 25), graph neural networks (26), evolution-informed pipelines (27), and more specialized interpretable deep learning architectures (28). Other approaches seek to utilize deep learning-based computer vision models to predict phenotypes using MRI images and facial recognition technology (29, 30)

Recent developments in attention-based models present a promising alternative ML paradigm for learning G–P maps (31, 32). Originally designed for natural language processing, attention-based models excel at capturing context-dependent interactions, a capability particularly relevant for genetic data, where the effect of a locus can be highly dependent on broader genetic and environmental contexts. Attention-based models have been successfully applied beyond natural language processing, notably to predict protein structure and function with remarkable accuracy (33, 34), and to predict binding of T cell receptors to epitopes (35).

In this study, we apply attention-based models to predict phenotype from genotype. These attention-based models offer several potential advantages over traditional methods. They are highly expressive because they use learned vectors that capture the entire context relevant to the effects of a given locus on phenotype, rather than relying on context-independent scalar effects. This capability enables them to capture subtle genetic interactions. Another key strength is the ability to incorporate environmental tokens naturally within the structure of the model, making it possible to account for potential gene–environment interactions and hence adapt predictions based on the environment. Traditional methods often lack this flexibility. Additionally, attention-based models can learn environment-dependent vectors, and hence interpolate to make predictions for environments not present in the training data.

The architecture of the attention-based models begins by converting genotypes into vectors, which are then processed through attention layers to capture epistatic and environmental interactions. This approach enables the model to capture effects of

complex epistasis, adapt to environmental contexts, and predict unseen genotypes by learning a generalizable mapping from genotype to phenotype. As a proof of principle, we first apply these methods to analyze simulated data in models with varying degrees of epistatic complexity, and then apply our approach to data from a recent QTL mapping study in budding yeast.

In the next section, we provide a detailed explanation of the attention-based architecture used in this study, highlighting its components and the underlying mechanisms that make it well-suited for G–P mapping. In subsequent sections, we then apply this architecture to analyze simulated data in models of varying epistatic complexity and experimental data from a recent QTL mapping study in budding yeast that quantitatively measured the growth rate of 100,000 individuals across 18 different environments (10). This unique study is ideally suited for exploring the efficacy of attention-based architectures which are thought to function best with large amounts of data. Finally, we present a multienvironment generalization of our attention-based approach, and analyze its ability to "transfer knowledge" to new environments.

Attention-based architecture for G–P mapping

Our ultimate goal is to develop attention-based models that can predict phenotypes based on genotype data, while also accounting for epistasis and gene–environment interactions. Before diving into the details of the attention-based architecture, it is essential to understand the standard regression model commonly used in the field. This model serves as a foundation for understanding G–P mapping.

Overview of standard model

To make the discussion concrete, we focus on the yeast QTL experiments in (10) where the authors constructed a panel of 100,000 F1 haploid offspring (segregants) from a cross between the laboratory strain BY and the vineyard strain RM. Each segregant was sequenced to determine its genotype, and the relative growth rate of all segregants was measured in each of 18 laboratory media conditions. For our purposes here, this growth rate data represents 18 different phenotypes.

In this dataset, the two parental strains differ at approximately $L = 41{,}594$ loci (corresponding primarily to single-nucleotide polymorphisms) and exhibit variation in many relevant phenotypes. For any given offspring g, we denote the two possible genotypes (corresponding to the BY or RM parent, respectively) at locus l by $x_l^{(g)} = \pm 1$. In other words, in this biallelic case the genotype of the gth individual can be represented by an L-dimensional vector $\mathbf{x}^{(g)}$, where the lth element in this vector is either $+1$ or -1, depending on the allele present at the lth locus. Each of these genotypes corresponds to a set of 18 measured phenotypes, $y_\alpha^{(g)}$, the growth rate in media condition α.

We then model each predicted phenotype for the gth individual, $y_{\text{pred}}^{(g)}$, as:

$$y_{\text{pred}}^{(g)} = \beta_0 + \sum_j \beta_j x_j^{(g)} + \sum_{j<k} \beta_{jk} x_j^{(g)} x_k^{(g)} + \sum_{j<k<l} \beta_{jkl} x_j^{(g)} x_k^{(g)} x_l^{(g)} + \cdots. \quad (1)$$

Here, the β coefficients capture the effect sizes of these loci and their epistatic interactions. Each of the phenotypes is modeled independently. The goal of the model is to estimate the coefficients β for each phenotype by minimizing the sum of squared differences between the measurement $y^{(g)}$ and the prediction $y_{\text{pred}}^{(g)}$. This is mathematically expressed as:

$$\hat{\beta} = \arg\min_\beta \left\{ \sum_g \left(y^{(g)} - y_{\text{pred}}^{(g)} \right)^2 + \text{Regularization} \right\}, \quad (2)$$

where the regularization term helps prevent overfitting by penalizing large coefficient values, ensuring that the model remains generalizable.

The purely linear version of this model (with only β_0 and β_j terms nonzero) assumes that the effects of different loci are additive. More generally, the model can be used to fit both linear and epistatic terms, but data limitations often make it difficult to infer even pairwise epistatic terms. Higher-order terms are in practice almost always impossible to estimate. These limitations lead to incomplete capture of the complexity of the genetic architecture. While the model uses regularization to prevent overfitting, this does not circumvent the fundamental issue of using limited data to explore a high-dimensional genomic space. In addition, standard models struggle to incorporate the impact of environmental factors (eg in the case of our yeast data, relationships between the growth rates in the different media conditions), which can significantly influence phenotypic traits.

Attention-based architecture

To address the limitations of traditional models, we propose an attention-based architecture. We first describe a single-environment version of this architecture that learns the G–P map for each environment separately. Later, we will discuss the multienvironment architecture, which can learn the G–P map for all environments jointly, capturing gene–environment interactions and cross-environment information.

In the subsequent subsections, we describe the key specific components of the single-environment attention-based model and explain how each part contributes to its capacity to handle complex epistasis. We outline the process from genotype representation to the final prediction of phenotypes and optimization, emphasizing throughout the distinctive features of attention-based architecture (see Fig. 1c–e).

Genotype representation and embedding

The first step in our statistical procedure is the transformation of genotype data into a format that can be processed by the attention-based architecture (see Fig. 1c). The L-dimensional genotype vector $\mathbf{x}^{(g)}$ is encoded as L one-hot vectors with a single nonzero element, which is ± 1. In this encoding, the lth element of the lth one-hot vector is equal to $x_l^{(g)}$, while the remaining elements are zeros. The one-hot embedding matrix $X^{(g)}$, where each row corresponds to the one-hot vector for each locus, can be written as:

$$X^{(g)} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & -1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}. \quad (3)$$

This matrix is then transformed into a continuous, dense representation using a learned $L \times d$-dimensional embedding matrix W_{embed}, resulting in an $L \times d$-dimensional matrix $Z^{(g)}$ for each individual:

$$Z^{(g)} = X^{(g)} W_{\text{embed}}. \quad (4)$$

The lth row in $Z^{(g)}$ corresponds to a d-dimensional embedding vector $\mathbf{z}_l^{(g)}$ for the lth locus, capturing its genetic information in a form suitable for subsequent processing by the attention-based model. This transformation is useful because the original genotype matrix $X^{(g)}$ is high-dimensional and sparse, making direct use

computationally expensive. The embedding matrix reduces dimensionality and captures essential genetic information, leading to more efficient computations.

The learned embedding vectors $\mathbf{z}^{(g)}$ represent complex relationships between genetic loci that might not be apparent in the raw one-hot encoded representation. Learned dense embeddings are more suitable for neural network-based models, like transformers, providing a richer representation of input data (36).

The embedding dimension d controls the trade-off between capturing complex genetic interactions and minimizing computational cost. We select d by evaluating validation performance across different values, $d = 2, 12, 30, 50$. The attention-based architectures showed good performance for all choices of d (see Fig. 2c). However, for simulated data where phenotypes are explicitly generated with high-order epistasis and a structured G–P relationship, we found $d = 30$ was optimal especially for highly epistatic models. However, for experimental data, where the true G–P map is unknown, our simulations suggested $d = 12$ was sufficiently large to capture complex phenotypes while minimizing computational costs.

Attention mechanism

The core of the attention-based architecture is the attention mechanism, which enables the model to weigh the importance of different loci. In the attention layer, the embeddings are transformed into three sets of vectors: queries $Q^{(g)}$, keys $K^{(g)}$, and values $V^{(g)}$ (see Fig. 1d). These transformations are achieved through learned weight matrices W^Q, W^K, and W^V:

$$Q^{(g)} = Z^{(g)}W^Q, \quad K^{(g)} = Z^{(g)}W^K, \quad V^{(g)} = Z^{(g)}W^V. \tag{5}$$

The necessity of these three different vectors arises from the design of the attention mechanism, which aims to flexibly capture complex interactions between loci while maintaining computational tractability. The queries $Q^{(g)}$ represent the perspective of the current locus, seeking information about other loci. The keys $K^{(g)}$ represent the potential loci that the current locus may attend to, effectively serving as points of reference. The values $V^{(g)}$ contain the actual information that should be integrated, based on the relevance determined by queries and keys.

The attention mechanism calculates the relevance between different loci by computing the attention scores, which are determined as the dot product of queries and keys. The output of an attention layer is given by:

$$Z'^{(g)} = \text{softmax}\left(Q^{(g)}K^{(g)\top}\right)V^{(g)}, \tag{6}$$

where the softmax function is defined as:

$$\text{softmax}(y_i) = \frac{\exp(y_i)}{\sum_j \exp(y_j)}. \tag{7}$$

The softmax function normalizes the attention scores into a probability distribution, emphasizing the most relevant loci while suppressing less relevant ones. The output of the attention layer, therefore, is a weighted sum of the value vectors $V^{(g)}$, where the weights are the normalized attention scores.

This process relates closely to the concept of epistasis, where the effect of one gene is influenced by others. In the attention mechanism, the query $Q^{(g)}$ from a given locus interacts with the keys $K^{(g)}$ of other loci to determine relevance. The softmax function then assigns higher weights to locus pairs with stronger interactions. This adjustment allows the model to capture the epistasis by emphasizing loci combinations that are contextually significant, thus highlighting how different genes influence each other within the genetic network. This capability to focus on important genetic relationships makes the attention-based model particularly effective for modeling epistasis.

Stacking multiple attention layers improves the model's ability to capture higher-order epistatic interactions by iteratively refining how loci influence each other. The first layer learns direct pairwise interactions, while deeper layers integrate signals across three or more loci, capturing complex, nonlinear dependencies. Such stacking of attention layers is commonly used in large language models, including those used to model proteins. We use three layers because they collectively capture both pairwise and higher-order interactions, and empirical tests showed that adding more layers did not improve performance.

Prediction

After the embeddings have been processed through the attention layers, the final representation $Z'''^{(g)}$ (obtained from the third attention layer) is used for prediction (see Fig. 1d). The predicted phenotype $y^{(g)}_{\text{pred}}$ for each individual g is given by taking these vectors and processing them through a "regression layer":

$$y^{(g)}_{\text{pred}} = \beta_0 + \sum_l \boldsymbol{\beta}_l \cdot \mathbf{z}'''^{(g)}_l. \tag{8}$$

Here, $\boldsymbol{\beta}_l$ represents the weights associated with each embedding, and β_0 is the bias term. This regression layer is necessitated by the fact that the output of our model is a continuous real-valued number: the predicted growth rate of an individual in a given environment.

Optimization

The objective is to minimize the prediction error, which is typically quantified by a loss function. A loss function measures the difference between the predicted and actual values of the quantities of interest. We employ mean squared error as our loss function $L(\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ represents all the parameters involved in the architecture. It is defined as:

$$L(\boldsymbol{\theta}) = \sum_g \left(y^{(g)} - y^{(g)}_{\text{pred}}\right)^2. \tag{9}$$

We minimize this function because, theoretically, at its minimum, $y^{(g)}$ equals $y^{(g)}_{\text{pred}}$.

The entire model is trained end-to-end using gradient-based optimization methods (see Fig. 1e). The process begins with an initial guess for the model parameters $\boldsymbol{\theta}_0$. The input embedding vectors are passed through the architecture, and the gradient of the loss function $\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta})$ is calculated. The gradients indicate the direction and magnitude by which each parameter should be adjusted to reduce the loss. The parameters are then updated by moving in the direction of the gradient using gradient descent or more sophisticated second-order methods such as ADAM (37, 38), until a local minimum of the loss function is reached.

In this work, the PyTorch library is utilized for its efficient automatic differentiation capabilities (39). PyTorch computes the gradients of the loss function using backpropagation, which applies the chain rule of calculus to propagate the error gradients backward through the network from the output layer to the input layer (38).

Implementation

A detailed explanation of how the attention-based architecture is implemented using PyTorch is given in the SI Appendix, Section 3.

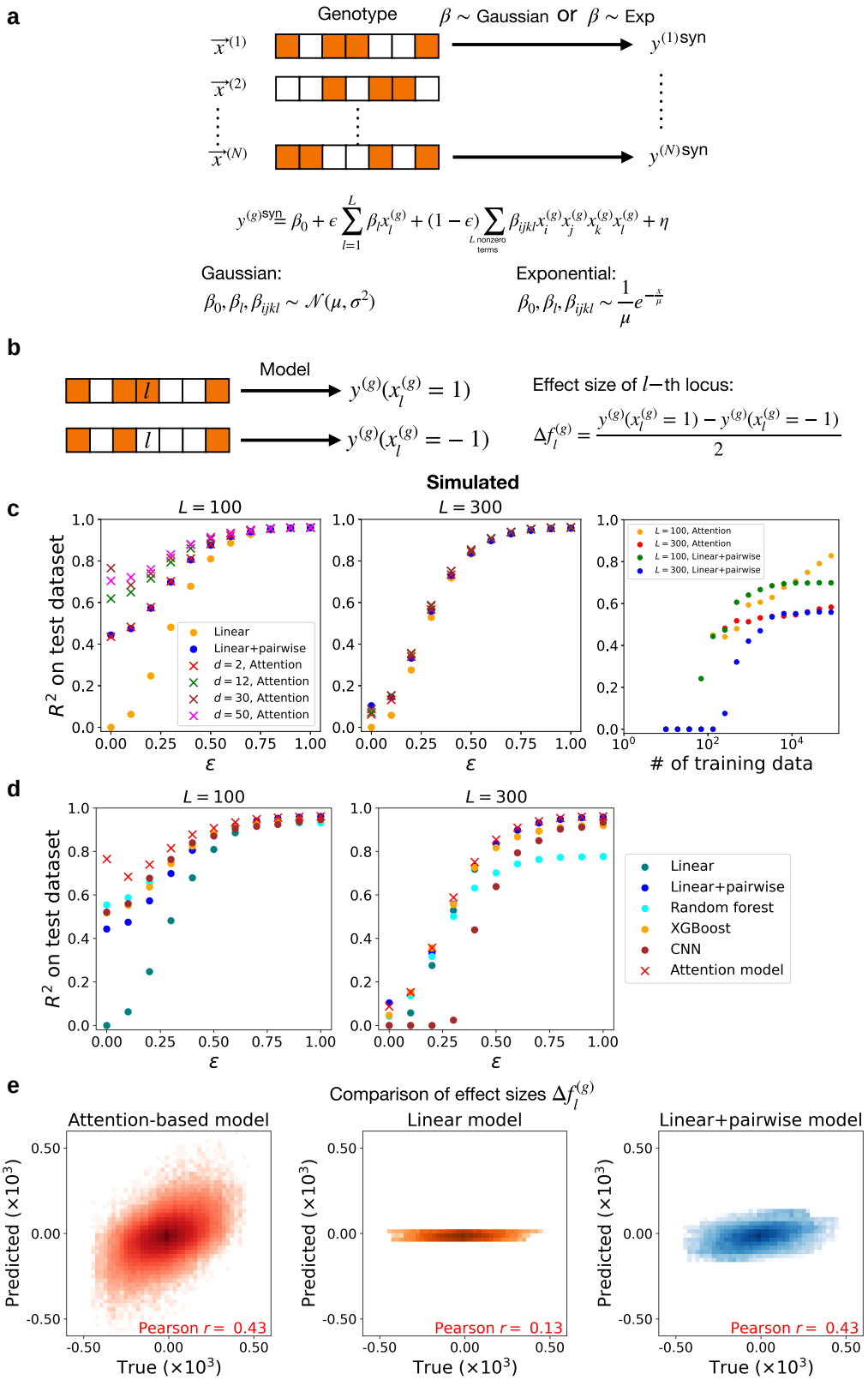

Fig. 2. Model comparison on synthetic data. a) Schematic of pipeline for generating simulated data. We create subsampled genotype vectors $\mathbf{x}^{(g)}$ for individual g from the real genotype data in (10) and simulate the phenotype of this individual according to the equation shown, with coefficients drawn from Gaussian (main text) or Exponential (SI Appendix) distributions. b) We define the model prediction for the effect size of the lth locus in the gth background genotype as $\Delta f_l^{(g)}$, which is the difference between the predicted phenotype with the lth locus being 1 vs. −1 in that genetic background at all other loci. c) R^2 scores for linear, linear+pairwise, and attention-based models across simulated data sets as a function of the simulated strength of epistasis ϵ. Left panel shows the case of $L = 100$ causal loci, while middle panel shows the case of $L = 300$ causal loci. Right panel shows performance at $\epsilon = 0.3$ for varying training dataset sizes. d) Comparison across six methods: test-set R^2 vs. ϵ for linear, linear+pairwise, random forest, XGBoost, a 1D-CNN, and the attention model, shown for $L = 100$ (left) and $L = 300$ (right). e) For $d = 30$, $\epsilon = 0.3$, and $L = 100$, the predicted effect sizes for each locus from different models are compared with the true effect sizes. Simulated data are generated using Gaussian-distributed coefficients.

Parameter complexity comparison

The attention-based model contains significantly fewer parameters compared to the standard linear plus pairwise model whenever the number of loci is much smaller than the embedding dimension ($L \ll d$). The total number of parameters in the attention-based model is given by $2Ld + 3d^2N_L + 1$, where N_L is the number of attention layers. In contrast, the standard linear +pairwise model, which accounts for additive effects and pairwise interactions, has $\frac{L(L+1)}{2} + 1$ parameters.

For example, when $L = 1{,}000$, $d = 12$, and $N_L = 3$, the attention-based model has 25,297 parameters, whereas the linear+pairwise model has 500,501 parameters. This means the linear+pairwise model contains roughly 20 times more parameters than the attention-based model. Despite this lower parameter count, we show below that the attention-based model architecture nevertheless efficiently captures higher-order interactions.

Benchmarking on simulated G–P maps

To evaluate the ability of our attention-based model to capture complex epistasis, we benchmarked it against traditional methods using synthetic data from a set of simulated G–P maps. This approach allows us to systematically test model performance in controlled scenarios with varying level of epistasis.

As described above, a key challenge in traditional methods is limited capacity to account for complex epistasis. Therefore, to probe whether our attention-based model can more accurately capture these complex interactions, we generated a set of simulated genetic architectures where phenotype is determined by a combination of additive (linear) terms along with fourth-order epistatic interactions (see Eq. 10 below). We tune the relative contributions of linear and fourth-order interactions across different simulated architectures, and analyze the performance of traditional vs. attention-based models as a function of these relative contributions of simple vs. complex epistasis.

Generating simulated data

A key challenge in inferring G–P maps is that the genotype structure of sampled individuals is typically shaped by numerous historical, demographic, and evolutionary processes that are unrelated to the phenotypes we aim to predict. We therefore cannot reliably compare methods by assessing their performance on simulated data with random genotypes. For this reason, to maintain realistic genotype structure, we constructed all our synthetic datasets using the actual observed yeast genomes of the ∼100,000 individuals from the QTL experiments in (10). This preserves the linkage structure inherent in the cross used to generate these individuals.

As discussed in the original paper, these individuals differ at 41,594 loci across the genome, but because of the tight linkage in these F1 segregants, genotypes at nearby loci are highly correlated. This creates a "fine-mapping" problem that is extensively discussed in (10). However, our goal here is to focus on inference of complex epistasic and environmental interactions, rather than on fine mapping (where we would not expect attention models to have any natural advantage over other methods). With this in mind, we subsample the loci (effectively combining highly correlated loci) to create a representative set of $L = 1{,}164$ independent loci (see section on applications to data below). Reference (10) found that, using traditional methods, several hundred loci could be identified as causal. For this reason, in our synthetic datasets, we typically choose 100 to 300 random loci we designate to be causal.

To assign a synthetic phenotype to each of these genotypes, we use the approach shown in Fig. 2a. As before, the genotype vector $\mathbf{x}^{(g)}$ of the gth individual, with L loci, is an L-dimensional vector. In this vector, the lth element is either +1 or −1, depending on the allele present at the lth locus (in the real genotypes, subsampled as described above). We then assign the phenotype $y^{(g)\mathrm{syn}}$ to each genotype:

$$y^{(g)\mathrm{syn}} = \beta_0 + \epsilon \sum_{l=1}^{L} \beta_l x_l^{(g)} + (1-\epsilon) \sum_{L \text{ nonzeroterms}} \beta_{ijkl} x_i^{(g)} x_j^{(g)} x_k^{(g)} x_l^{(g)} + \eta, \tag{10}$$

where the sums are over the randomly designated causal loci (see above). By tuning the parameter ϵ, we can generate synthetic data with varying amounts of epistasis, ranging from purely additive ($\epsilon = 1$) to purely higher-order epistatic interactions ($\epsilon = 0$). In the main text, we focus on the case where β terms are drawn from a Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$; in Fig. S2, we show analogous results for the case when these coefficients are drawn from an exponential distribution. The noise term η represents random environmental or measurement noise and is modeled as a Gaussian distribution with a mean of zero and a SD equal to 20% of the SD of the simulated fitness.

Comparison of out-of-sample predictive power

Figure 2c shows a comparison of our attention-based model with classical linear regression-based approaches (see Eq. 1) on synthetic data generated as described above. For the linear and linear+pairwise regression models, we applied L1 regularization using Python's built-in *ElasticNet* implementation, sweeping over the regularization strength to optimize performance. All comparisons are based on R^2 in a held-out test dataset (see SI Appendix, Section 3 for details of fitting procedures). Throughout this study, we use out-of-sample predictions to compare model performance. Unlike in-sample predictions, which are made on the data used to train the model, out-of-sample predictions are made on new, unseen data. This allows us to assess how well the model can generalize, a key indicator of its practical utility.

As can be seen in the left most panel of Fig. 2c, for $L = 100$ loci, the purely linear model performs well in highly linear settings but degrades as the amount of epistasis increases (ie as ϵ decreases). The linear + pairwise model captures some effects of epistasis, but struggles with highly epistatic genotype-to-phenotype maps (ie $\epsilon \ll 1$). The attention-based model consistently outperforms other methods, particularly in epistatic regimes, demonstrating its ability to capture high-order epistasis. We find similar results for synthetic data generated with exponentially distributed β (see Fig. S2). However, we note that when we increase the number of causal loci used in the synthetic data to $L = 300$, the attention-based and linear regression models have similar predictive power. As discussed below, this likely reflects the fact that as the number of causal loci increases, more data is required to take advantage of the increased expressivity of attention-based models vis-à-vis regression-based approaches.

Comparison with alternative ML architectures

While standard GWAS approaches rely on linear and linear+pairwise models, we also benchmarked our attention-based framework against three widely used ML approaches: a one-dimensional convolutional neural network (1D-CNN), gradient-boosted trees (XGBoost), and random forests. For the 1D-CNN, we used a kernel size of 2, and for both XGBoost and random forests, we used 100

trees. As shown in Fig. 2d, these three models outperform the linear +pairwise baseline in the high-epistasis regime ($\epsilon < 0.5$) but still fall short of the attention model, occupying an intermediate performance tier. Because CNNs and tree-based models can only combine features over a limited window or depth (eg a fixed kernel size or maximum tree depth), they capture some nonlinear effects but are unable to model the arbitrary, long-range, high-order interactions that attention layers are designed to capture. In Fig. S3, we show the comparison between these methods when coefficients are drawn from an exponential distribution.

Dependence of performance on embedding dimension and amount of data

The performance of the attention-based model depends on the dimension d of the embedding and the amount of available training data (see Fig. 2c). The optimal dimension is influenced by both the complexity of the data and the number of causal loci L. A higher number of loci or more complex data necessitates a larger embedding dimension to adequately capture the variability and epistasis present in the dataset. However, setting d too high can make it difficult to train the model. Therefore, it is crucial to optimize the embedding dimension using a validation set. Through experimentation, we found that for simulated data with $L = 100$ or $L = 300$, an embedding dimension of $d = 30$ works well. Therefore, we use $d = 30$ for all simulations with synthetic datasets unless explicitly noted otherwise.

As shown in the right panel of Fig. 2c, the performance of the attention-based model improves consistently as the number of data points increases. In contrast, the performance of the linear +pairwise model tends to plateau when the dataset becomes large, indicating its limited capacity to capture more complex interactions. Notably, although the attention-based model has fewer parameters and is roughly as data-hungry as the linear +pairwise model, it achieves equivalent or even greater predictive performance.

Comparison of learned effect sizes

Next, we investigated whether the attention-based model and other models are capable of learning the effect sizes of causal loci. To do so, we made use of our synthetic data, where we know the true underlying G–P maps. This allows us to compare model predictions to exact effect sizes. In the presence of epistasis, the effect size of a locus depends on the genetic background of an individual. For this reason, we focused on the effect size $\Delta f_l^{(g)}$ of changing locus l in an individual with genetic background g (ie the difference in phenotype when $x_l^{(g)}$ is set to +1 and −1 in background g):

$$\Delta f_l = \frac{1}{N}\sum_{g=1}^{N} \frac{y^{(g)}(x_l^{(g)} = 1) - y^{(g)}(x_l^{(g)} = -1)}{2}. \quad (11)$$

We focused on the case where the G–P maps have both a linear component and moderate amounts of epistasis ($\epsilon = 0.3$). As seen in Fig. 2e, the attention-based architectures are particularly effective at capturing context-dependent genetic interactions in the test dataset. The linear+pairwise model also showed good performance at this task, as measured by Pearson correlation. However, as can be seen in the plots, the predicted effect sizes are systematically lower than the true effect sizes. Unsurprisingly, the linear model is unable to capture the background dependent effects, since the predictions do not depend on the genomic backgrounds.

An alternative quantity of interest is "average" effect size of a locus across all genomic backgrounds in the dataset. This quantity is a natural proxy for the importance of a locus when epistatic interactions are limited. Figure S4 shows a plot of the genome-averaged effect size of loci for attention-based models, compared to a purely linear model and a linear+pairwise model. All models are extremely good at predicting average effect sizes as measured by Pearson correlation (attention: $r = 0.82$, linear: $r = 0.83$, and linear+pairwise: $r = 0.87$), with the attention-based models having slightly smaller correlation coefficient. However, as can be seen in the Fig. 2e, right panel, unlike attention-based architectures the linear+pairwise model consistently underestimates the effect size of loci with large effects.

Application to inference of empirical G–P maps in budding yeast

Having benchmarked attention-based models on synthetic data, we next applied these models to the yeast QTL experiments in Ref. (10). As described above, this earlier work created a panel of ~100,000 haploid F1 offspring (segregants) from a cross between a laboratory strain, BY, and a vineyard strain RM. The authors then measured the relative growth rates of these segregants in 18 different laboratory media conditions, including defined minimal media, rich media, several carbon sources, and a range of chemical and temperature stressors (see Ref. (10) for details on the experimental setup and data acquisition process).

To analyze these data, we divided the 100,000 segregants into training, validation, and test datasets, with 85% of the data set aside for training-validation and the remaining 15% used as the test dataset. The training-validation data were further split into 85% for training and 15% for validation. Using the training dataset, we identified $L = 1{,}164$ independent loci, defined as a set of loci such that the correlation between the SNPs present at any pair of loci is <94% (see SI Appendix, Section 1 for details).

Attention-based model application

We began by training a separate attention-based model for each of the 18 environments in the dataset (see Fig. 1d). Based on hyperparameter sweeps on the validation set, we chose an embedding dimension of $d = 12$ for all models used to analyze experimental data. As expected, the performance of the attention-based model, as characterized by R^2 on the test dataset, is much better than that of the linear model (see Fig. 3). Note that the linear regression is performed with both L1 and L2 regularization, implemented via *ElasticNet*. This shows that the attention-based architecture is able to successfully capture the epistatic interactions in the yeast QTL experiments across a wide variety of environments.

Given the increased predictive power of attention-based architectures compared to linear models, we sought to better understand the relationship between the predictions of the linear model and attention-based models. Figure S5 shows a plot comparing the predicted growth rates of the attention-based models and linear models in all 18 environments for the test dataset. There is strong agreement, with a Pearson correlation coefficients ranging between $r = 0.9$–0.95. Figure S6 shows a comparison of the background-averaged effect size of different loci from the attention-based models and linear regression. Once again there is remarkable agreement (Pearson r = 0.83–0.95). This shows that attention-based architectures can learn all the information contained in linear models while also learning subtle epistatic interactions.

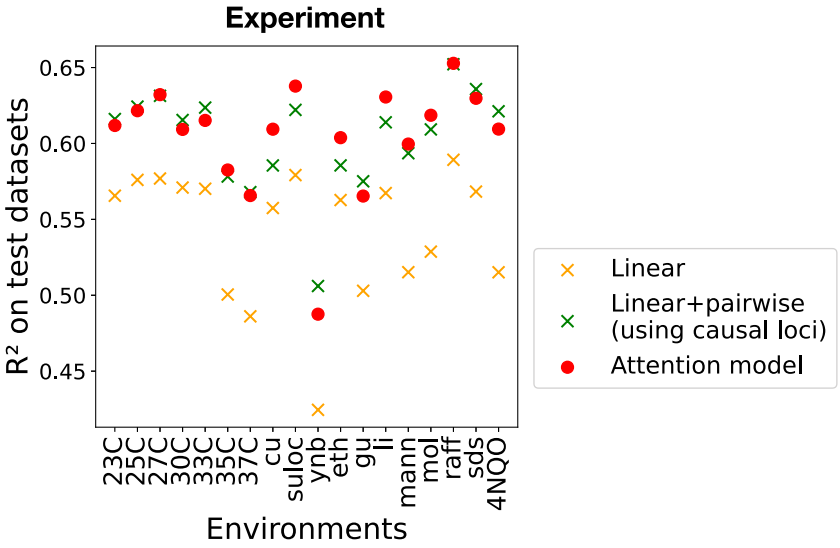

Fig. 3. Comparison of model performance in yeast QTL mapping data. We show R^2 on test datasets for linear, linear+pairwise, and attention-based model (with $d = 12$) across 18 phenotypes (relative growth rates in various environments). For the linear+pairwise mode, the causal loci inferred by (10) are used.

To better understand the kind of epistatic information being captured by attention-based architectures, we also compared our models to a linear plus pairwise model with both L1 and L2 regularization. One major drawback of the linear+pairwise model is that it is computationally infeasible to fit a linear+pairwise model using all $L = 1{,}164$ loci (see discussion of the number of parameters above). For this reason, we restricted our analysis of the linear+pairwise model to the causal loci identified in Ref. (10). Focusing on causal loci reduced the number of loci L in our statistical models by an order of magnitude. We found that the linear +pairwise model restricted to causal loci gave similar performance to the attention-based architectures.

Collectively, these results suggest that attention-based architectures can capture subtle epistatic interactions. Moreover, even without any pretraining, the models can automatically focus on the subset of causal loci responsible for these interactions. Our simulations suggest that the performance of the attention-based architectures are limited by the amount of training data. Therefore, it would not be surprising if given more data (or for G–P maps with more epistasis), attention-based models outperform linear+pairwise models.

In Fig. S7, we present results on alternative attention based architectures where the genotype loci are represented using one-hot encoding and fed directly into the first attention layer, bypassing any dimensionality reduction. This approach allows the model to process the full-resolution genotype information without compressing it into a lower-dimensional embedding space. In this setup, the dimension of all embeddings is equal to $d = L + 1$, where L is the number of loci. We find that for this data regime, these models without dimensionality reduction perform slightly worse than those discussed in the main text.

Multienvironment attention-based architecture

Thus far, we have analyzed each G–P map independently. However, phenotypes are often correlated across similar environmental conditions. For example, the growth rate of yeast shows a stereotyped temperature dependence. Standard methods based on linear regression have historically often neglected any information in these correlations, and instead simply infer the genetic basis of each different phenotype independently. More recently, a wide variety of methods have been developed within this general regression-based framework to integrate multiple different phenotypes (eg multiple types of xQTL data) into a single joint inference framework.

Attention-based architectures offer a potentially powerful alternative approach to capturing these effects. In principle, given enough data, a multienvironment attention-based model (ie a *single* attention model trained on data from *multiple environments*) should be able share information across environments by incorporating different environmental contexts as additional environmental input vectors (see Fig. 4a). This opens the possibility of using attention-based mechanisms to transfer information across environments (often called "transfer learning" in the ML literature (40)). In this section, we explore these ideas using both synthetic data and empirical data from the yeast QTL experiments in Ref. (10). A more detailed analysis of the performance of multienvironment attention models, including details on construction of the synthetic dataset, model architectures, and training procedure, is provided in SI Appendix, Section 4.

Multienvironment attention-based architecture

We start by providing a detailed explanation of the multienvironment attention-based architecture used in our studies (see Fig. 4a). In this architecture, we introduce additional input vectors to represent the environments. Each environment, indexed by α (where $\alpha \in \{1, 2, \ldots, E\}$), is represented by a one-hot encoded vector, denoted as $\mathbf{e}_\alpha$. This encoding allows the model to distinguish between various environmental conditions while processing genotype data. The locus embeddings $\mathbf{z}^{(g)}$ have E zeros at the right side, and $\mathbf{e}$ has L zeros at the left side. This ensures that all vectors are of the same length ($d + E$) and have unique representations for each locus and environment. Additionally, a column of ones is added at the right end to enhance the model's capability to capture epistasis. This is motivated by our analytical results (see SI Appendix, Section 6). The combined genotype and environment vectors are processed through multiple attention layers, capturing interactions between loci within the environmental context. The output from the final attention layer is used to predict the phenotype $y^{(g)}_{\text{pred},\alpha}$ and update the model parameters (as shown in Fig. 1e).

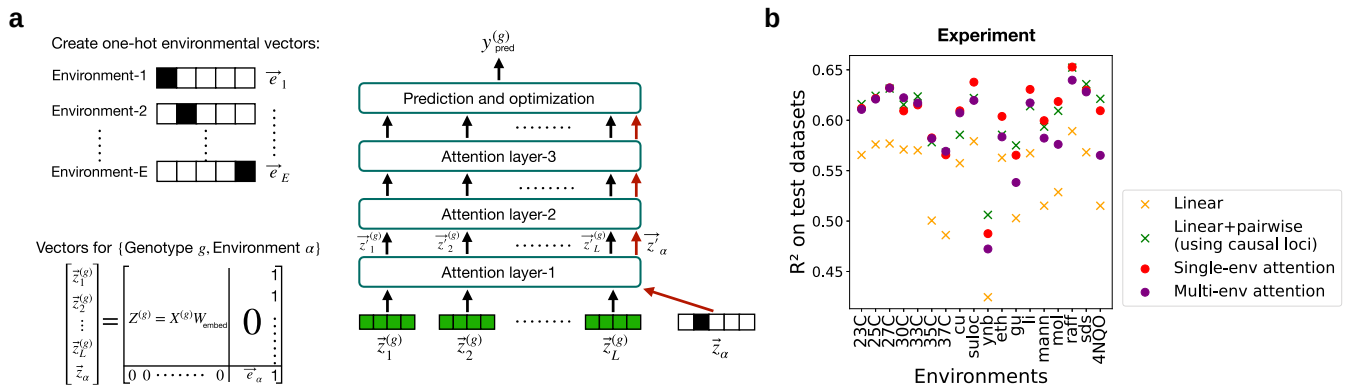

Fig. 4. Multienvironment attention-based model and performance comparison. a) Schematic of multienvironment attention-based architecture. One-hot environmental vectors are created for each environment, combined with genotype embeddings $Z^{(g)}$, and processed through multiple attention layers to predict phenotypes $y^{(g)}_{\text{pred}}$. b) R^2 performance on test datasets for linear, linear+pairwise, and single-environment and multienvironment attention-based models (with $d = 12$) across various environments. Note that the linear and linear+pairwise models were trained separately for each environment.

Prediction on yeast data

The results from applying the model to yeast data are shown in Fig. 4b. This figure compares the performance (R^2 on held-out test datasets) of the multienvironment attention-based model with traditional linear and linear+pairwise models, as well as the single-environment attention-based model, across different environments. The multienvironment attention-based model performs better than the single-environment linear models and is comparable to the single-environment linear+pairwise models. We find that the single-environment attention-based model performs better than the multienvironment model. Because the joint model shares parameters across environments, it must fit heterogeneous G–P relationships and noise profiles. With limited data, this can cause negative transfer, pulling the shared representation away from the per-environment optimum. Our results show that it is possible to train a single model that can capture epistatic interactions across multiple environments. In Fig. S7, we show additional results for alternative different multienvironment attention architectures.

We note that, unlike in protein language models—where attention maps often correlate with structural features (33)—we have found that the attention maps of the learned models are hard to interpret (see Fig. S8). This may reflect the fact that the outputs of our models are continuous rather than discrete, or other complexities of the underlying biology of predicting growth rates from genotype.

Transfer learning to new environments

One of the crucial applications of the multienvironment attention-based architecture is its ability to predict phenotypes for environments where data are sparse. This capability to transfer information from data rich environments to new environments with limited data is known as transfer learning (see Fig. 5a). Since attention-based mechanisms are known to excel at transfer learning, we wanted to ask if our multienvironment architecture could make predictions in a new environment with limited data.

To test this ability, we focused on a subset of seven environments from the yeast QTL experiments in Ref. (10), where temperature was varied from 23 °C to 37 °C. We started by constructing a set of synthetic G–P maps that mimicked the experimentally measured correlation structure across temperatures (see SI Appendix, Section 2 for details). We then trained the model on all the training data for six of the temperatures (~ 7.2×10^4 individuals), while varying the amount of training data used for the final temperature from a single individual to the full training dataset (~ 7.2×10^4 individuals). See embedding and training details in SI Appendix, Section 5. The results are shown in Fig. 5b. We found that with as little as 100 training data points, the multienvironment architecture can successfully transfer information to new temperatures. This shows that our multienvironment attention model is capable of transfer learning on our synthetic dataset.

Having established that transfer learning is possible on synthetic data, we repeated this exercise on data from the QTL experiments in Ref. (10). As can be seen in Fig. 5c, the multienvironment attention-based model was once again able to learn genotype-to-phenotype maps in a new temperature with as little as 100 training data points. These results suggests that transfer learning may be a generic feature of attention-based models for G–P maps. In contrast, there is no natural way to implement such a transfer learning approach within the context of standard regression-based methods.

Discussion

In this article, we explored the use of attention-based architectures for learning complex G–P maps. We demonstrated the potential of these models on both synthetic datasets and experimental yeast QTL data (10). Our attention-based models successfully learn complex epistatic interactions. We then explored a multienvironment attention-based model that can predict phenotypes across multiple environments. The multienvironment architecture is specifically designed to leverage shared information across a wide range of environmental conditions, allowing the model to integrate genetic and environmental data into a unified framework. We demonstrated that this multienvironment attention model is capable of "transfer learning" G–P maps in new environments with limited data. Collectively, our results show that attention-based architectures offer a powerful new class of models for learning G–P maps.

The logic underlying our attention-based architectures is fundamentally different from traditional regression-based methods used for learning G–P maps. Methods based on linear regression commonly parameterize the genetic architecture of complex

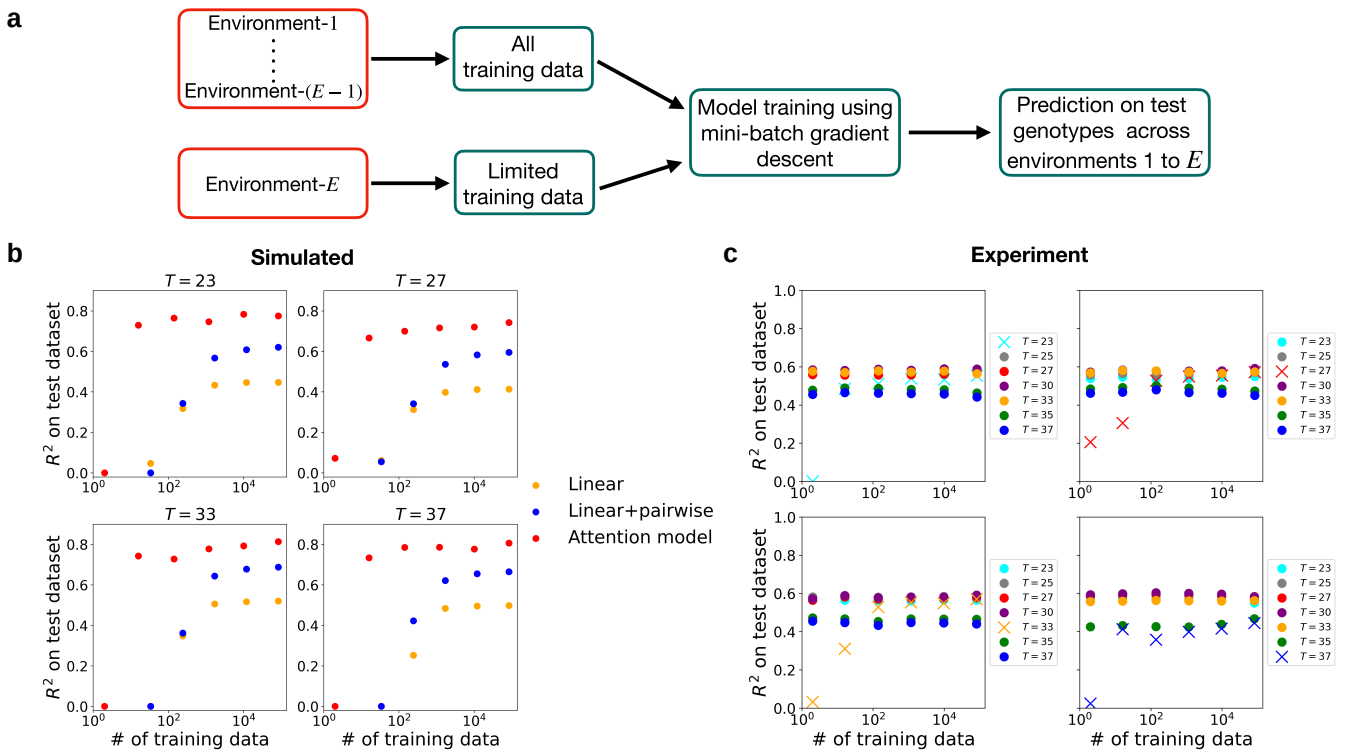

Fig. 5. Transfer learning. a) Schematic of our approach to transfer learning. Environment—E has fewer training data points than the other environments. Training data from different environments are sampled and used to train the model for phenotype prediction. b, c) R^2 on the test dataset for b) simulated data (with $d = 30$) and c) experimental yeast QTL data (with $d = 12$). Along the horizontal axes, "training data" refers specifically to the number of genotypes from the new temperature T included in the training set; for all other temperatures, the maximum available training set size is used. For the linear and linear+pairwise models in (b), the same number of training data points from temperature T is used. Simulated data are generated using Gaussian-distributed coefficients with $\epsilon = 0.3$ and $L = 100$.

traits in terms of an additive model, where the effects of loci are independent. Less often, these models also include pairwise epistatic interactions between loci. Unlike these traditional methods, attention-based models make no assumptions about how loci interact. Instead, they exploit the expressivity of the attention mechanism to directly learn these epistatic interactions from data with minimal assumptions.

We note that a practical concern is that only a subset of loci likely contributes meaningfully to any given phenotype. To avoid assuming prior knowledge of these causal loci, we do not preselect features. Instead, we mitigate linkage-driven noise by collapsing highly correlated (>94%) SNPs and analyzing a reduced set of loci.

There are numerous interesting potential extensions and future research directions using attention-based models. These include exploring alternative architectures for how genetic and environmental tokens interact and the use of nonlinear multilayer perceptron (MLP) layers and skip connections. In addition, it will be interesting to apply attention-based architectures to experimental datasets with more demographic structure and genomic diversity. In the yeast QTL dataset analyzed here, all 100,000 individuals are from a single cross between a lab and vineyard yeast strain. This limits both the complexity of the genomic backgrounds and the number and frequency distribution of alleles across loci. We expect attention-based models to perform even better on these richer and more complex datasets. A natural next step is to apply an attention-based framework to genetically diverse, structured cohorts with population stratification, heterogeneous allele-frequency spectra, long-range linkage disequilibrium, and rare variants.

Attention-based architectures also open the possibility of directly learning the genomic representations and demographic structure of populations from data at the same time as one learns the G–P map. One potential method for doing this is to randomly mask loci during training, analogous to certain large language models such as BERT (bidirectional encoder representations from transformers) (41). Such a procedure would force the attention-architecture to learn correlations between loci present in the training data set without the need for extensive preprocessing.

More generally, our work suggests that as the amount of sequence data increases, it should be possible to use more expressive statistical models to learn G–P maps. This opens up the exciting possibility that we may be able to harness advances in ML to learn subtle epistatic and gene–environment interactions directly from data. These techniques should also allow us to better understand how environments shape and modulate G–P maps.

Materials and methods

The SI Appendix provides detailed procedures and numerical parameters used to generate synthetic datasets, implement the attention-based models, and perform transfer learning. Here, we give an overview of our methods. We implemented single-environment and multienvironment attention-based neural architectures in PyTorch with three stacked attention layers followed by a linear regression head for phenotype prediction of growth rate. Genotypes for ~100,000 haploid *Saccharomyces cerevisiae* segregants from a BY × RM cross were determined once per

individual, and relative growth rates were measured for each segregant across 18 laboratory environments as described in Ref. (10). To mitigate linkage, we collapsed correlated SNPs and analyzed a set of 1,164 quasi-independent loci with a correlation coefficient magnitude <0.94. Models were trained on 85% of individuals, with a 15% validation split within the training data, and evaluated on a 15% held out test set. Optimization used mean squared error loss and the Adam optimizer, and predictive accuracy is reported as out of sample R^2. For simulations, we preserved the real genotype structure and generated phenotypes with a tunable mixture of additive terms and fourth-order epistasis, while adding Gaussian noise to emulate measurement variability. Embedding dimensions were selected by validation, with d equal to 12 for experimental data and d equal to 30 for synthetic data.

Acknowledgments

We would like to thank the Mehta group and Desai lab for helpful discussions.

Supplementary Material

Supplementary material is available at PNAS Nexus online.

Funding

P.M. and K.R. were supported by the National Institute of General Medical Sciences of the National Institutes of Health under Award No. R35GM119461 and by a Chan Zuckerberg Initiative Investigator Award to P.M. M.M.D. was supported by the National Science Foundation under Grant No. PHY-1914916 and by the National Institutes of Health under Grant No. GM104239. G.R. was supported by a joint research agreement between NTT Research, Inc. and Princeton University.

Author Contributions

K.R. wrote the code, performed the computations, and generated the figures. P.M. and M.M.D. jointly conceptualized the project. P.M., G.R., and K.R. designed and constructed the model architectures. M.M.D. and P.M. provided project administration and supervision. All authors contributed through scientific discussions, critical feedback, and analysis. All authors contributed to writing the manuscript.

Preprints

This manuscript was posted as a preprint at: https://www.biorxiv.org/content/10.1101/2025.04.11.648465v1.

Data Availability

All code used throughout this work is available on GitHub at our GitHub repository https://github.com/Emergent-Behaviors-in-Biology/GenoPhenoMapAttention.

References

1 Wagner GP, Zhang J. 2011. The pleiotropic structure of the genotype–phenotype map: the evolvability of complex organisms. *Nat Rev Genet*. 12(3):204–213.
2 Paaby AB, Rockman MV. 2013. The many faces of pleiotropy. *Trends Genet*. 29(2):66–73.
3 Solovieff N, Cotsapas C, Lee PH, Purcell SM, Smoller JW. 2013. Pleiotropy in complex traits: challenges and strategies. *Nat Rev Genet*. 14(7):483–495.
4 Rockman MV. 2008. Reverse engineering the genotype–phenotype map with natural genetic variation. *Nature*. 456(7223):738–744.
5 Mackay TFC. 2001. The genetic architecture of quantitative traits. *Annu Rev Genet*. 35(1):303–339.
6 Manolio TA, *et al.* 2009. Finding the missing heritability of complex diseases. *Nature*. 461(7265):747–753.
7 Visscher PM, Brown MA, McCarthy MI, Yang J. 2012. Five years of GWAS discovery. *Am J Hum Genet*. 90(1):7–24.
8 Visscher PM, *et al.* 2017. 10 years of GWAS discovery: biology, function, and translation. *Am J Hum Genet*. 101(1):5–22.
9 Uffelmann E, *et al.* 2021. Genome-wide association studies. *Nat Rev Methods Primers*. 1(1):59.
10 Ba ANN, *et al.* 2022. Barcoded bulk QTL mapping reveals highly polygenic and epistatic architecture of complex traits in yeast. *Elife*. 11:e73983.
11 Petti S, Reddy G, Desai MM. 2023. Inferring sparse structure in genotype–phenotype maps. *Genetics*. 225(1):iyad127.
12 Myles C, Wayne M. 2008. Quantitative trait locus (QTL) analysis. *Nat Educ*. 1(1):208.
13 Powder KE. Quantitative trait loci (QTL) mapping. In: *eQTL analysis: methods and protocols*. Springer; 2019. p. 211–229.
14 Yang J, Lee SH, Goddard ME, Visscher PM. 2011. GCTA: a tool for genome-wide complex trait analysis. *Am J Hum Genet*. 88(1):76–82.
15 Pazokitoroudi A, *et al.* 2020. Efficient variance components analysis across millions of genomes. *Nat Commun*. 11(1):4020.
16 Kumar SK, Feldman MW, Rehkopf DH, Tuljapurkar S. 2016. Limitations of GCTA as a solution to the missing heritability problem. *Proc Natl Acad Sci U S A*. 113(1):E61–E70.
17 Boyle EA, Li YI, Pritchard JK. 2017. An expanded view of complex traits: from polygenic to omnigenic. *Cell*. 169(7):1177–1186.
18 Hunter DJ. 2005. Gene–environment interactions in human diseases. *Nat Rev Genet*. 6(4):287–298.
19 Smith EN, Kruglyak L. 2008. Gene–environment interaction in yeast gene expression. *PLoS Biol*. 6(4):e83.
20 Llewellyn C, Wardle J. 2015. Behavioral susceptibility to obesity: gene–environment interplay in the development of weight. *Physiol Behav*. 152:494–501.
21 Korte A, *et al.* 2012. A mixed-model approach for genome-wide association studies of correlated traits in structured populations. *Nat Genet*. 44(9):1066–1071.
22 Moore R, *et al.* 2019. A linear mixed-model approach to study multivariate gene–environment interactions. *Nat Genet*. 51(1):180–186.
23 Nicholls HL, *et al.* 2020. Reaching the end-game for GWAS: machine learning approaches for the prioritization of complex disease loci. *Front Genet*. 11:350.
24 Zeng S, *et al.* 2021. G2PDeep: a web-based deep-learning framework for quantitative phenotype prediction and discovery of genomic markers. *Nucleic Acids Res*. 49(W1):W228–W236.
25 Zhou J, *et al.* 2019. Whole-genome deep-learning analysis identifies contribution of noncoding mutations to autism risk. *Nat Genet*. 51(6):973–980.
26 Li H, Zeng J, Snyder MP, Zhang S. PRS-Net: interpretable polygenic risk scores via geometric learning. In: *International Conference on Research in Computational Molecular Biology*. Springer; 2024. p. 377–380.
27 Cheng C-Y, *et al.* 2021. Evolutionarily informed machine learning enhances the power of predictive gene-to-phenotype relationships. *Nat Commun*. 12(1):5627.

28 van Hilten A, *et al.* 2021. Gennet framework: interpretable deep learning for predicting phenotypes from genetic data. *Commun Biol*. 4(1):1094.
29 Pirruccello JP, *et al.* 2022. Deep learning enables genetic analysis of the human thoracic aorta. *Nat Genet*. 54(1):40–51.
30 Dingemans AJM, *et al.* 2023. Phenoscore quantifies phenotypic variation for rare genetic diseases by combining facial analysis with other clinical features using a machine-learning framework. *Nat Genet*. 55(9):1598–1607.
31 Vaswani A, *et al.* 2017. Attention is all you need. *Adv Neural Inf Process Syst*. 30:5998–6008.
32 Lin T, Wang Y, Liu X, Qiu X. 2022. A survey of transformers. *AI Open*. 3(120):111–132.
33 Jumper J, *et al.* 2021. Highly accurate protein structure prediction with alphafold. *Nature*. 596(7873):583–589.
34 Rives A, *et al.* 2021. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc Natl Acad Sci U S A*. 118(15):e2016239118.
35 Meynard-Piganeau B, Feinauer C, Weigt M, Walczak AM, Mora T. 2024. TULIP: a transformer-based unsupervised language model for interacting peptides and t cell receptors that generalizes to unseen epitopes. *Proc Natl Acad Sci U S A*. 121(24):e2316401121.
36 Bengio Y, Courville A, Vincent P. 2013. Representation learning: a review and new perspectives. *IEEE Trans Pattern Anal Mach Intell*. 35(8):1798–1828.
37 Kingma DP, Ba J. 2014. Adam: A method for stochastic optimization. arXiv:1412.6980.
38 Mehta P, *et al.* 2019. A high-bias, low-variance introduction to machine learning for physicists. *Phys Rep*. 810(3):1–124.
39 Paszke A, *et al.* 2019. Pytorch: an imperative style, high-performance deep learning library. *Adv Neural Inf Process Syst*. 32:8026–8037.
40 Pan SJ, Yang Q. 2010. A survey on transfer learning. *IEEE Trans Knowl Data Eng*. 22(10):1345–1359.
41 Devlin J, Chang M-W, Lee K, Toutanova K. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805.