DNA language models are powerful zero-shot predictors of non-coding variant effects

Gonzalo Benegas[1], Sanjit Singh Batra[2], and Yun S. Song[2,3,*]

[1] Graduate Group in Computational Biology, University of California, Berkeley
[2] Computer Science Division, University of California, Berkeley
[3] Department of Statistics, University of California, Berkeley

Abstract

Variant effect prediction has traditionally focused on training supervised models on labeled data. Motivated by recent advances in natural language processing that have demonstrated substantial gains on diverse tasks by pre-training on large unlabeled data, however, unsupervised pre-training on massive databases of protein sequences has proven to be an effective approach to extracting complex information about proteins. Such models have been shown to learn variant effects in coding regions in a zero-shot manner. In a similar vein, we here introduce **GPN** (**G**enomic **P**re-trained **N**etwork) which can learn variant effects in non-coding DNA using unsupervised pre-training on genomic DNA sequence alone. Our model is also able to learn gene structure and DNA motifs without any supervision. We demonstrate the utility of GPN by showing that it outperforms supervised deep learning models such as DeepSEA trained on vast amounts of functional genomics data in *Arabidopsis thaliana*, a model organism for plant biology. Additionally, GPN trained on a single genome outperforms popular conservation scores such as phyloP and PhastCons, which are computed using aligned genomes from multiple species and can be used to predict the pathogenicity of variants that perturb highly conserved positions. We provide code (`https://github.com/songlab-cal/gpn`) to train GPN for any given species using its DNA sequence alone and learn corresponding zero-shot variant effects genome-wide.

Introduction

Cis-regulatory non-coding mutations underlie numerous human diseases [1] as well as crop domestication and improvement traits [2]. While advances in genome-editing technologies such as CRISPR have supercharged our ability to modulate gene expression for human disease therapies and crop improvement [3, 4, 5, 6], targeted engineering of gene expression requires thorough knowledge of cis-regulatory elements to precisely control either endogenous or introduced genes [7, 8]. Unfortunately, the exponentially large search space of promoter DNA sequence has precluded a deeper understanding of how transcription factors interpret cis-regulatory DNA sequence to control gene expression.

A popular approach to non-coding variant effect interpretation is to first train a supervised model to predict functional genomics data such as chromatin accessibility, transcription factor

*To whom correspondence should be address: yss@berkeley.edu

binding or gene expression, and then score variants based on how much they disrupt these predictions. This approach was pioneered by DeepSEA [9] (using 919 functional genomics tracks) and further improved upon recently by Enformer [10] (6956 tracks) and Sei [11] (21,907 tracks). The success of this approach relies on having good quality functional genomics data from a large variety of cell types, which is prohibitively expensive to generate for most species. An alternative approach is to train a supervised model to distinguish putatively benign and putatively pathogenic variants inferred from conservation patterns in closely related species [12, 13].

Another approach is to focus on a particular class of non-coding variants. For example, using sequence alone, training a classifier to predict where splicing occurs has been a successful strategy to predict the impact of cryptic splice variants [14, 15]. To predict the impact of regulatory variants, a successful strategy employed by Lee *et al.* [16] was to train a support vector machine to distinguish putative regulatory sequences from random genomic sequences. Recently, a deep learning model trained to predict Hi-C signal from sequence was able to demonstrate the ability to predict the impact of regulatory variants impacting DNA folding within the nucleus [17]. Zeng and Gifford [18] successfully trained a deep learning model to predict DNA methylation level of CpG sites from sequence which allowed them to predict non-coding variant effects on DNA methylation.

In the field of natural language processing, impressive advances have been made by pre-training language models on large text corpora. Pre-trained models, such as BERT [19], can be fine-tuned to achieve high performance on downstream tasks such as sentiment analysis. Minerva [20], a recent language model, has shown promising performance on even more complex tasks such as undergraduate-level STEM problems. Pre-training on large amounts of unlabeled sequences also holds a promise for biology. Recently, state-of-the-art performance in missense variant effect prediction was achieved by training unsupervised models on large databases of protein sequences [21] or their multiple sequence alignments [22]. Large language models are capable of predicting missense variant effects in a zero-shot manner, i.e., without any further training on labeled data. However, language models trained on non-coding DNA sequences [23, 24, 25, 26, 27, 28, 29] have not yet demonstrated the ability to make accurate zero-shot variant effect predictions.

In this paper, we introduce **GPN** (**G**enomic **P**re-trained **N**etwork), a DNA language model capable of accurate zero-shot prediction of non-coding variant effects. It is a unified approach that can predict the impact of all non-coding variants in a genome, and we demonstrate its utility by showing that it achieves state-of-the-art performance in *Arabidopsis thaliana*, a model organism for plant biology as well as a source of insight into human disease [30]. Despite being trained only on DNA sequence, GPN outperforms the DeepSEA model trained on functional genomics data from *Arabidopsis thaliana*. Furthermore, GPN trained on only a single genome outperforms widely-used conservation scores such as phyloP and PhastCons [31, 32, 33] which rely on whole-genome alignments of 18 closely related species. GPN's internal representation of DNA sequences is able to accurately distinguish genomic regions such as introns, untranslated regions and coding sequences. The confidence of GPN's predictions can also be used to illuminate regulatory grammar such as transcription factor binding motifs. Our results pave way for building state-of-the-art non-coding variant effect predictors for any given species using its genomic sequence alone, even in the absence of expensive functional genomics data.

Results

Unsupervised clustering of genomic regions. Using the *Arabidopsis thaliana* reference genome, we pre-trained a language model based on convolutional neural network to predict masked nucleotides from their local genomic context (Figure 1, Materials and Methods).

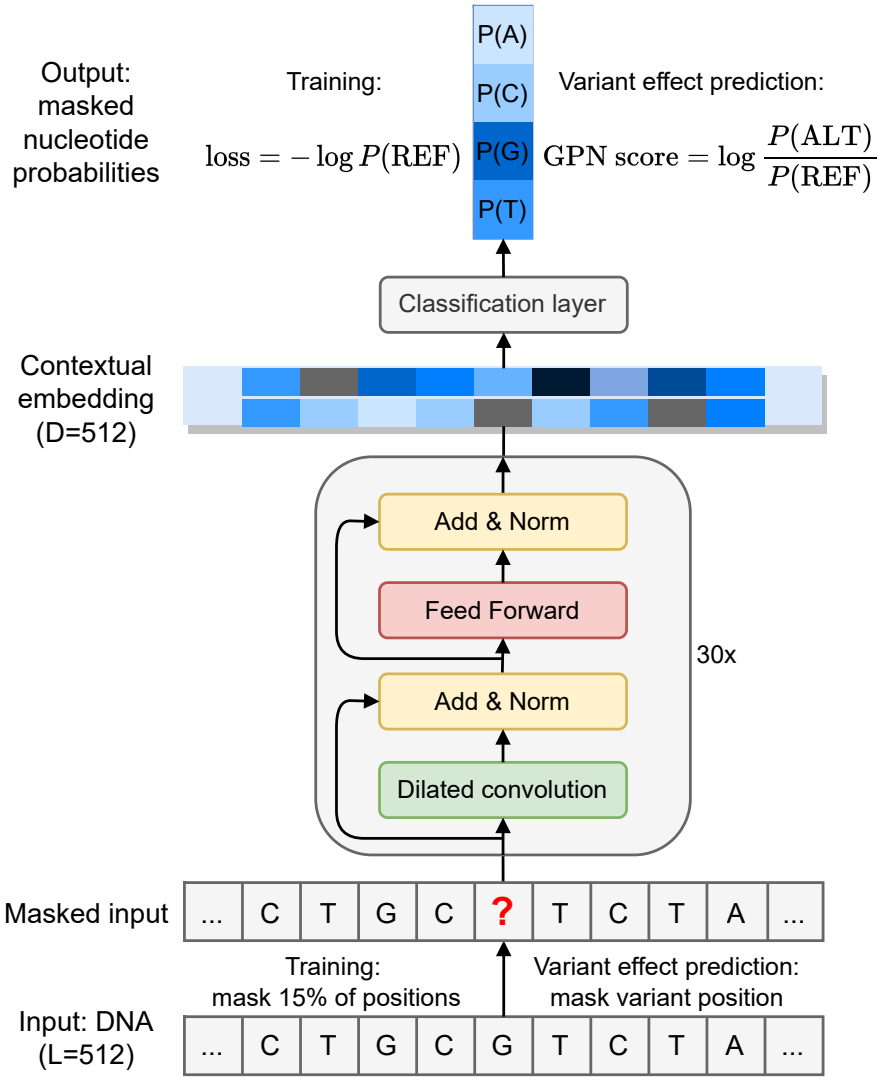


Figure 1: Overview of GPN (Genomic Pre-trained Network). The input is a 512-bp DNA sequence where certain positions have been masked, and the goal is to predict the nucleotides at the masked positions. During training, 15% of the positions are masked. During variant effect prediction, only the variant position is masked. The sequence is processed through a convolutional neural network resulting in a high-dimensional contextual embedding of each position. Then, a final layer outputs four nucleotide probabilities at each masked position. The model is trained on the reference sequence with the cross-entropy loss. The GPN variant effect prediction score is defined as the log-likelihood ratio between the alternate and reference allele. L: window length in base pairs. D: embedding dimension. REF: reference allele. ALT: alternate allele.

To understand how well the model has learned the structure of the genome, we averaged GPN's contextual embeddings (512 dimensions) of nucleotides over 30 base pair (bp) windows and visualized them with UMAP [34] (Figure 2a). We observed that GPN, which was trained without any supervision, has learned to distinguish genomic regions such as intergenic, introns, coding sequences (CDS) and untranslated regions (UTR). Unsupervised Leiden clustering [35] of the above model embeddings in a randomly sampled 1 Mb region of the genome led to the identification of eight

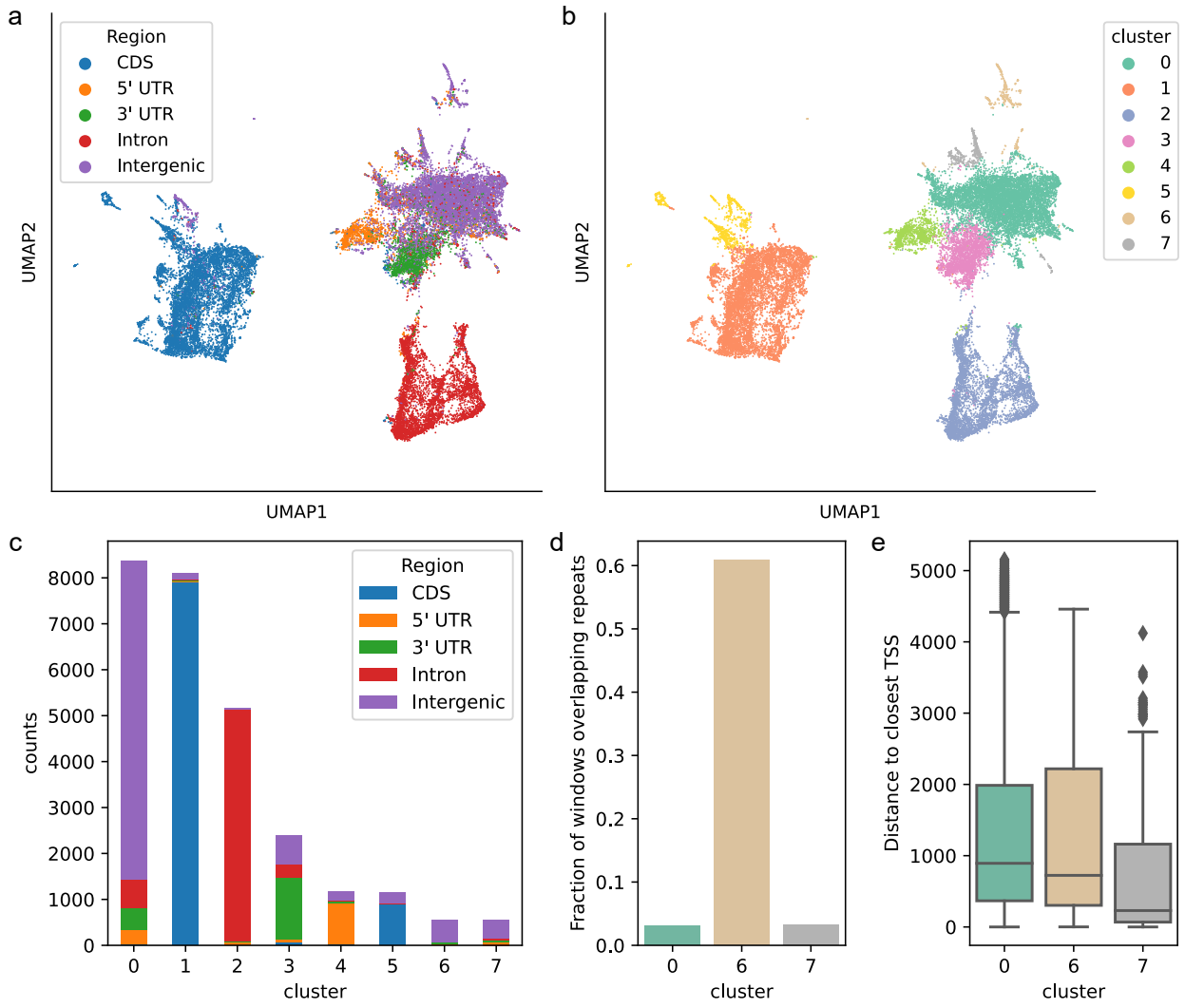


Figure 2: Unsupervised clustering of genomic windows. (a) UMAP visualization of model embeddings averaged over non-overlapping 30-bp windows in a megabase region, annotated with gene region. (b) Unsupervised Leiden clusters obtained from the averaged embeddings. (c) Number of windows in each cluster. (e) Fraction of windows overlapping a repetitive element, among majority-intergenic clusters. (d) Distance to closest TSS among majority-intergenic clusters.

distinct clusters (Figure 2b). The largest five clusters have a majority of intergenic, CDS, intronic, 3' UTR and 5' UTR genomic windows, respectively (Figure 2c). Overlap between 3' UTR and intergenic windows in cluster 3 suggests that either the separation between 3' UTR and intergenic regions is less well understood by the model, or there is a subset of intergenic regions that closely resemble 3' UTRs in sequence composition. It is also conceivable that some of these regions are misannotated. Cluster 5, which predominantly contains CDS windows, is enriched in genes associated with protein kinase activity when compared to genes from cluster 1 (adj. $p = 5.7 \times 10^{-3}$, [36, 37, 38]). Among clusters with a majority of intergenic windows, those in cluster 6 have a particularly high overlap with repetitive elements (Figure 2d). Intergenic windows in cluster 7 tend to be closer to transcription start sites (TSS) (Figure 2e), and hence this cluster is enriched in promoter regions.

DNA motifs revealed by high-confidence model predictions. We hypothesized that GPN might be able to predict, with high confidence, masked nucleotides belonging to short DNA motifs. With this in mind, we independently masked each position in a set of candidate regions and obtained the model output distribution over nucleotides, given its context. To visualize these distributions, we created sequence logos [39], where the height of each letter is proportional to its probability, and the overall height is given by the information content, measured in bits (Figure 3). The start codon (ATG) clearly stands out from its surrounding nucleotides (Figure 3a). This means the model is very certain of the presence of a start codon at this location, and can effectively remember its motif. In the immediate upstream region, the model shows a slight preference for A nucleotides, which are over-represented [40] and are associated with increased translational efficiency [41]. The model is not, however, predicting a global consensus sequence, but rather a context-specific distribution. For example, in a different gene depicted in Figure 3b, the model predicts that upstream C nucleotides are preferred over A nucleotides. Sequence logos similarly reveal that the model has learned the splice donor and splice acceptor dinucleotide motifs (Figure 3c,d respectively). More remarkably, the sequence logo in Figure 3e indicates that the model has also learned the nine-letter binding site motif for transcription factor AT1G72740, previously annotated with FunTFBS [33].

Zero-shot variant effect prediction. GPN can be used to compute a pathogenicity score for any single-nucleotide polymorphism (SNP) in the genome using the log-likelihood ratio between the alternate and reference allele (GPN score, Figure 1). We evaluated the distribution of GPN scores for all possible SNPs in a 100 kb region and aggregated the results across variant types (Figure 4a, full distribution in Supplementary Figure S1). We observed that the lowest GPN scores were assigned to variants that significantly disrupted the open reading frame, such as splice donor, splice acceptor, stop gained and start lost variants, which is consistent with well-known notions of pathogenicity. We also observed that missense variants had lower GPN scores than synonymous variants.

In order to assess whether variants with low GPN scores are under negative selection in naturally occurring accessions of *Arabidopsis thaliana*, we computed GPN scores on the test chromosome for SNPs present in the 1001 Genomes Project [42]. We observed a monotonic relationship between GPN scores and observed allele frequency in the 1001 Genome Project (Figure 4b). For instance, SNPs with the bottom 1% GPN scores have a mean allele frequency of 1%, while those with the top 1% GPN scores have a mean allele frequency of 5% (full distribution in Supplementary Figure S2).

To evaluate the concordance between GPN scores and allele frequency, we performed a Fisher's exact test for all SNPs on the test chromosome (Figure 4c). Rare variants are 2.73 times enriched in pathogenic scores, when defined as the lowest 0.1% of GPN scores. We computed an odds ratio for different variant types as well as different thresholds for defining pathogenic scores, across a variety of models (Figure 4d, additional variant types in Supplementary Figure S3). As we increased the stringency for calling a variant pathogenic (from the lowest 10% to only the lowest 0.1% scores), the odds ratio for enrichment in rare variants increased, for most models under consideration. At the most stringent cutoff for calling pathogenicity, evaluated on all variants, GPN achieved a higher odds ratio when compared to other models. This suggests that GPN may be a better predictor of pathogenicity or deleteriousness for non-coding variants genome-wide.

Notably, GPN often achieved higher odds ratios than conservation scores such as phyloP and PhastCons, which leverage information from whole-genome alignments with 18 closely related species. It is illustrative to look at the rapidly saturating odds ratios of phyloP, most pronounced in missense variants. This is because all positions that are perfectly conserved across the 18 species in the whole-genome alignment, which is the case for many missense variants, are assigned the same phyloP score. GPN and PhastCons, however, can distinguish pathogenicity of variants among

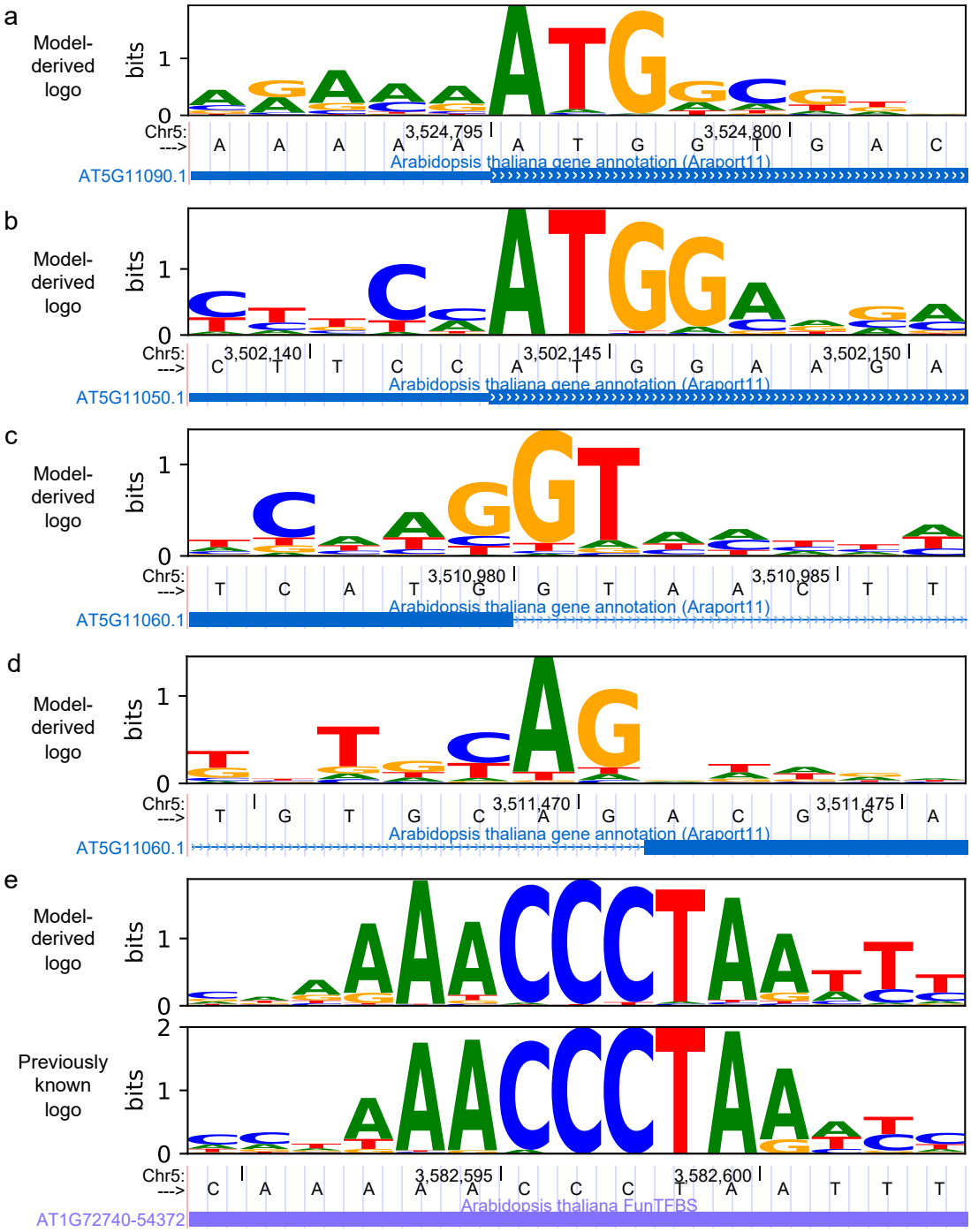
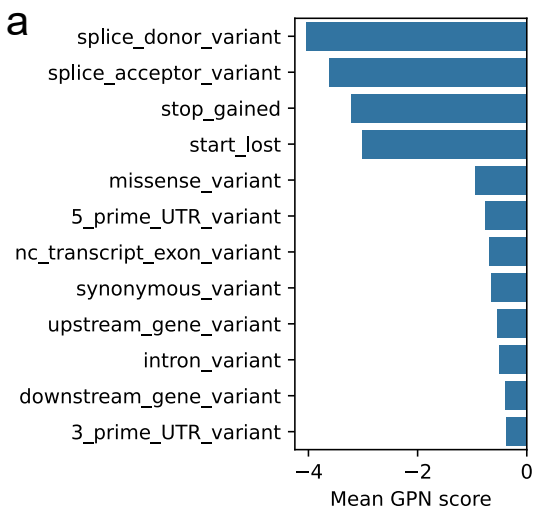
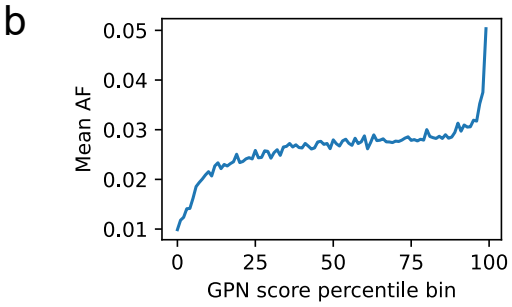


Figure 3: Sequence logos derived from model predictions. Each position was independently masked and the model distribution over the four nucleotides was used to construct a sequence logo. The height of each letter is proportional to its probability, while the overall height at each position is equal to 2 minus the entropy of the distribution. (a) Start codon for AT5G11090. (b) Start codon for AT5G11050. (c) Splice donor. (d) Splice acceptor. (e) Transcription factor binding site previously annotated by FunTFBS.





c

	Rare (AC=2)	Common (AC ≥ 20)
Pathogenic (lowest 0.1% GPN scores)	708	200
Benign (rest of GPN scores)	511805	394533

All variants OR: 2.73, p-value: 5e-42

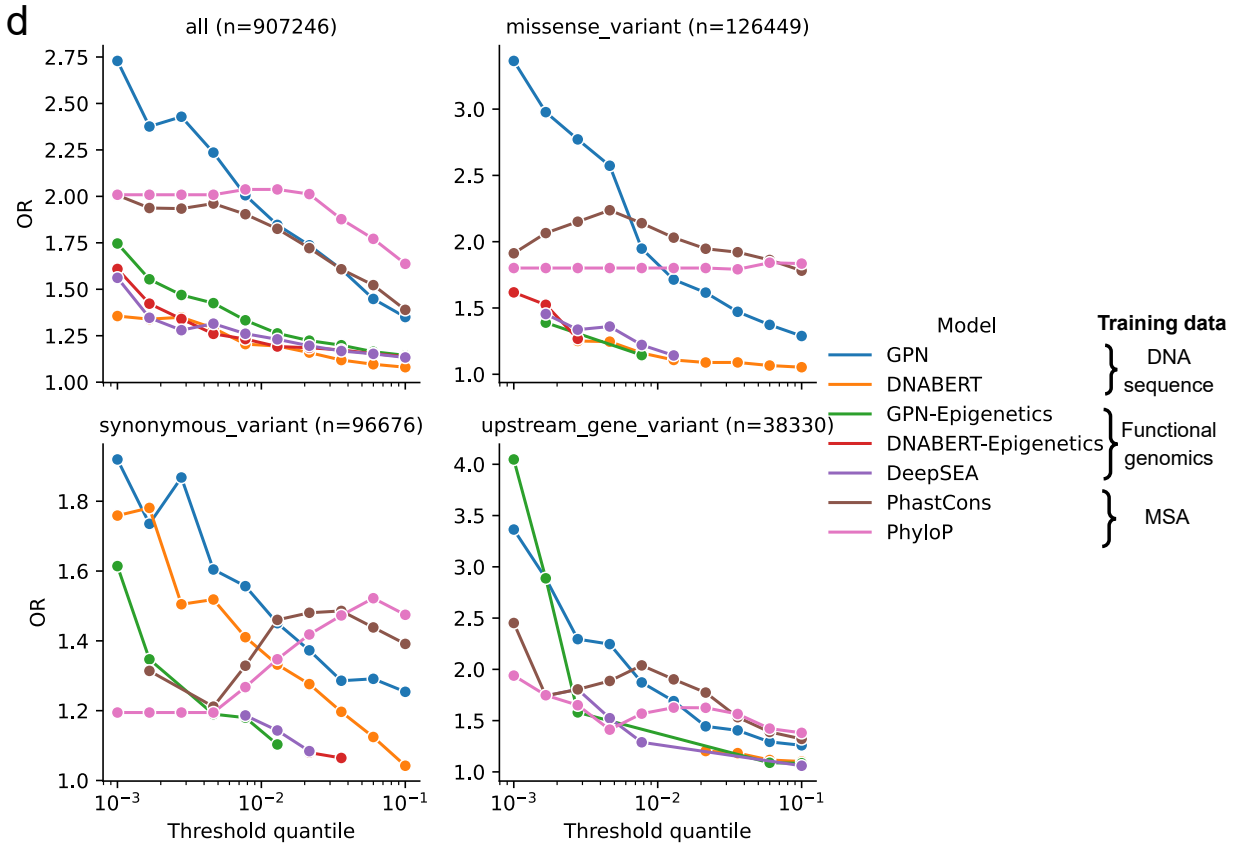


Figure 4: Variant effect prediction results. (a) Mean GPN score across variant types, for all possible single-nucleotide polymorphisms (SNPs) in a 100 kb region. (b) Mean allele frequency in each percentile bin of GPN scores of SNPs observed in the 1001 Genomes (chromosome 5). (c) Contingency table and odds ratio showing enrichment of pathogenic GPN scores in rare variants. AC: allele count. (d) The x-axis corresponds to model score thresholds used for defining pathogenicity. The y-axis corresponds to odds ratios obtained for various models, across different variant types (only significant odds-ratios are plotted). OR: odds ratio.

those in perfectly conserved sites as well. Additionally, approximately 6% of the SNPs on the test chromosome did not have an assigned phyloP or PhastCons score due to lack of alignments, but could be effectively scored by GPN (for consistency these are not included in Figure 4).

GPN also substantially outperformed the human DNA language model DNABERT [24] (without any further training on the *Arabidopsis thaliana* genome), which we adapted for zero-shot variant effect prediction, an application not reported by its authors.

In order to benchmark zero-shot variant effect prediction against traditional supervised training on functional genomics data, we trained a *de novo* DeepSEA model for *Arabidopsis thaliana* (Materials and Methods). Additionally, we fine-tuned both GPN and DNABERT on the above functional genomics data (GPN-Epigenetics and DNABERT-Epigenetics, respectively) as well. We evaluated these three models in their ability to predict functional genomics data on the test chromosome, and incidentally, GPN-Epigenetics achieved the highest predictive performance (Supplementary Figure S4).

GPN outperformed all of the *de novo* DeepSEA, GPN-Epigenetics and DNABERT-Epigenetics across most variant types (Figure 4d, Supplementary Figure S3). However, GPN-Epigenetics is competitive with GPN on upstream gene variants, likely because GPN-Epigenetics is able to successfully leverage functional genomics tracks to learn cis-regulatory grammar. In summary, GPN outperformed a variety of models when considering all variants, as well as within specific variant types such as missense and synonymous variants (Figure 4d). It is important to note that performance on all variants is not just the average performance in different variant categories; it also entails the ability to compare variant pathogenicity across categories. This capability, intrinsic to GPN, cannot be trivially obtained by an ensemble of models that are trained to predict variant effects in different types of variants.

Discussion

Here we present the first zero-shot non-coding variant effect predictor based on unsupervised pretraining of DNA language models. We demonstrate that GPN outperforms various other non-coding variant effect predictors in *Arabidopsis thaliana*, a model species for plant biology. We provide an alternative approach to the traditional paradigm of training supervised models on massive amounts of functional genomics data. Since GPN is trained only on DNA sequence, it can be readily applied to understudied non-model organisms even in the absence of extensive functional genomics data, while still providing state-of-the-art non-coding variant effect prediction.

An interesting observation is that GPN outperforms a *de novo* DeepSEA model trained on 106 *Arabidopsis thaliana* functional genomics tracks. One explanation could be that these tracks are missing critical transcription factors whose motifs are therefore implicitly excluded from the training data. More generally, entire variant types could be underrepresented in the functional genomics data, for instance RNA binding sites for splicing factors. Uncorrected technical biases in certain functional genomics assays may also have an adverse effect on variant effect prediction. Another challenge of applying supervised multi-task models to variant effect prediction is that it requires a post-hoc conversion of high-dimensional model predictions to a single scalar score, for which currently there is no consensus approach. In this work we use the euclidean norm of the change in predictions for the different functional genomics tracks to obtain a variant effect prediction score from each of the three models supervised on functional genomics data. However, Enformer [10] performs principal component analysis to distill 5313 predictions into summary statistics for variant effect prediction; Sei [11] adopts a more complex approach based on sequence classes obtained using unsupervised clustering. GPN, on the other hand, is optimized end-to-end to output a single scalar

variant effect prediction.

Although the current implementation of GPN outperforms state-of-the-art variant effect predictors for *Arabidopsis thaliana*, there is room for improving its training scheme. There is growing evidence that training bigger models with larger training data can lead to improved performance [43]. Therefore, it is conceivable that using DNA sequence of closely related species to expand dataset size could lead to improved GPN performance. Additionally, an insight that could be borrowed from protein modeling is to explicitly incorporate multiple sequence alignments [44, 45], which could lead to a further improvement; although this improvement is contingent on the quality of alignment in non-coding regions of the genome. Adding DNA-specific inductive biases such as reverse-complement equivariance [46] (as opposed to our current approach of averaging model outputs at test time across the two strands) also holds promise for DNA language modeling. Incorporating long-range information using recent developments in state space models [47] may also lead to improved performance. In summary, DNA language models are a powerful framework for non-coding variant effect prediction and we believe that it would be worthwhile to explore the above avenues to further improve GPN.

Materials and Methods

Pre-training. We donwloaded the TAIR10 reference genome of *Arabidopsis thaliana* from EnsemblPlants [48]. The genome was split into contigs and filtered for a minimum length of 512. Chromosomes 1-4 were used for training and chromosome 5 for testing.

We set up a masked language modeling task [19], in which 15% of the tokens in a nucleotide sequence were masked and had to be predicted from their context. In contrast with most DNA language models that tokenize sequences into overlapping k-mers [24, 26, 28] or use byte-pair encoding [23], we used bare nucleotides as tokens. While a thorough benchmark of different tokenization strategies is lacking, using single-nucleotide tokens makes interpretation easier, in particular for zero-shot variant effect prediction. To make the task more challenging, we masked spans of 1-5 nucleotides. 512-bp windows were sampled on-the-fly at each training step.

While language model pre-training successes were first showcased by transformer architectures, convolutional models have shown similarly good performance in natural language [49] and protein modeling [50]. In our initial experiments, we noticed that convolutional models converged faster than transformer models. The locality of convolutions may be a good inductive bias for modeling DNA sequences at this scale. The linear complexity of convolution also simplifies inference or fine-tuning on longer sequences such as entire chromosomes, which in the case of transformers might require chunking (with some overlap) and aggregating the results.

We implemented GPN, a convolutional neural network, using the Huggingface library [51]. The masked DNA sequence was one-hot encoded and then consecutively processed by 30 convolutional blocks. Each convolutional block consisted of a dilated convolutional layer followed by a feed-forward layer, with intermediate residual connections and layer normalization (Figure 1). Throughout the layers, the embedding dimension (number of convolutional filters) was kept fixed at 512. The dilation was increased exponentially up to a certain value and then cycled. A list of hyperparameters is displayed in Supplementary Table S1. We trained the model for 1.08 million steps, after which the test perplexity started increasing. Training took approximately 11 days using 4 NVIDIA GTX 2080ti GPUs. A final perplexity of 3.01 was achieved on the test chromosome.

Analysis of model embeddings. Model embeddings were obtained from the megabase region `Chr5:3066700-4066700`. Embeddings from the forward and reverse strand were averaged, and then standardized. Embeddings were averaged over non-overlapping 30-bp windows. The

gene annotation was downloaded from EnsemblPlants. Windows with ambiguous annotation (e.g. 50% CDS and 50% intron) were excluded from the analysis. The annotation of repetitive elements was downloaded from http://ucsc.gao-lab.org/cgi-bin/hgTables?hgsid=167291_E9nY5UIAQRUOAR01xJAsum4vDukw. Leiden clustering was performed following similar steps to the single-cell clustering tutorial from Scanpy [52].

Motif analysis. Each position in the window `Chr5:3500000-3600000` was independently masked and the model distribution over nucleotides was extracted. The distribution was averaged between the results from the forward and reverse strands. Sequence logos were plotted using Logomaker [53]. The location of predicted functional TFBS was downloaded from http://plantregmap.gao-lab.org/download.php#Cis-elements-pretfbs. The known TFBS motif was obtained from http://planttfdb.gao-lab.org/tf.php?sp=Ath&did=AT1G72740.1.

Fine-tuning for epigenetic feature prediction. We fine-tuned GPN to predict epigenetic features in Arabidopsis from three categories: DNase I hypersensitive sites, histone modifications and transcription factor binding sites, naming this model GPN-Epigenetics. A total of 106 epigenetic features encompassing a variety of organs and conditions were downloaded from PlantRegMap [33] http://plantregmap.gao-lab.org/download.php#cis-elements. Each window of 200 bps in the genome was intersected with the called peaks, and overlaps of at least 100 bps were considered as positive examples. The window was then expanded with flanks up to a total length of 1kb. The task was multi-objective prediction of all 106 binary features, weighted with the square root of the ratio of negative to positive examples. Chromosome 5 was held out as test set. As a baseline, we compared against DeepSEA [9], a convolutional neural network designed for epigenetic profile prediction in humans. The final layer of the network was modified to produce predictions for the smaller set of 106 features instead of the original 919 features available for the human genome. We also compared against fine-tuning of DNABERT [24], a DNA language model pre-trained on the human genome. We had to modify their code to support more than one output. Since the window size exceeded 512 tokens with their tokenization, we did as recommended in their paper: split the window into smaller subsequences and concatenated their embeddings before passing it into a final classifier layer. In particular we split the 1000 bp window into three sequences of length 400 bp, overlapping 100 bp. Our model achieved the best area-under-the-precision-recall-curve (AUPRC), with a smaller advantage on histone modifications (Supplementary Figure S4).

Variant effect prediction. All possible SNPs in the region `Chr5:3500000-3600000` were generated and their consequences annotated with Ensembl Variant Effect Predictor [54] web interface https://plants.ensembl.org/Arabidopsis_thaliana/Tools/VEP. We scored variants by masking the position and calculating the log-likelihood ratio between the alternate and reference allele. Scores computed from the forward and reverse strands were averaged.

Allele frequency in the 1001 Genomes project was downloaded from https://1001genomes.org/data/GMI-MPI/releases/v3.1/1001genomes_snp-short-indel_only_ACGTN.vcf.gz. Variant consequences produced by Ensembl Variant Effect Predictor were downloaded from Ensembl Plants. Conservation scores were downloaded from http://plantregmap.gao-lab.org/download.php#alignment-conservation. Chromosome 5 variants were filtered to only include bi-allelic SNPs, called in at least 1000 individuals, with reference allele frequency larger than 50% and with defined conservation scores, resulting in a final total of 907,246 variants. For GPN, we calculated zero-shot scores as described before. For DNABERT, tokenized with overlapping 6-mers, we masked 6 positions around the variant site, and obtained the log-likelihood ratio between the alternate 6-mer and the reference 6-mer, predicted at one of the center masked positions. For conservation scores PhyloP and PhastCons, we simply flipped the sign to obtain a variant score, i.e., variants at conserved sites should be considered more pathogenic. For GPN-Epigenetics, DNABERT-Epigenetics

and DeepSEA, we derived a score from the opposite of the euclidean norm of the change in epigenetic feature predictions, i.e., variants that disrupt epigenetic features the most should be considered the most pathogenic. We noticed there were no singletons which can be explained by the fact that Arabidopsis is predominantly selfing. We defined rare variants as those with allele count equal to 2, and common variants as those with allele count 20 or more. Model scores were defined as pathogenic or benign based on a quantile threshold that we varied from 0.1% to 10%. We calculated odds ratio and p-value with Fisher's exact test.

Code availability. Code to reproduce all results, including instructions to load the pre-trained model, is available at https://github.com/songlab-cal/gpn.

Competing interest statement

The authors declare no competing interests.

Acknowledgments

We would like to thank Eyes Robson, Nilah Ioannidis and Allison Gaudinier for helpful discussions. This research is supported in part by an NIH grant R35-GM134922.

References

[1] Zhang F, Lupski JR (2015) Non-coding genetic variants in human disease. *Human Molecular Genetics* 24(R1):R102–R110.

[2] Wang X, et al. (2021) Dissecting cis-regulatory control of quantitative trait variation in a plant stem cell circuit. *Nature Plants* 7(4):419–427.

[3] Jinek M, et al. (2012) A programmable dual-RNA–guided DNA endonuclease in adaptive bacterial immunity. *Science* 337(6096):816–821.

[4] Rees HA, Liu DR (2018) Base editing: precision chemistry on the genome and transcriptome of living cells. *Nature Reviews Genetics* 19(12):770–788.

[5] Ma X, Zhang X, Liu H, Li Z (2020) Highly efficient DNA-free plant genome editing using virally delivered crispr–cas9. *Nature Plants* 6(7):773–779.

[6] Kang BC, et al. (2018) Precision genome engineering through adenine base editing in plants. *Nature Plants* 4(7):427–431.

[7] de Boer CG, et al. (2020) Deciphering eukaryotic gene-regulatory logic with 100 million random promoters. *Nature Biotechnology* 38(1):56–65.

[8] Jores T, et al. (2021) Synthetic promoter designs enabled by a comprehensive analysis of plant core promoters. *Nature Plants* 7(6):842–855.

[9] Zhou J, Troyanskaya OG (2015) Predicting effects of noncoding variants with deep learning–based sequence model. *Nature Methods* 12(10):931–934.

[10] Avsec Ž, et al. (2021) Effective gene expression prediction from sequence by integrating long-range interactions. *Nature Methods* 18(10):1196–1203.

[11] Chen KM, Wong AK, Troyanskaya OG, Zhou J (2022) A sequence-based global map of regulatory activity for deciphering human genetics. *Nature Genetics* pp. 1–10.

[12] Rentzsch P, Witten D, Cooper GM, Shendure J, Kircher M (2019) Cadd: predicting the deleteriousness of variants throughout the human genome. *Nucleic Acids Research* 47(D1):D886–D894.

[13] Quang D, Chen Y, Xie X (2015) Dann: a deep learning approach for annotating the pathogenicity of genetic variants. *Bioinformatics* 31(5):761–763.

[14] Jaganathan K, et al. (2019) Predicting splicing from primary sequence with deep learning. *Cell* 176(3):535–548.

[15] Cheng J, et al. (2019) Mmsplice: modular modeling improves the predictions of genetic variant effects on splicing. *Genome Biology* 20(1):1–15.

[16] Lee D, et al. (2015) A method to predict the impact of regulatory variants from DNA sequence. *Nature Genetics* 47(8):955–961.

[17] Fudenberg G, Kelley DR, Pollard KS (2020) Predicting 3d genome folding from DNA sequence with akita. *Nature Methods* 17(11):1111–1117.

[18] Zeng H, Gifford DK (2017) Predicting the impact of non-coding variants on DNA methylation. *Nucleic Acids Research* 45(11):e99–e99.

[19] Devlin J, Chang MW, Lee K, Toutanova K (2018) Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

[20] Lewkowycz A, et al. (2022) Solving quantitative reasoning problems with language models. *arXiv preprint arXiv:2206.14858*.

[21] Meier J, et al. (2021) Language models enable zero-shot prediction of the effects of mutations on protein function. *Advances in Neural Information Processing Systems* 34.

[22] Frazer J, et al. (2021) Disease variant prediction with deep generative models of evolutionary data. *Nature* 599(7883):91–95.

[23] Zaheer M, et al. (2020) Big bird: Transformers for longer sequences. *Advances in Neural Information Processing Systems* 33:17283–17297.

[24] Ji Y, Zhou Z, Liu H, Davuluri RV (2021) DNABERT: pre-trained bidirectional encoder representations from Transformers model for DNA-language in genome. *Bioinformatics* 37(15):2112–2120.

[25] Mo S, et al. (2021) Multi-modal self-supervised pre-training for large-scale genome data in *NeurIPS 2021 AI for Science Workshop*.

[26] Yang M, et al. (2022) Integrating convolution and self-attention improves language model of human genome for interpreting non-coding regions at base-resolution. *Nucleic Acids Research* 50(14):e81–e81.

[27] Hoarfrost A, Aptekmann A, Farfañuk G, Bromberg Y (2022) Deep learning of a bacterial and archaeal universal language of life enables transfer learning and illuminates microbial dark matter. *Nature Communications* 13(1):1–12.

[28] Gwak HJ, Rho M (2022) Vibe: a hierarchical bert model to identify eukaryotic viruses using metagenome sequencing data. *Briefings in Bioinformatics* 23(4). bbac204.

[29] Bai Z, et al. (2022) Identification of bacteriophage genome sequences with representation learning. *Bioinformatics*. btac509.

[30] Jones AM, et al. (2008) The impact of Arabidopsis on human health: diversifying our portfolio. *Cell* 133(6):939–943.

[31] Pollard KS, Hubisz MJ, Rosenbloom KR, Siepel A (2010) Detection of nonneutral substitution rates on mammalian phylogenies. *Genome Research* 20(1):110–121.

[32] Siepel A, et al. (2005) Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. *Genome Research* 15(8):1034–1050.

[33] Tian F, Yang DC, Meng YQ, Jin J, Gao G (2020) PlantRegMap: charting functional regulatory maps in plants. *Nucleic Acids Research* 48(D1):D1104–D1113.

[34] McInnes L, Healy J, Melville J (2018) Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

[35] Traag VA, Waltman L, Van Eck NJ (2019) From louvain to leiden: guaranteeing well-connected communities. *Scientific Reports* 9(1):1–12.

[36] Ashburner M, et al. (2000) Gene ontology: tool for the unification of biology. *Nature Genetics* 25(1):25–29.

[37] Gene Ontology Consortium (2021) The Gene Ontology resource: enriching a GOld mine. *Nucleic Acids Research* 49(D1):D325–D334.

[38] Mi H, Muruganujan A, Ebert D, Huang X, Thomas PD (2019) Panther version 14: more genomes, a new panther go-slim and improvements in enrichment analysis tools. *Nucleic Acids Research* 47(D1):D419–D426.

[39] Schneider TD, Stephens RM (1990) Sequence logos: a new way to display consensus sequences. *Nucleic Acids Research* 18(20):6097–6100.

[40] Nakagawa S, Niimura Y, Gojobori T, Tanaka H, Miura Ki (2008) Diversity of preferred nucleotide sequences around the translation initiation codon in eukaryote genomes. *Nucleic Acids Research* 36(3):861–871.

[41] Kim Y, et al. (2014) The immediate upstream region of the 5'-utr from the aug start codon has a pronounced effect on the translational efficiency in Arabidopsis thaliana. *Nucleic Acids Research* 42(1):485–498.

[42] Alonso-Blanco C, et al. (2016) 1,135 genomes reveal the global pattern of polymorphism in Arabidopsis thaliana. *Cell* 166(2):481–491.

[43] Kaplan J, et al. (2020) Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

[44] Rao RM, et al. (2021) Msa transformer in *International Conference on Machine Learning*. (PMLR), pp. 8844–8856.

[45] Jumper J, et al. (2021) Highly accurate protein structure prediction with alphafold. *Nature* 596(7873):583–589.

[46] Zhou H, Shrikumar A, Kundaje A (2022) Towards a better understanding of reverse-complement equivariance for deep learning models in genomics in *Machine Learning in Computational Biology*. (PMLR), pp. 1–33.

[47] Gu A, Goel K, Re C (2021) Efficiently modeling long sequences with structured state spaces in *International Conference on Learning Representations*.

[48] Yates AD, et al. (2022) Ensembl genomes 2022: an expanding genome resource for non-vertebrates. *Nucleic acids research* 50(D1):D996–D1003.

[49] Tay Y, et al. (2021) Are pretrained convolutions better than pretrained transformers? in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. (Association for Computational Linguistics, Online), pp. 4349–4359.

[50] Yang KK, Lu AX, Fusi N (2022) Convolutions are competitive with transformers for protein sequence pretraining in *ICLR2022 Machine Learning for Drug Discovery*.

[51] Wolf T, et al. (2019) Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

[52] Wolf FA, Angerer P, Theis FJ (2018) Scanpy: large-scale single-cell gene expression data analysis. *Genome Biology* 19(1):1–5.

[53] Tareen A, Kinney JB (2020) Logomaker: beautiful sequence logos in python. *Bioinformatics* 36(7):2272–2274.

[54] McLaren W, et al. (2016) The Ensembl Variant Effect Predictor. *Genome Biology* 17(1):1–14.

Supplementary Table and Figures

Table S1: Training hyperparameters

Parameter	Value
Window size (L)	512
Embedding dimension (D)	512
Convolutional blocks	30
Convolutional kernel size	9
Convolutional dilation schedule	1,2,4,8,16,1,2,4,8,16,...
Batch size	1K
Learning rate	10^{-3} for 1 M steps, then 10^{-4} for 80 K steps
Learning rate warmup	10 K steps

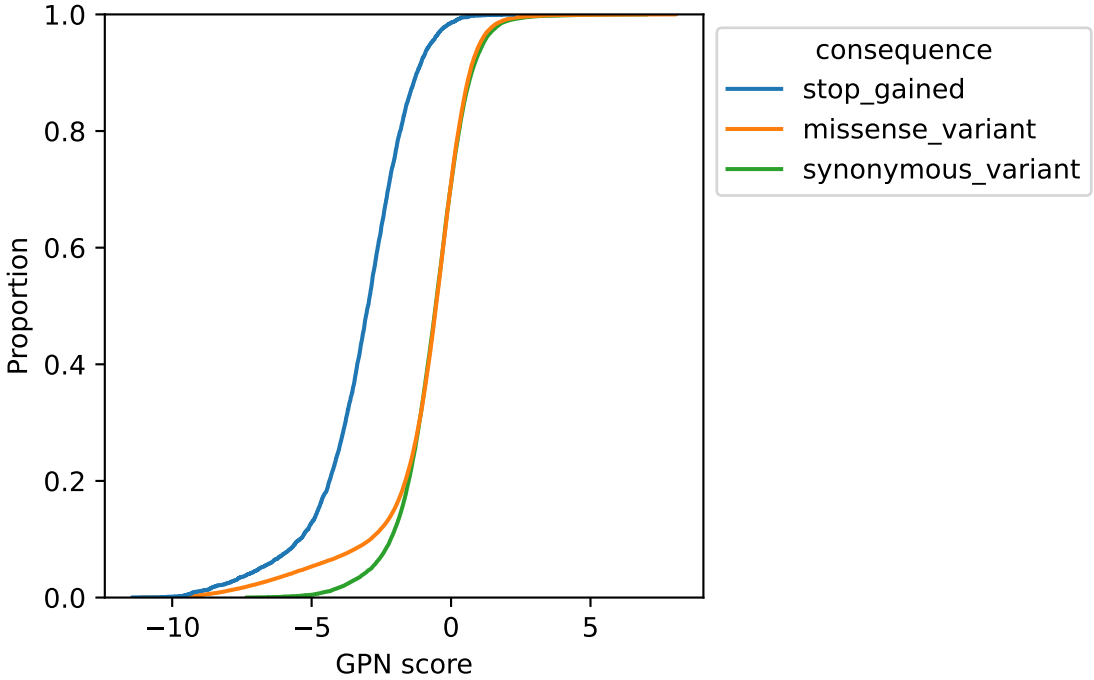


Figure S1: Cumulative distribution function of GPN scores for variants in specific categories, as described in Figure 4a.

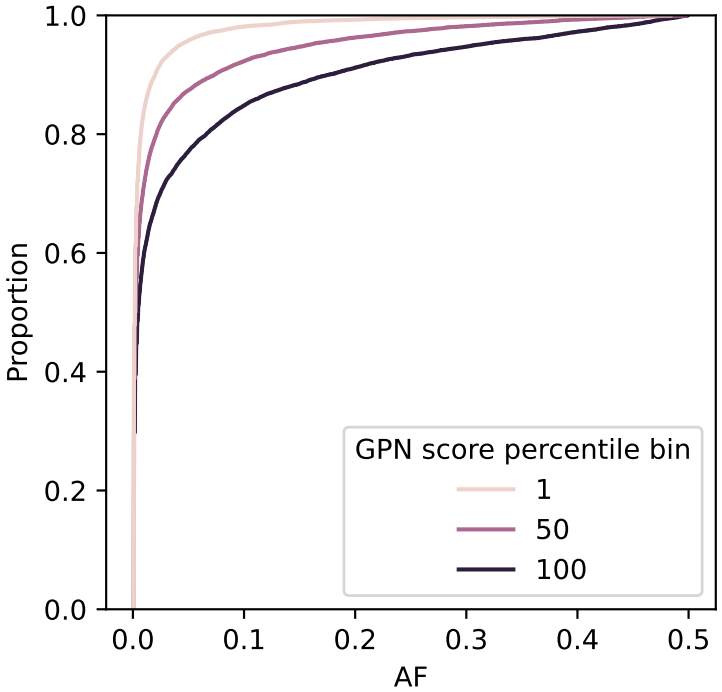


Figure S2: Cumulative distribution function of allele frequency (AF) for variants in different GPN score percentile bins, as described in Figure 4b.

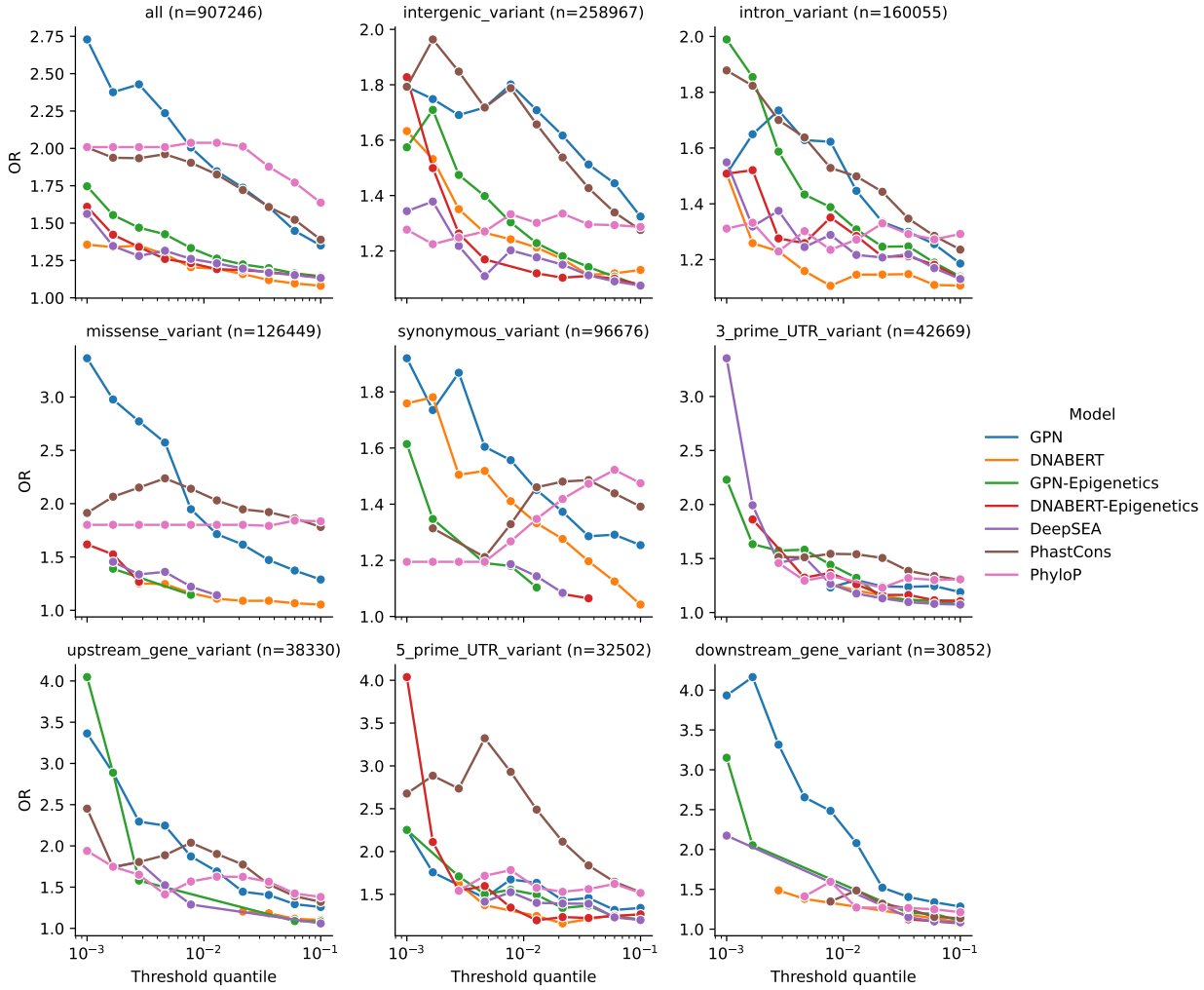


Figure S3: Variant effect prediction odds ratios for additional variant categories. Odds ratios (OR) were calculated as described in Figure 4d.

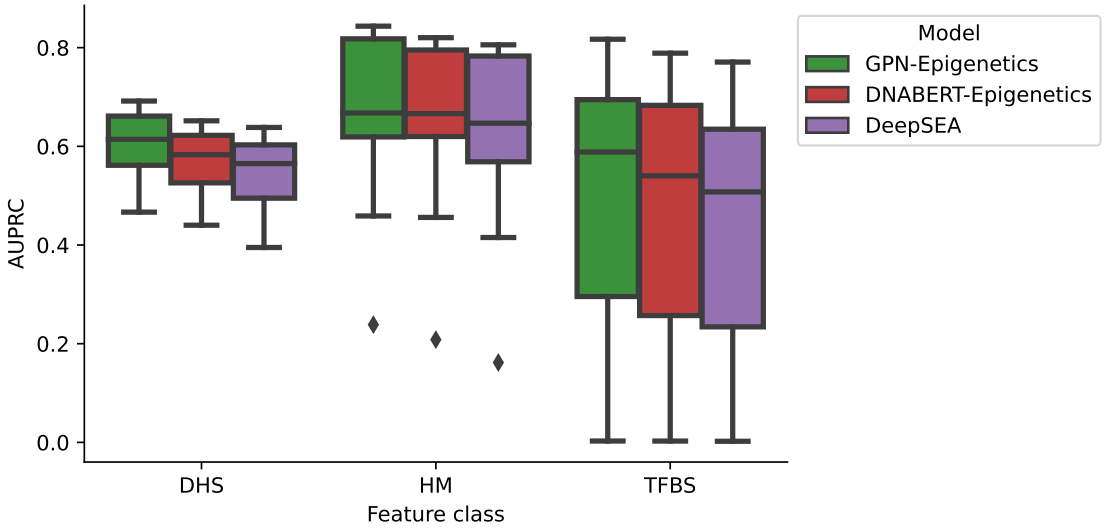


Figure S4: Results in the epigenetic features prediction task. DHS: DNase I hypersensitive sites. HM: histone modifications. TFBS: transcription factor binding sites.