Available online at www.sciencedirect.com

ScienceDirect

Current Opinion in Biomedical Engineering

Using machine learning to enhance and accelerate synthetic biology

Kshitij Rai[1,3], Yiduo Wang[1,4], Ronan W. O'Connell[1,4], Ankit B. Patel[5,6] and Caleb J. Bashor[1,2]

Engineering synthetic regulatory circuits with precise input–output behavior—a central goal in synthetic biology—remains encumbered by the inherent molecular complexity of cells. Non-linear, high-dimensional interactions between genetic parts and host cell machinery make it difficult to design circuits using first-principles biophysical models. We argue that adopting data-driven approaches that integrate modern machine learning (ML) tools and high-throughput experimental approaches into the synthetic biology design/build/test/learn process could dramatically accelerate the pace and scope of circuit design, yielding workflows that rapidly and systematically discern design principles and achieve quantitatively precise behavior. Current applications of ML to circuit design are occurring at three distinct scales: 1) learning relationships between part sequence and function; 2) determining how part composition determines circuit behavior; 3) understanding how function varies with genomic/host-cell context. This work points toward a future where ML-driven genetic design is used to program robust solutions to complex problems across diverse biotechnology domains.

Addresses
[1] Department of Bioengineering, Rice University, Houston, TX 77030, USA
[2] Department of Biosciences, Rice University, Houston, TX 77030, USA
[3] Graduate Program in Systems and Synthetic Biology, Rice University, Houston, TX 77030, USA
[4] Graduate Program in Bioengineering, Rice University, Houston, TX 77030, USA
[5] Department of Neuroscience, Baylor College of Medicine, Houston TX 77030, USA
[6] Department of Electrical & Computer Engineering, Rice University, Houston TX 77030, USA

Corresponding author: Bashor, Caleb J. (caleb.bashor@rice.edu)

Current Opinion in Biomedical Engineering 2024, **31**:100553

This review comes from a themed issue on **VSI: Synthetic Signaling and Engineering Cell Therapy**

Edited by **Wilson Wong** and **Ahmad Khali**

Received 20 April 2024, revised 3 July 2024, accepted 28 July 2024

Available online xxx

https://doi.org/10.1016/j.cobme.2024.100553

2468-4511/Published by Elsevier Inc.

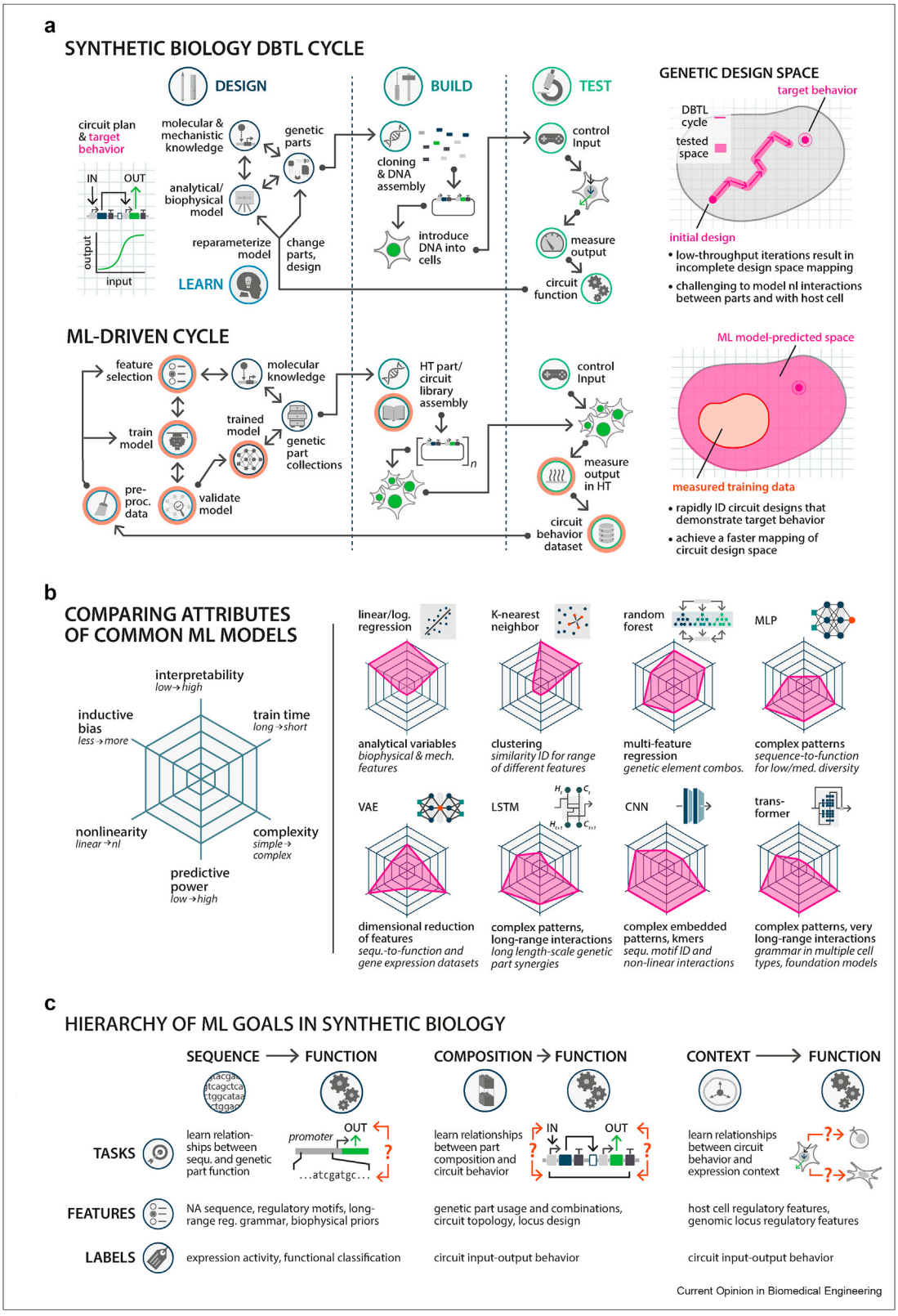
Keywords
Synthetic biology, Machine learning, Artificial intelligence, Genetic circuit design, Transfer learning, Active learning, Interpretable machine learning.

Introduction: Re-imagining the synthetic biology DBTL cycle with machine learning

Synthetic biology is the quantitative design of living systems at the genetic level [1–3]. Since its inception, a major focus in the field has been on engineering synthetic regulatory circuits to establish artificial control over cellular behavior [4,5]. Circuits with functions ranging from environmental sensing [6,7] and dynamic gene expression control [8–10], to directing cell differentiation and controlling population-level behavior [11–13], have been constructed in both model and non-model host organisms, with particular emphasis on *Escherichia coli*, yeast, and human cells. In the >20 years since the first demonstrations that DNA-encoded gene circuits can facilitate human-designed cellular behavior [14,15], the field has created a variety of circuits that are making biotechnological impacts in areas ranging from metabolic engineering [16,17] and bioproduction [18] to environmental remediation [19] and therapeutics [20,21].

Despite these advances, the vision of precise, programmable control over cellular behavior that originally motivated the field is far from fully realized, and synthetic circuit engineering remains a primarily slow, labor-intensive process that relies on low-to-medium throughput design-build-test-learn (DBTL) cycles (Figure 1a, top) [22,23]. The first step in a DBTL cycle involves designing a circuit's regulatory connections from molecular elements (e.g., promoters and transcription factors [TFs]) using first-principles biophysical models to quantitatively predict circuit behavior [24,25]. The elements are then embodied as DNA-encoded genetic parts that are assembled into circuits using molecular cloning approaches. After introducing circuits into cells, input–output behavior is experimentally assessed, typically using a single-cell assay (e.g., flow cytometry). Since measured circuit behavior nearly always deviates from the target on the first iteration, the model is adjusted or re-parameterized to account for apparent inaccuracies, and the cycle is repeated to test updated circuit designs. Through this

Figure 1



Reshaping synthetic biology DBTL cycles with ML. (**a**) Comparing the historical synthetic circuit DBTL cycle with one driven by ML. The design phase quantitatively defines a circuit's target input–output behavior and leverages existing knowledge of regulatory mechanism to identify genetic parts for its construction. Biophysical modeling frameworks with part-associated parameters are typically used to guide circuit design and predict behavior. In

circuit "tuning" process, it is possible to iteratively progress through a "genetic design space"—the range of part variant compositions available to construct a circuit—to reach a designated target behavior.

Why does navigating circuit design space remain challenging after more than two decades of effort? To a great extent, it is because of the inherently complex molecular nature of the cellular interior, where unforeseen coupling can occur between genetic parts or with native host cell machinery [26,27]. Because these interactions are non-linear and highly context-dependent, they violate basic assumptions about part modularity and challenge the predictive power of low-dimensional biophysical models, forcing the circuit engineering process away from predictive design and into a regime of *ad hoc* tinkering. In this piece, we argue that a potential solution to these challenges is offered by machine learning (ML), a branch of artificial intelligence (AI) that develops computational tools to recognize patterns and make functional predictions without explicit human input (see Box 1 for a brief summary of key ML terms and concepts) [28,29]. Rather than solving problems using first-principles parameterizations, ML models solve problems by passing large datasets through learning algorithms to "train" high-dimensional parameter sets. Because ML models have the potential to account for latent variables beyond those that can be explicitly derived, they can capture complex, non-linear patterns in the data that are not known *a priori*. In systems for which there is sparse or incomplete knowledge, ML models therefore have the potential to achieve higher predictive power than those that use analytical frameworks. In recent years, ML has been implemented across multiple technological domains, with notable examples that include reinforcement learning models such as AlphaGo [30] which defeated champion human GO players, and WaveNet, a CNN that performs accurate text-to-speech generation for a multitude of applications [31]. Incorporation of ML approaches into life-science research has produced rapid and significant progress in genetics [32], systems biology [33] and, perhaps most notably, protein structure prediction [34,35], where deep NN models trained on thousands of publicly available protein structures have outperformed historical prediction approaches that rely on parameterized energy functions. This has unlocked the ability to design protein sequences *de novo* that have desired structural and functional properties (well-reviewed elsewhere [36–38]). While ML is still in the early stages of being harnessed for forward engineering of synthetic regulatory systems, the potential exists for models trained on functional genetic part or circuit datasets to learn complex, high-dimensional design rules, rapidly yield mappings of circuit design space, and offer far more accurate predictions of circuit target behavior.

DBTL cycles that incorporate ML models will look very different from traditional ones (Figure 1a, bottom). Since a central goal is to furnish relevant data for training a model to make accurate design-to-function predictions, such a process would begin by conceptualizing a design space that contains the target behavior, and then devising a HT experimental strategy to build and test libraries of part sequence or circuit design variants that sample the space. A model trained with the resulting dataset would then be tested for predictive

the build phase, parts are assembled into circuits via molecular cloning methods and introduced into cells, and the test phase involves quantitative measurement of circuit input–output behavior. The cycle is closed during the learn phase, where measured circuit behavior is compared to expectations of the model, which may be reparametrized or used to identify circuit designs for testing in a subsequent cycle. (top right) Using this approach to navigate a genetic design space (grey area) is challenging, often requiring numerous DBTL iterations to converge on a target behavior. (bottom left) By modifying the DBTL cycle (orange highlights), an ML-driven process could enable more efficient design space navigation. Generating training datasets for ML-guided processes requires assembling collections of parts using HT molecular cloning approaches—either in arrayed or pooled format—to build circuit libraries. After introducing the libraries into cells, they are analyzed by HT measurement (e.g., with NGS as the output) to yield datasets relating aspects of circuit design (features) to output behavior (labels). The data are used to train a model, which is then validated for accuracy by using it to make behavior predictions on an unseen (test) dataset. If accuracy is poor, the model architecture may be adjusted, or additional data may be gathered in a subsequent cycle. (bottom right) Using this approach, measured data for a small subset of design space (orange area) can be used to create a more comprehensive and accurate mapping of the overall space (pink area) or efficiently converge on a target behavior within a limited number of cycles. pre-proc.; preprocess. (**b**) Comparing model classes commonly used for ML tasks in biological research. (left) Key model performance characteristics include interpretability**:** extent to which a human can understand a model's decision-making process; train time**:** time and computational resources required for model training; complexity**:** a model's ability to capture relationships between input features in the data; non-linearity**:** how well a model can learn structures in the data that cannot be described by linear analytical relationships between variables; inductive bias: the set of assumptions made by a model about the underlying relationship between features and labels; predictive power**:** ability of a model to make accurate predictions on unseen data. (right) Radar plots scoring model classes based on these characteristics. Use cases for each class are mentioned below each plot. MLP, multi-layer perceptron; VAE, variational autoencoder; LSTM, long short-term memory; CNN, convolutional neural network. (**c**) A hierarchy of ML tasks for learning genetic design-to-function models in synthetic biology. ML tasks learn relationships between design features and function (orange arrows) at different levels in a hierarchy that is reflective of biological organization. The goal of sequence-to-function tasks is to learn relationships between the NA sequence of a genetic part or element and its function (e.g., expression activity). For composition-to-function tasks, relationships between constituent parts and emergent circuit function are learned. For context-to-function tasks, the goal is to learn how circuit function varies in different host environments.

Box 1. Elementary ML terms and concepts

Creating an ML model to achieve a specific **goal** (e.g., understanding how sequence influences promoter activity) typically involves first defining a **prediction task**, where **features** (inputs) and **labels** (outputs) for the model are designated, with the objective of developing the model to accurately predict or classify labels when presented with a new set of features. Training datasets typically undergo **preprocessing**, which involves removing outliers or errors in the data, scaling the data to remove imbalance/skewness, and normalization of the data into an operable range. This is followed by **feature engineering**, which can involve generating new features from existing data (e.g., introducing new categorical groupings), converting categorical variables into numerical formats (e.g., **one-hot encoding**) or selecting/extracting the most relevant features to reduce dimensionality and improve model performance. Data are then randomly **split** between **training** and **held-out validation** and **test sets**. The training set (usually 80% of the data) is used to train the model's internal parameters, while the validation set (10%) is used to assess training progress. **Cross-validation**, an alternative validation method, involves partitioning data into equal subsets and iteratively using one subset for validation while training on the remaining ones. Averaging results across iterations mitigates biases from any specific split, and utilizes the entire dataset for training/validation. Model **accuracy** (how well predictions match actual data) is assessed by computing an error metric for both the training and validation sets. A high **bias** (**underfitted**) model may show poor accuracy for both training and validation sets, while one with high **variance** (**overfitted**) displays training set accuracy, but performs poorly on the validation set. Iterative progress toward accuracy in both sets can be achieved via **hyperparameter-tuning**—adjusting the architectural features of the model—or by introducing additional data. A final model performance evaluation is based on the **test set**, which serves as the sole indicator of unbiased model accuracy.

Choosing an appropriate **model class** depends on the task, dataset size, and available computational resources. **Supervised learning** refers to ML algorithms in which the prediction task requires that features be mapped to known labels. **Regression** models predict quantitative continuous outputs (e.g., gene expression level), while **classification** models predict discrete labels (e.g., presence or absence of a phenotype). **Unsupervised methods**, such as clustering, identify variants with similar input features and profiles and are used when underlying data structure characteristics are unknown or the labels are missing.

Inductive bias refers to the set of assumptions made by an ML algorithm about the underlying relationship between the labels and features. Simpler ML models include **linear/logistic regression**, which assume linear/log-odds relationships between features and labels and therefore have high inductive bias, but are human-interpretable and resource-efficient. Among the most popular models are **neural networks** (NNs), which exploit complex relationships by applying non-linear **activation-functions** and show utility for complex pattern recognition (e.g., NA sequence-to-function relationships) with lower inductive bias. NN models include deep (more than three layers) **multi-layer perceptrons** (MLPs) and **convolutional neural networks** (CNNs). Other common classes include: **Random forests**, highly robust ensemble decision tree models that individually have a step-wise inductive bias, excel at incorporating categorical features; **Language models** (LMs), (e.g., **transformer** and **long short-term memory** (LSTM) models), which capture sequential-dependencies and long-range and contextual interactions with a sequential-dependency inductive bias, but may require large quantities of training data. **Generative models** such **GANs** or **VAEs** learn the statistics of input data and can generate novel designs that follow the data distribution.

Model classes trade-off between **accuracy** and **interpretability** (how understandable a model's choices are to humans). Deep NNs can achieve high predictive accuracy on complex tasks but are black boxes; i.e., the functional mapping of inputs to outputs is encoded in a complex matrix of model "weights," which are not human-interpretable. Thus, they deliver excellent predictive power for unseen measurements, but are less desirable when a task requires understanding mechanistic principles. By contrast, linear regression models have lower predictive power but are more interpretable due to explicit correspondence between variables and features. Other key model considerations include **complexity**, which is determined by factors such as the number and type of parameters, the structure, and the amount of data that can be handled, all of which can affect the capacity of a model to represent complex relationships in data; and **non-linearity**, the ability to capture non-linear relationships between inputs and outputs. Selecting an optimal model class for a given task and determining the appropriate model complexity requires carefully balancing the trade-offs between the above mentioned considerations. While general-purpose models can capture broad patterns and scale to a broad array of tasks, they sacrifice performance on specific, narrow tasks. However, complex models with purpose-optimized architectures and training processes can achieve higher accuracy for specific tasks.

power and, in the absence of accurate target behavior prediction, subsequent DBTL cycles would be undertaken to reduce error by reconfiguring the model (e.g., with alternate hyperparameterization or featurization), gathering new training data, or selecting a model class better suited to learn the data (Figure 1b). Such a vision for data-centric, ML-driven circuit engineering is already starting to impact synthetic biology. As we describe, these efforts can be organized into a three-tiered hierarchy of modeling goals (Figure 1c): **1)** Sequence-to-function goals, which seek to understand relationships between nucleic acid (NA) sequences of individual genetic parts and their effect on circuit behavior; **2)** Composition-to-function goals, which focus on learning how combinations of genetic parts work together to yield emergent circuit input–output function, and; **3)** Context-to-function goals, where the aim is to predict how circuits behave across different host-cell contexts. We argue that developing ML-driven approaches for each set of goals, and ultimately integrating them, will hinge on the establishment of data collection pipelines and utilization of model architectures that match the scale and complexity of a given circuit engineering challenge.

Sequence-to-function models for ML-driven design of genetic parts

The most fundamental level of functional encoding for a genetic element (e.g., a promoter) is its NA sequence, which determines both its physiochemical properties

and interactions with other molecular species (e.g., binding to a TF). Decoupling the contributions that features of an NA sequence make to these properties is challenging for biophysical models [39] and has motivated application of ML to create predictive sequence-to-function mappings for different categories of genetic elements. These efforts were predicated on the widespread availability of next-generation sequencing (NGS) technology, primarily the Illumina platform [40], which has emerged in the past two decades as a primary workhorse for generating large-scale datasets for fundamental regulatory processes including transcription [41–43], translation [44,45], epigenetic regulation [46], and chromatin accessibility [47,48] (Figure 2a). Seminal work applying ML models to these datasets has provided deep insight into relationships between NA sequence and evolved regulatory grammar [49–52] and, critically, demonstrated an important ML scaling principle: parallel increases in NGS training dataset size and ML model complexity can yield higher predictive power. Examples highlighting this principle include Sei, a large deep NN trained on >22,000 ChIP datasets across >1,300 cell types [53]. Sei accurately classifies enhancer sequences and predicts the effects of mutations on cell type-specific expression activity. More recently, Avsec and colleagues developed Enformer [54], a transformer LM trained on >7,000 genomic and ChIP datasets that is capable of integrating features across long (>100 kb) length scales (e.g., long-range promoter-enhancer interactions) to accurately predict chromatin effects and DNA accessibility.

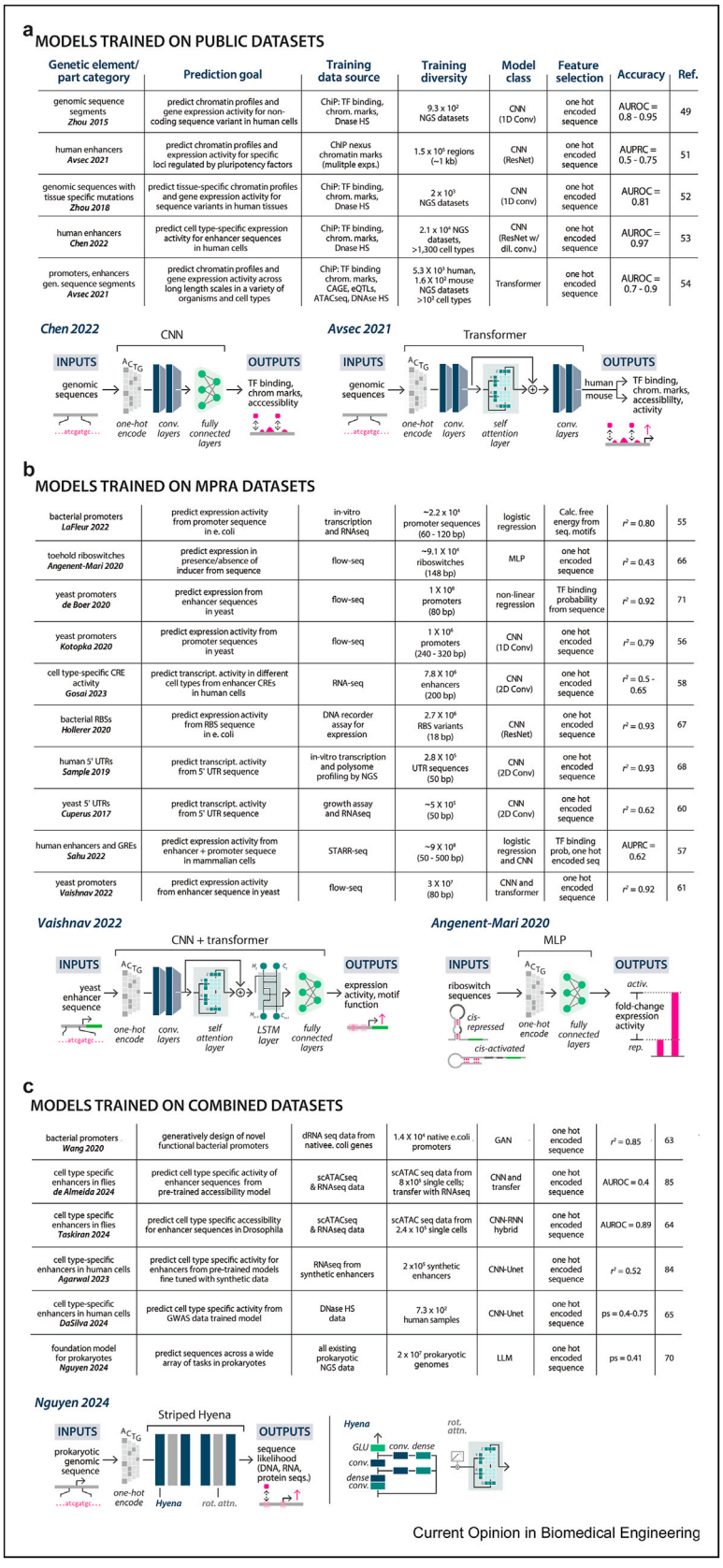
While these studies demonstrate that highly predictive models can be trained on native genomic sequence diversity, the development of massively parallel reporter assays (MPRAs)—an approach enabled by advances in low-cost DNA synthesis—has facilitated use of non-native sequence libraries to train models that predict activity for elements ranging from promoters [55,56] and enhancers [57,58], to UTRs and introns [59,60] (Figure 2b). In one notable study, data from a ~30M-member library of 80 bp enhancer sequences tested in yeast were used to train a transformer model to predict promoter activity [61]. The model predicted how enhancer mutations can "evolve" new, programmed expression activities. Following a similar approach to decipher TF regulatory grammar in mammalian cells, Sahu and colleagues measured diverse ($>10^8$) synthetic enhancer libraries (50–170 bp in length) and used the data to train logistic regression and CNN models, revealing that TFs regulate promoters in an additive manner with weak motif grammar, consistent with a billboard model of gene regulation [57]. Taken together, these studies have demonstrated the ability of ML models to explore non-native sequence design space to not only program artificial regulatory activity, but to gain insight into native regulatory function [62].

More recent studies have focused on using ML to explore NA sequence design for genetic elements explicitly intended for synthetic circuit construction [63–65] (Figure 2b). In one notable study, data from a library of $\sim 10^5$ RNA toe-hold switches were used to train an MLP model to predict fold-change expression between repressed and activated switch configurations, significantly outperforming regression-based models that rely on biophysical parameterization [66]. In another piece, expression measurements from $>10^5$ bacterial RBSs were measured and used to train a CNN-ResNet model to predict expression with high accuracy [67]. The use of a DNA recombinase-mediated activity assay enabled time series measurements across their library and led to less noisy data and greater model accuracy. In more recent work, activity measurements of a 280k-member library of 75 bp human 5′ UTRs were used to train a CNN to predict effects on tuning transgene reporter expression in human cells [68]. A follow-up to this work expanded the model's capabilities, training on ~200k shorter length sequences (25 and 50 bp) to learn cell type-specific expression [69].

A number of reports have appeared over the last year describing the use of data from both native and synthetic sequences to train models of varied size, featurization, and accuracy (Figure 2c). One standout is the recently reported Evo (Figure 2c, bottom), a versatile regulatory model trained on >20 million prokaryotic genomes that can make functional predictions for a range of tasks across DNA, RNA, and protein expression landscapes [70]. Like Enformer, Evo was developed as an early-generation foundation model: a large model (in excess of 10^7 parameters) trained across diverse datasets using significant computational power. Foundation models can be appropriated as community resources to achieve a variety of goals, potentially including assisting gene circuit design, as we discuss below.

The studies described in this section demonstrate the feasibility of using HT biological data to train highly predictive sequence-to-function ML models for different genetic part categories, and offer a new strategy for generating registries of systematically tuned functional variants for each category—an activity that has been historically carried out using random mutagenesis and selection approaches. Importantly, they also reveal key technical and logistical considerations for devising ML-based projects. First, NA composition and overall library diversity should align with the prediction task. Probing underdetermined design spaces may require large, diverse libraries that prioritize coverage, while elements containing well-characterized sequence motifs can permit more targeted diversification. Model choice and feature selection are closely coupled with these considerations. While model classes that use biophysically consistent feature selection can offer mechanistic explanations of circuit behavior [55,62], higher

Figure 2


a
MODELS TRAINED ON PUBLIC DATASETS
Genetic element/ part category | Prediction goal | Training data source | Training diversity | Model class | Feature selection | Accuracy | Ref.
genomic sequence segments Zhou 2015 | predict chromatin profiles and gene expression activity for non-coding sequence variant in human cells | ChIP: TF binding, chrom. marks, Dnase HS | 9.3 x 10² NGS datasets | CNN (1D Conv) | one hot encoded sequence | AUROC = 0.8 - 0.95 | 49
human enhancers Avsec 2021 | predict chromatin profiles and expression activity for specific loci regulated by pluripotency factors | ChIP nexus chromatin marks (mulitple exps.) | 1.5 x 10⁵ regions (~1 kb) | CNN (ResNet) | one hot encoded sequence | AUPRC = 0.5 - 0.75 | 51
genomic sequences with tissue specific mutations Zhou 2018 | predict tissue-specific chromatin profiles and gene expression activity for sequence variants in human tissues | ChIP: TF binding, chrom. marks, Dnase HS | 2 x 10³ NGS datasets | CNN (1D conv) | one hot encoded sequence | AUROC = 0.81 | 52
human enhancers Chen 2022 | predict cell type-specific expression activity for enhancer sequences in human cells | ChIP: TF binding, chrom. marks, Dnase HS | 2.1 x 10⁶ NGS datasets, >1,300 cell types | CNN (ResNet w/ dil. conv.) | one hot encoded sequence | AUROC = 0.97 | 53
promoters, enhancers gen. sequence segments Avsec 2021 | predict chromatin profiles and gene expression activity across long length scales in a variety of organisms and cell types | ChIP: TF binding chrom. marks, CAGE, eQTLs, ATACseq, DNAse HS | 5.3 X 10³ human, 1.6 X 10² mouse NGS datasets >10³ cell types | Transformer | one hot encoded sequence | AUROC = 0.7 - 0.9 | 54
Chen 2022
CNN
INPUTS
genomic sequences
...atcgatgc...
one-hot encode
conv. layers
fully connected layers
OUTPUTS
TF binding, chrom marks, accessiblity
Avsec 2021
Transformer
INPUTS
genomic sequences
...atcgatgc...
one-hot encode
conv. layers
self attention layer
conv. layers
OUTPUTS
human
mouse
TF binding, chrom. marks, accessibility, activity
b
MODELS TRAINED ON MPRA DATASETS
bacterial promoters LaFleur 2022 | predict expression activity from promoter sequence in e. coli | in-vitro transcription and RNAseq | ~2.2 x 10⁴ promoter sequences (60 - 120 bp) | logistic regression | Calc. free energy from seq. motifs | r² = 0.80 | 55
toehold riboswitches Angenent-Mari 2020 | predict expression in presence/absence of inducer from sequence | flow-seq | ~9.1 X 10⁴ riboswitches (148 bp) | MLP | one hot encoded sequence | r² = 0.43 | 66
yeast promoters de Boer 2020 | predict expression from enhancer sequences in yeast | flow-seq | 1 X 10⁶ promoters (80 bp) | non-linear regression | TF binding probability from sequence | r² = 0.92 | 71
yeast promoters Kotopka 2020 | predict expression activity from promoter sequences in yeast | flow-seq | 1 X 10⁶ promoters (240 - 320 bp) | CNN (1D Conv) | one hot encoded sequence | r² = 0.79 | 56
cell type-specific CRE activity Gosai 2023 | predict transcript. activity in different cell types from enhancer CREs in human cells | RNA-seq | 7.8 X 10⁶ enhancers (200 bp) | CNN (2D Conv) | one hot encoded sequence | r² = 0.5 - 0.65 | 58
bacterial RBSs Hollerer 2020 | predict expression activity from RBS sequence in e. coli | DNA recorder assay for expression | 2.7 X 10⁶ RBS variants (18 bp) | CNN (ResNet) | one hot encoded sequence | r² = 0.93 | 67
human 5' UTRs Sample 2019 | predict transcript. activity from 5' UTR sequence | in-vitro transcription and polysome profiling by NGS | 2.8 X 10⁵ UTR sequences (50 bp) | CNN (2D Conv) | one hot encoded sequence | r² = 0.93 | 68
yeast 5' UTRs Cuperus 2017 | predict transcript. activity from 5' UTR sequence | growth assay and RNAseq | ~5 X 10⁵ (50 bp) | CNN (2D Conv) | one hot encoded sequence | r² = 0.62 | 60
human enhancers and GREs Sahu 2022 | predict expression activity from enhancer + promoter sequece in mammalian cells | STARR-seq | ~9 X 10⁸ (50 - 500 bp) | logistic regression and CNN | TF binding prob, one hot encoded seq | AUPRC = 0.62 | 57
yeast promoters Vaishnav 2022 | predict expression activity from enhancer sequence in yeast | flow-seq | 3 X 10⁷ (80 bp) | CNN and transformer | one hot encoded sequence | r² = 0.92 | 61
Vaishnav 2022
CNN + transformer
INPUTS
yeast enhancer sequence
...atcgatgc...
one-hot encode
conv. layers
self attention layer
LSTM layer
fully connected layers
OUTPUTS
expression activity, motif function
Angenent-Mari 2020
MLP
INPUTS
riboswitch sequences
cis-repressed
cis-activated
one-hot encode
fully connected layers
OUTPUTS
activ.
fold-change expression activity
rep.
c
MODELS TRAINED ON COMBINED DATASETS
bacterial promoters Wang 2020 | generatively design of novel functional bacterial promoters | dRNA seq data from nativee. coli genes | 1.4 X 10⁴ native e.coli promoters | GAN | one hot encoded sequence | r² = 0.85 | 63
cell type specific enhancers in flies de Almeida 2024 | predict cell type specific activity of enhancer sequences from pre-trained accessibility model | scATACseq & RNAseq data | scATAC seq data from 8 x10⁵ single cells; transfer with RNAseq | CNN and transfer | one hot encoded sequence | AUROC = 0.4 | 85
cell type specific enhancers in flies Taskiran 2024 | predict cell type specific accessibility for enhancer sequences in Drosophila | scATACseq & RNAseq data | scATAC seq data from 2.4 x 10⁵ single cells | CNN-RNN hybrid | one hot encoded sequence | AUROC = 0.89 | 64
cell type-specific enhancers in human cells Agarwal 2023 | predict cell type specific activity for enhancers from pre-trained models fine tuned with synthetic data | RNAseq from synthetic enhancers | 2 x10⁵ synthetic enhancers | CNN-Unet | one hot encoded sequence | r² = 0.52 | 84
cell type-specific enhancers in human cells DaSilva 2024 | predict cell type specific activity from GWAS data trained model | DNase HS data | 7.3 x 10² human samples | CNN-Unet | one hot encoded sequence | ps = 0.4-0.75 | 65
foundation model for prokaryotes Nguyen 2024 | predict sequences across a wide array of tasks in prokaryotes | all existing prokaryotic NGS data | 2 x 10⁷ prokaryotic genomes | LLM | one hot encoded sequence | ps = 0.41 | 70
Nguyen 2024
Striped Hyena
INPUTS
prokaryotic genomic sequence
...atcgatgc...
one-hot encode
Hyena
rot. attn.
OUTPUTS
sequence likelihood (DNA, RNA, protein seqs.)
Hyena
GLU
conv. dense
conv.
dense conv.
rot. attn.
Current Opinion in Biomedical Engineering

complexity models can maximize prediction accuracy at the cost of mechanism interpretability [57]. Linear and logistic regression models, which directly fit coefficients to input features, are examples of the former. NNs, which excel at extracting underlying non-linear sequence patterns, have emerged as a leading choice for instances of the latter. It is worth noting that LM architectures, which leverage more complex feature embeddings, show performance advantages for learning interactions that are distal across sequence space. For protein parts, models could incorporate features derived from 3D protein structures by learning graphical embeddings of structures with graph convolutional neural networks [71]. Finally, since HT data acquisition in biological systems is inherently noisy, model accuracy may be challenging to assess using standard data splits due to test sets having a high degree of measurement error. It is therefore beneficial to obtain smaller, high quality test sets acquired via gold standard experimental approaches to validate trained models [72].

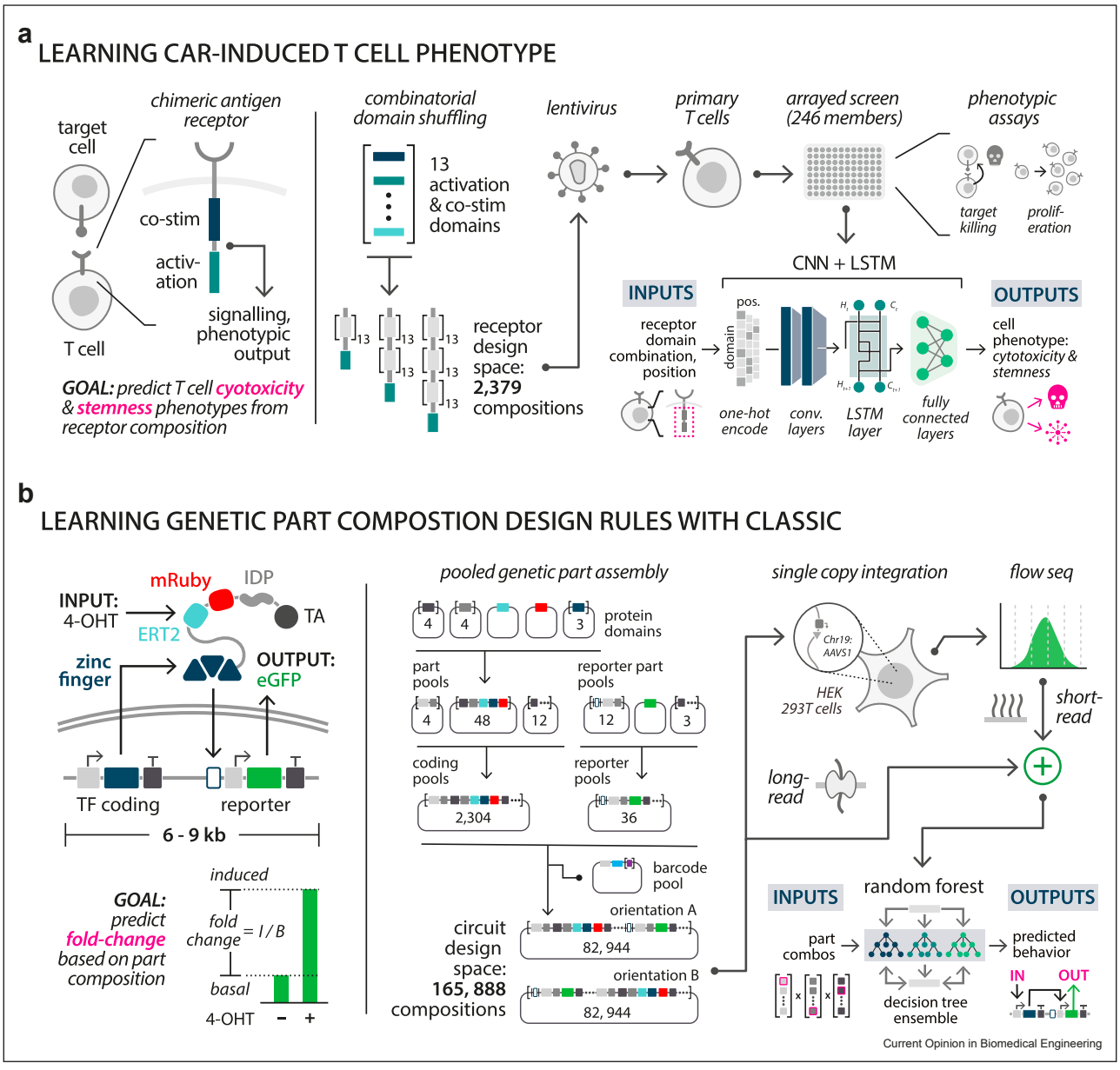
Composition-to-function models for predictive circuit design

While the observation of modularity in native biological systems [73] originally inspired the idealization of circuit design frameworks as consisting of fully interoperable sets of genetic parts, few fully composable platforms have been developed to date [74] and most circuit engineering schemes still must contend with non-linear part composition dependencies [75]. Therefore, beyond using sequence-to-function models to systematically tune genetic part function, developing *composition-to-function* models that predictively combine parts into functional circuits represents perhaps the most tantalizing potential application of ML to genetic design. However, few examples have been reported to date, in part due to technical challenges with building and measuring circuits in high throughput. Using NGS to profile constructs that are long enough to encode compositions comprising multiple parts (>1 kb) is limited by the Illumina-based read length limit of ~500bp. To address this, indexed multiplexing has emerged as a method by which short barcode sequences can be appended to the construct to facilitate NGS readout by short PCR amplicon [76,77]. However, because it requires arrayed cloning to unambiguously associate barcodes with specific designs, indexed multiplexing precludes pooled assembly and therefore limits library diversity compared to the part libraries described in the previous section.

In a pioneering example of using ML to achieve predictive composition-to-function insight, Daniels et al. explored how combinations of human receptor activation and co-stimulatory domains work together to determine chimeric antigen receptor (CAR) function in T cells [78] (Figure 3a). Their library consisted of a set of 13 domains diversified across 3 positions to yield a design space of ~2,400 receptors. The library was virally transduced into primary T cells and assayed for stemness and cytotoxicity, two phenotypic traits that correlate with tumor clearance during CAR T therapy. The data were used to train a NN model consisting of convolutional layers combined with an LSTM layer, an

Highlights from recent literature featuring sequence-to-function ML models. (**a**) Publicly available NGS data from HT functional assays (e.g., ChIP-seq, ATAC-Seq, and DNAse HS) were used to train ML models to predict expression activity or classify genetic elements (e.g., presence/absence of enhancers) based on NA sequence feature inputs (model inputs and outputs are colored pink in illustrations). CNNs are frequently used because of their efficiency in learning sequence motif patterns and accurately modeling non-linear regulatory effects. Utilization of larger datasets during training generally correlates with model complexity. These models perform classification tasks so area under precision–recall curve (AUPRC) or receiver–operator curve (AUROC) are used as accuracy metrics. (bottom left) Chen et al. (2022) developed a CNN model to predict TF binding, epigenetic marks, and chromatin accessibility for genomic sequence inputs. A common architecture was used, comprising one-hot encoding of sequence feature inputs, convolutional layers (blue), and fully connected layers (green). (bottom right) Avsec et al. (2021) combined CNN and transformer architectures to predict multi-modal epigenetic and expression activity across large sequence blocks (~196 kb) in numerous mouse and human cell types. The model consists of convolutional layers, a self-attention transformer layer (teal) featuring multi-head attention blocks with relative positional encodings, and a convolution with organism-specific heads to predict output genomic tracks. CNN, convolutional neural network; (**b**) Studies that use data from MPRA (massively parallel reporter assay) experiments for model training use a range of measurement approaches, including flow-seq to sort cells based on fluorescent reporter intensity followed by NGS to quantify variants expression; RNA-seq to quantify the number of unique RNA molecules detected for each barcode; DNA recorders that use recombinase flipping frequency to assess expression strength; Ribo-seq to sequence ribosome bound fragments; fitness-based assays that link output to growth rate, and; STARR-seq: self-transcribing enhancers placed downstream of a promoter to enable quantitation of variant expression. NN models are often used, but can be combined with interpretable regression models that featurize biophysical priors. Library size is influenced by regulatory complexity of genetic elements; models for prokaryotic genetic elements deliver accurate predictions on smaller datasets compared to that of eukaryotes. These tasks are regression-based so model accuracy is assessed by Pearson r^2, or Spearman ρ. (bottom left) A model developed by Vaishnav et al. (2022) that predicted reporter expression from enhancer sequence consisted of convolution layers, a transformer self-attention layer, a long-short term memory layer (LSTM), followed by fully connected layers. (bottom right) Angenent-Mari (2020) developed a simple MLP model made up of three fully connected layers that predicted expression differences (fold change) between cis-repressed and cis-activated RNA switches based on toehold sequence. MLP, multi-layer perceptron; LSTM, long short-term memory. (**c**) Studies featuring models trained on combined natural and synthetic datasets. Included are generative models that can design sequences with specified phenotypic behavior, and foundational models that can predict a wide range of tasks and applications. Model class choice depends heavily on task, size of dataset, and number of output modalities. (bottom) Nguyen et al., (2024) used a novel LLM architecture called striped hyena to train on every available prokaryotic genome sequence. The model predicts the likelihood that a DNA, RNA, or protein sequence can fulfill a queried function (e.g., what is the fitness effect of a DNA sequence?). GLU, Gated linear unit; dense: fully connected MLP block, conv.: convolutional block, rot. attn., Rotary attention layer, which uses positional encoding instead of static encoding used by self-attention.

Figure 3



Recent examples developing composition-to-function ML models. (**a**) Using ML to learn phenotypic behavior of engineered CAR T cells. Signaling output of a CAR upon encounter with cells bearing a target antigen leads to activation of signaling pathways that determine T cell phenotype, including cytotoxicity and proliferative capacity. To develop an ML model that predicts induced phenotype from CAR design, a library was constructed that consisted of 13 signaling domains selected from literature, arrayed across one, two, or three-position receptors (total design space of 2,379). T cell killing capacity and stemness were measured for 246 designs. Receptor composition features for this dataset were one-hot encoded as a 13 × 3 matrix, and used to train a model consisting of convolution layers, an LSTM layer, and a fully connected layer. After validating generalizability and prediction accuracy (MAE of 0.11 and r^2 of 0.71), the trained model was used to predict output labels for all 2,379 combinations in the design space, and validated by testing promising designs. The model was also used to design a 4-part co-stim CAR, that boosted the stemness without loss of cytotoxicity. Co-stim, co-stimulatory domain; conv., convolutional; LSTM, long short-term memory. (**b**) Machine learning for predicting gene circuit behavior based on part-composition. CLASSIC uses a modular cloning scheme for combinatorial assembly of entire design spaces. The platform assembles library members with semi-degenerate barcode pools and uses a combination of long- and short-read NGS to characterize phenotypic responses in bulk. (left) To validate the platform, an inducible zinc finger-TF based transcriptional circuit library was constructed consisting of two genes: one encoding the TF and the other a GFP reporter. The circuit is induced by addition of 4-OHT, whereupon the TF undergoes translocation to the nucleus and activates reporter transcription. A part composition library was designed by combinatorially varying parts across 10 categories, producing a total design space of 165,888 variants. (right) The library was assembled and integrated into HEK293T cells at single copy (AAVS1 locus). Flow-seq was performed to assay the induced and uninduced circuit expression for the library. Two independent gradient-boosted random forest models were trained on the data with categorical parts as input features. The model was used to predict basal and induced expression profiles for all library members, generating an accurate and complete mapping of the design space (r^2 of 0.83 and 0.87 for basal and induced expression respectively). Feature importance scores from the

architecture that could predict phenotype for novel domain combinations. In addition to demonstrating that ML models can learn genetic design rules for complex cellular behavior, this manuscript is perhaps the first to demonstrate the potential for applying ML to engineer cells with novel therapeutic function [79].

In a second example, a recent preprint reported a generalizable platform for building and testing large ($>10^5$) circuit libraries comprising diverse genetic part compositions [80] (Figure 3b). The platform, known as CLASSIC (combining long and short-range sequencing to interrogate genetic complexity), enables implementation of indexed multiplexing through a multi-stage pooled assembly scheme that incorporates barcodes in the final step, dramatically increasing scale and decreasing cost for part-diversified libraries. Pooled circuit compositions are indexed to their corresponding barcode via Oxford Nanopore long-read NGS, creating a lookup table that assigns circuit compositions to phenotype in subsequent short-read experiments. The authors use CLASSIC to explore a design space for a two-gene (6–9 kb), small molecule-inducible transcriptional circuit in HEK293T cells, with the goal of identifying compositions that maximize fold-change induction. Ten part categories were diversified within several circuit architectures to yield >165k possible compositions—the largest circuit library constructed and tested to date. A circuit activity dataset covering ~70% of the total design space was obtained by flow-seq and used to train a random forest model to predict expression output based on part composition. The model accurately predicted behavior of unmeasured part compositions and produced a highly accurate, measurement error-corrected map of the entire design space, allowing identification of fold change-optimized compositions and revealing circuit design rules, many of which were highly non-intuitive.

Despite interrogating relatively small design spaces that are learnable via low complexity models (~3k and ~7k parameters, respectively), the above-described examples offer an early glimpse of how ML can overcome enduring part composability issues in synthetic biology and more efficiently guide circuit tuning. While CLASSIC facilitates quantification of design spaces at unprecedented scales, there is an acute need for HT approaches that further scale circuit construction and data acquisition to incorporate even larger pools of part sequence variants. Data from such multi-scale design spaces could be featurized with both part sequence and composition, though it remains to be seen what model classes will prove to be the most useful for learning these spaces. Solutions could incorporate multivariate regression models that combine outputs from independent sequence- and composition-trained convolutional layers, or LM layers that capture sequence-level part interaction grammar across a circuit locus.

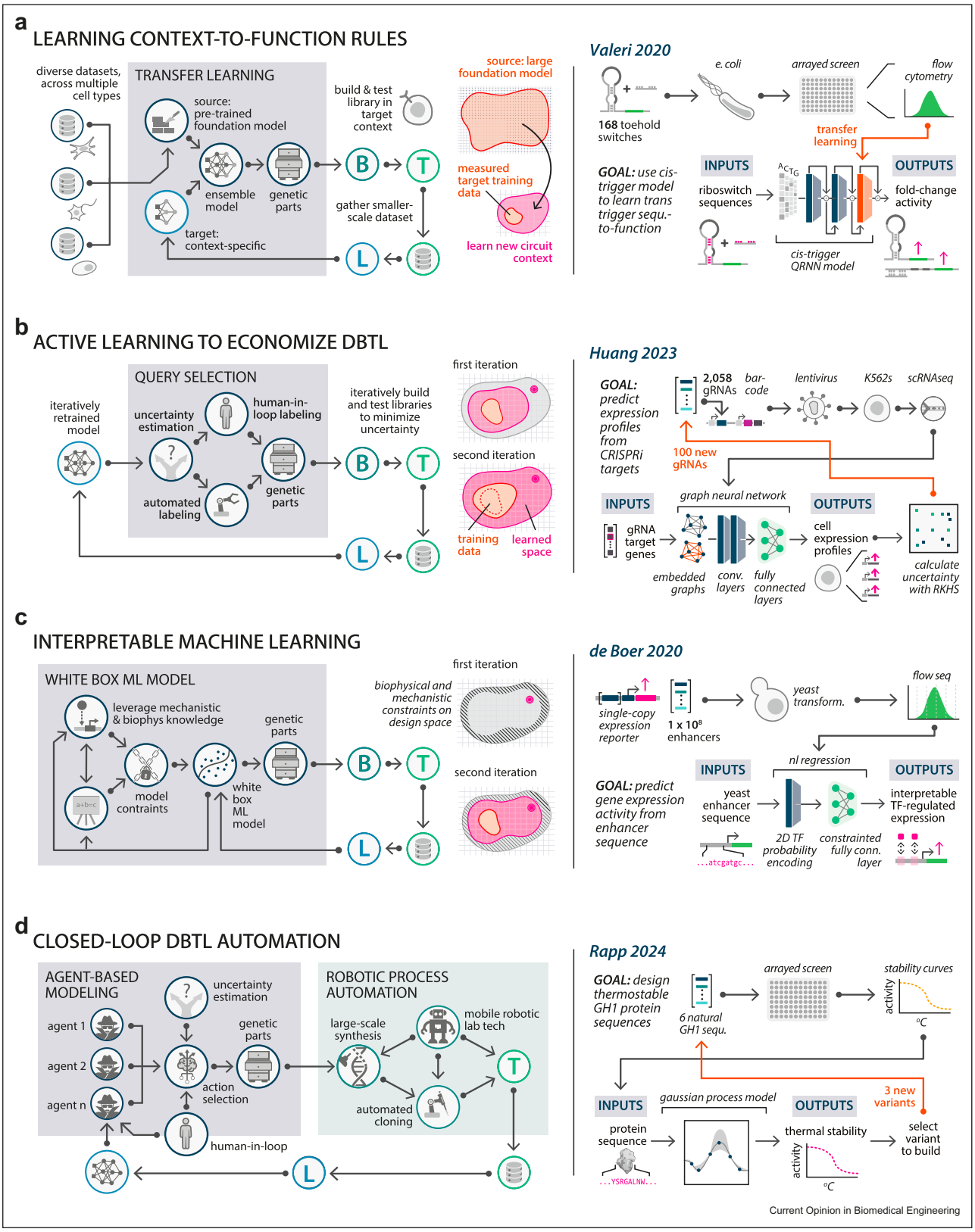
Context-to-function models and diversification of ML-driven DBTL cycles

The studies described in the previous sections developed models that learn sequence and composition design spaces from datasets that were each collected under specific circumstances. We anticipate that the next phase of ML in synthetic biology will produce context-to-function models that can predict how parts and circuits behave when scaled across diverse experimental or host cell contexts. For microorganisms, these models could be used to predict circuit behavior across culture conditions or in different species [81], while in mammalian cells they could predict behavior in different cell types or genomic loci. A key challenge across both phyla will be accounting for interactions between genetic circuits and host machinery [82]. The resource burden associated with circuit expression can impinge upon cellular growth rate, creating feedback between cell health and circuit output that would need to be addressed through composition-to-function modeling. A key to developing these models will be establishing experimental approaches for making HT measurements across contexts, and developing host-aware models that account for context and resource sharing effects. For example, data from integration site analysis [83] could be used to gather circuit behavior data from many genomic loci in mammalian cells, facilitating training of models that featurize epigenetic marks or local 3D chromatin structure and identify loci that permit optimized circuit behavior.

One potentially enabling strategy for developing context-to-function models will be **transfer learning** (TL) (Figure 4a, left), a technique whereby a model trained for one task can be reused, either in part or in whole, as the starting point for a second task [84]. For example, once a "source" model is trained on circuit design features in one species or cell type (e.g., HEK293 cells), it could be extended by training a smaller "target" model in a different context (e.g., iPSCs), potentially with a far smaller dataset. The source model could then be "fine-tuned" by appending the target model to the source model, or using it to replace the source model's output layers. One recently reported example of TL that built off the work of Angenent-Mari (see Figure 2b) involved learning sequence-to-function rules for trans-activated RNA

trained model, coupled with statistical inference methods from the model-completed design space were used to analyze design principles for the circuit class, and identify variants with optimized behavior profiles such as high fold-change in induced gene expression. 4-OHT, 4-hydroxytomoxifen; ERT2, tamoxifen response element; IDP, intrinsically disordered protein.

Figure 4



Future applications of ML to learn design-to-function rules for synthetic gene circuits. (**a**) By incorporating TL into the synthetic circuit DBTL cycle, previously trained models can be leveraged to more efficiently learn how to design circuits for new contexts. (left) Large foundation models (e.g.,

toehold switch function by appending layers from a model trained on a small set of newly-obtained data to a larger model trained on previously collected cis-regulated data (Figure 4a, right) [85]. A second enabling strategy is to use TL in conjunction with foundation models such as Enformer, its successor Borzoi [86], or Evo. In an early example of this strategy, Agarwal and colleagues [87] constructed libraries of synthetic enhancer sequences and delivered them via lentivirus into 3 different cell-types. They found that predictions made by Enformer out-of-the-box (without fine-tuning) outperformed a model trained on their data alone. One challenge for utilizing foundation models is their requirement for significant computing power, since they are parameter-intensive and utilizing them for diverse tasks may require hyperparameter optimization. Furthermore, because foundation models are trained primarily on native data, they may lack strong predictive power for synthetic sequence or circuit design tasks. However, given sufficient resources, TL workflows could be developed in which foundation models fine-tuned with modestly-sized libraries achieve high accuracy for tasks such as predicting circuit function in a new bacterial species, or in various genomic loci within a mammalian cell type of interest.

These use cases highlight the potential utility of performing successive ML-driven DBTL cycle iterations for sets of related tasks, whether the goal is to iteratively improve a model's predictive power across a design space, or to navigate to new regions of interest by fine-tuning pre-trained models to recognize new output labels [88]. Here, a relevant concept is active learning (AL), which encompasses a well-established set of ML techniques in which models iteratively select the most informative data from which to learn in subsequent cycles, with the goal of maximizing generalization accuracy (Figure 4b). This approach improves model performance by focusing on obtaining training data that is maximally impactful, rather than sampling a design space at random. AL has been widely used for speech recognition [89], text analysis [90], and has recently seen use in biology to guide perturbation of transcriptional networks [91] and navigate human enhancer sequence design space [92,93]. Incorporating AL algorithms into a synthetic circuit DBTL cycle could involve initial evaluation of a model to identify poorly predicted regions of a design space, followed by construction of sequence variants that sample the regions while simultaneously optimizing for external constraints like DNA synthesis and circuit assembly costs, or assay throughput. One key consideration when deploying AL for synthetic biology is the exploration-exploitation trade-off. Over-exploitation of insufficiently diversified data can cause a model to stall at local minimum, while over-exploration can limit model improvement due to sparse selection of data. Striking a proper balance is crucial for efficient AL, and could be challenging for design spaces that are too expansive or underdetermined to explore by random sampling at each cycle.

Because many ML model classes (e.g., deep NNs and LMs) are "black box" in nature and trade interpretability for predictive accuracy, understanding the basis for successful circuit designs in biophysical and mechanistic terms—an inherent aspect of the historic DBTL cycle—can potentially be lost in an ML-driven process (Figure 4c, left). Current approaches address this by either selecting model classes that strike a balance

Enformer, Evo) pre-trained on diverse datasets (e.g., expression data from multiple cell types) can serve as TL source models that incorporate new output layers trained on data gathered in new target contexts (e.g., an untested host cell type), yielding ensemble models that can predict circuit function in the new context with high accuracy. (right) In Valeri et al. (2020), TL was used to learn sequence rules for toehold riboswitch trans-activation by leveraging a cis-activation trained quasi-recurrent NN (QRNN) model as a source. TL was performed by freezing all but the final layers, and allowing parameters to be trained with a set of just 168 measured switches. (**b**) Using AL to enhance ML-driven circuit design. (left) In query selection, an AL process probes a model trained during an initial cycle for design space regions of high uncertainty, and then directs design of circuits in a subsequent cycle to maximize expected information gain. This can either be carried out in an automated fashion, or though choices made by a user-in-the-loop. This speeds a model's ability to learn by requiring less data or fewer iterations to converge on a target behavior. (right) AL was used to direct iterative selection of CRISPRi perturbations to gather training data for a graph neural network that modeled transcriptional network regulation in a human cell line. In each round, the model's errors for a holdout set of perturbations were computed, while incorporating existing priors such as published scRNA-seq datasets. A set of gRNAs was then selected for another perturbation cycle to minimize the model's loss on the held-out test set, enabling rapid convergence of the model compared to an unbiased gRNA sampling. (**c**) Interpretable ML for developing mechanistic understanding of circuit function. (left) A ML model is constrained by encoding features in an interpretable manner, only enabling interactions between terms that are biophysically consistent. This "white-box" approach enables the learned weights to be directly translatable to known features such as binding interactions between TFs and cis-regulatory motifs. (right) An example of an interpretable ML model trained on yeast flow-seq data and used to predict expression from enhancer sequences. The model encodes sequence inputs as a 2D matrix representing binding probability for all TFs in yeast. The sum of these probabilities is combined in a second layer with learned TF accessibility parameters, resulting in a model that relates quantitative expression level to TF occupancy. (**d**) Automating ML-driven DBTL cycles. (left) agent-based ML models deploy multiple independent model instances that query a design space for sequences or compositions that meet design optimization requirements. Model uncertainty estimates, combined with a set of human defined rules, lead to action selection, which selects diverse circuit designs to build and test. The diversity required for this approach could be enabled by robotic process automation, where build steps utilize high-throughput cloning and acquisition of data, which is fed back to the model for the next cycle of agent-based action selection. (right) In one example, agent-based ML was used to optimize protein thermostability. Arrayed measurements from 6 seed sequences of a family 1 glycoside hydrolase (GH1) were fed into a gaussian process (GP) model. Four independent agents predicted thermostability boosting mutations. Three sequences were selected and built in each round for each agent, and the process was repeated for a set number of rounds of optimization, resulting in GH1 variants that were significantly more thermostable than WT.

between predictability and interpretability, or by interrogating trained models to evaluate their choices. An outstanding recent example of the former approach is a precursor study to Vaishnav et al. (see Figure 2b) performed by the same group, where flow-seq data from a yeast enhancer library (~100M 80 bp sequences) was used to train a biophysically consistent model architecture for TF binding [59] (Figure 4c, right). The model accurately identified features of TF motif grammar driving transcription, and could predict expression activity with exceptional accuracy. Other examples include application of sparse linear models such as LASSO or RIDGE regularization, which minimize model parameter magnitude to improve interpretability while maintaining predictive performance [94]. One example of the latter explainable AI (xAI) approaches are SHAP (Shapley Additive exPlanations) values, which can estimate feature contributions to model output and infer their importance [94]. Other examples include saliency maps, which highlight the most influential/important input features for a model's prediction [95], and LIME (Local Interpretable Model-agnostic Explanations), which approximates model behavior locally using more interpretable model classes [96]. These approaches have seen widespread use for medical image analysis [97], audio processing [98], gene regulatory network inference [99], and in biological sequence-to-function tasks [66,86]. However, tradeoffs between prediction accuracy and interpretability must be balanced when using these approaches, which should only be implemented after fully validating model accuracy. Further, when computationally feasible, training a "ladder" of multiple models that range from more interpretable to more predictive may help to assess the extent to which the data can be explained based on the current feature selection. The incorporation of interpretable ML approaches into the circuit engineering process might involve first training a model to predictively map design space, and then using xAI techniques to understand how choices made by the ML model correspond to known mechanisms. This could facilitate biophysical understanding of genetic parts and circuits for which mechanistic modelling frameworks already exist, or could potentially be used to uncover new, previously unknown mechanistic features of a system.

Conclusions

Given the dizzying pace at which ML/AI is currently advancing, there will be an abundance of computational tools that can be applied to ML-driven synthetic biology as the partnership between the two fields advances. Therefore, in addition to working together with data scientists to establish best-practices for model choice, training, and validation for various experimental domains, a central focus for synthetic biologists in the coming years should be on developing HT methods to obtain low noise training datasets through efficient and cost-effective means. Toward this goal, a combination of DNA assembly and NGS/3GS innovations will continue to play a role, and their adoption by synthetic biology labs should be encouraged. Additionally, advances in robotics-assisted lab automation could facilitate scale-up of the entire DBTL cycle (Figure 4d) [100]. Closed-loop pipelines have already been developed for protein engineering [101], and the opportunity exists to apply agent-based and reinforcement ML approaches that employ human-in-loop or fully autonomous algorithm ensembles to map vast, high-dimensional design spaces [101,102]. Groups lacking capabilities to perform HT work could be enabled by community efforts aimed at developing accessible fine-tuning and TL approaches that leverage established foundation models, thereby enabling investigators to take on problems that are data-sparse.

While the three-tiered hierarchy highlighted in this review—sequence-to-function, composition-to-function, and context-to-function—provides an initial framework for categorizing learning goals, it is important to recognize that tasks taken on by circuit engineers will rarely fall under just a single category. Parts with defined sequences can show varying function depending on their position within a circuit composition, while different circuit designs may display performance that varies within a host cell context. Therefore, it will be imperative to develop modelling frameworks that connect all three tiers into "ensemble" models that can be fine-tuned to incorporate relevant source models or datasets. These models could make it feasible to design circuits *de novo* by outputting candidate circuit compositions based on user-defined functional "prompts". Such generative design tools could be used to enable a variety of end users across diverse technology domains to productively explore complex, multi-modal design spaces that address key challenges in areas like renewable biomanufacturing, disease- and patient-specific programming of therapeutic cells and gene therapies, and bioremediation using diverse chassis organisms equipped with customized biosensor function.

Author contributions

All authors contributed to the conceptualization, organization, and writing of the manuscript.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We acknowledge colleagues and collaborators who helped us develop the ideas in this review through correspondence and discussion, including Pankaj Mehta, Jason Yang, Oleg Igoshin, Todd Treangen, and Jeffrey Tabor, as well as other members of the Bashor lab. This work was supported by

grants from NIH R01 EB029483, NIH R01 EB032272, ONR N00014-21-1-4006, and funding from the Robert J. Kleberg Jr. and Helen C. Kleberg Foundation.

References

Papers of particular interest, published within the period of review, have been highlighted as:

* of special interest
** of outstanding interest

1. Cameron DE, Bashor CJ, Collins JJ: **A brief history of synthetic biology**. *Nat Rev Microbiol* 2014, **12**:381–390.
2. Endy D: **Foundations for engineering biology**. *Nature* 2005, **438**:449–453.
3. Meng F, Ellis T: **The second decade of synthetic biology: 2010-2020**. *Nat Commun* 2020, **11**:5174.
4. Nielsen AA, Segall-Shapiro TH, Voigt CA: **Advances in genetic circuit design: novel biochemistries, deep part mining, and precision gene expression**. *Curr Opin Chem Biol* 2013, **17**: 878–892.
5. English MA, Gayet RV, Collins JJ: **Designing biological circuits: synthetic biology within the operon model and beyond**. *Annu Rev Biochem* 2021, **90**:221–244.
6. Brink KR, Hunt MG, Mu AM, Groszman K, Hoang KV, Lorch KP, Pogostin BH, Gunn JS, Tabor JJ, An E: **Coli display method for characterization of peptide-sensor kinase interactions**. *Nat Chem Biol* 2023, **19**:451–459.
7. Alnahhas RN, Sadeghpour M, Chen Y, Frey AA, Ott W, Josic K, Bennett MR: **Majority sensing in synthetic microbial consortia**. *Nat Commun* 2020, **11**:3659.
8. Bashor CJ, Patel N, Choubey S, Beyzavi A, Kondev J, Collins JJ, Khalil AS: **Complex signal processing in synthetic gene circuits using cooperative regulatory assemblies**. *Science* 2019, **364**:593–597.
9. Zhu R, Del Rio-Salgado JM, Garcia-Ojalvo J, Elowitz MB: **Synthetic multistability in mammalian cells**. *Science* 2022, **375**, eabg9765.
10. Li HS, Israni DV, Gagnon KA, Gan KA, Raymond MH, Sander JD, Roybal KT, Joung JK, Wong WW, Khalil AS: **Multidimensional control of therapeutic human cell function with synthetic gene circuits**. *Science* 2022, **378**:1227–1234.
11. Prochazka L, Michaels YS, Lau C, Jones RD, Siu M, Yin T, Wu D, Jang E, Vazquez-Cantu M, Gilbert PM, *et al.*: **Synthetic gene circuits for cell state detection and protein tuning in human pluripotent stem cells**. *Mol Syst Biol* 2022, **18**, e10886.
12. Guye P, Ebrahimkhani MR, Kipniss N, Velazquez JJ, Schoenfeld E, Kiani S, Griffith LG, Weiss R: **Genetically engineering self-organization of human pluripotent stem cells into a liver bud-like tissue using Gata6**. *Nat Commun* 2016, **7**, 10243.
13. Chen Y, Kim JK, Hirning AJ, Josic K, Bennett MR: **SYNTHETIC BIOLOGY. Emergent genetic oscillations in a synthetic microbial consortium**. *Science* 2015, **349**:986–989.
14. Gardner TS, Cantor CR, Collins JJ: **Construction of a genetic toggle switch in Escherichia coli**. *Nature* 2000, **403**:339–342.
15. Elowitz MB, Leibler S: **A synthetic oscillatory network of transcriptional regulators**. *Nature* 2000, **403**:335–338.
16. Nielsen J, Keasling JD: **Engineering cellular metabolism**. *Cell* 2016, **164**:1185–1197.
17. Choi KR, Jang WD, Yang D, Cho JS, Park D, Lee SY: **Systems metabolic engineering strategies: integrating systems and synthetic biology with metabolic engineering**. *Trends Biotechnol* 2019, **37**:817–837.
18. Voigt CA: **Synthetic biology 2020-2030: six commercially-available products that are changing our world**. *Nat Commun* 2020, **11**:6379.
19. Rylott EL, Bruce NC: **How synthetic biology can help bioremediation**. *Curr Opin Chem Biol* 2020, **58**:86–95.
20. Bashor CJ, Hilton IB, Bandukwala H, Smith DM, Veiseh O: **Engineering the next generation of cell-based therapeutics**. *Nat Rev Drug Discov* 2022, **21**:655–675.
21. Roybal KT, Lim WA: **Synthetic immunology: hacking immune cells to expand their therapeutic capabilities**. *Annu Rev Immunol* 2017, **35**:229–253.
22. Kitano S, Lin C, Foo JL, Chang MW: **Synthetic biology: learning the way toward high-precision biological design**. *PLoS Biol* 2023, **21**, e3002116.
23. Son HI, Weiss A, You L: **Design patterns for engineering genetic stability**. *Curr Opin Biomed Eng* 2021, **19**.
24. Karr JR, Sanghvi JC, Macklin DN, Gutschow MV, Jacobs JM, Bolival Jr B, Assad-Garcia N, Glass JI, Covert MW: **A whole-cell computational model predicts phenotype from genotype**. *Cell* 2012, **150**:389–401.
25. Bottaro S, Lindorff-Larsen K: **Biophysical experiments and biomolecular simulations: a perfect match?** *Science* 2018, **361**:355–360.
26. Muller IE, Rubens JR, Jun T, Graham D, Xavier R, Lu TK: **Gene networks that compensate for crosstalk with crosstalk**. *Nat Commun* 2019, **10**:4028.
27. Prindle A, Selimkhanov J, Li H, Razinkov I, Tsimring LS, Hasty J: **Rapid and tunable post-translational coupling of genetic circuits**. *Nature* 2014, **508**:387–391.
28. Mehta P, Wang CH, Day AGR, Richardson C, Bukov M, Fisher CK, Schwab DJ: **A high-bias, low-variance introduction to Machine Learning for physicists**. *Phys Rep* 2019, **810**:1–124.
29. Bennett NR, Coventry B, Goreshnik I, Huang B, Allen A, Vafeados D, Peng YP, Dauparas J, Baek M, Stewart L, *et al.*: **Improving de novo protein binder design with deep learning**. *Nat Commun* 2023, **14**:2625.
30. Silver D, Schrittwieser J, Simonyan K, Antonoglou I, Huang A, Guez A, Hubert T, Baker L, Lai M, Bolton A, *et al.*: **Mastering the game of Go without human knowledge**. *Nature* 2017, **550**: 354–359.
31. Oord Avd, Dieleman S, Zen H, Simonyan K, Vinyals O, Graves A, Kalchbrenner N, Senior A, Kavukcuoglu K: *Wavenet: a generative model for raw audio*. arXiv preprint; 2016.
32. Libbrecht MW, Noble WS: **Machine learning applications in genetics and genomics**. *Nat Rev Genet* 2015, **16**:321–332.
33. Camacho DM, Collins KM, Powers RK, Costello JC, Collins JJ: **Next-generation machine learning for biological networks**. *Cell* 2018, **173**:1581–1592.
34. Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, Tunyasuvunakool K, Bates R, Zidek A, Potapenko A, *et al.*: **Highly accurate protein structure prediction with AlphaFold**. *Nature* 2021, **596**:583–589.
35. Baek M, DiMaio F, Anishchenko I, Dauparas J, Ovchinnikov S, Lee GR, Wang J, Cong Q, Kinch LN, Schaeffer RD, *et al.*: **Accurate prediction of protein structures and interactions using a three-track neural network**. *Science* 2021, **373**: 871–876.
36. Notin P, Rollins N, Gal Y, Sander C, Marks D: **Machine learning for functional protein design**. *Nat Biotechnol* 2024, **42**: 216–228.
37. Khakzad H, Igashov I, Schneuing A, Goverde C, Bronstein M, Correia B: **A new age in protein design empowered by deep learning**. *Cell Syst* 2023, **14**:925–939.

38. Chu AE, Lu T, Huang PS: **Sparks of function by de novo protein design**. *Nat Biotechnol* 2024, **42**:203–215.

39. Bartley BA, Kim K, Medley JK, Sauro HM: **Synthetic biology: engineering living systems from biophysical principles**. *Biophys J* 2017, **112**:1050–1058.

40. Goodwin S, McPherson JD, McCombie WR: **Coming of age: ten years of next-generation sequencing technologies**. *Nat Rev Genet* 2016, **17**:333–351.

41. Neumayr C, Pagani M, Stark A, Arnold CD: **STARR-Seq and UMI-STARR-seq: assessing enhancer activities for genome-wide-, high-, and low-complexity candidate libraries**. *Curr Protoc Mol Biol* 2019, **128**, e105.

42. Arnold CD, Gerlach D, Stelzer C, Boryn LM, Rath M, Stark A: **Genome-wide quantitative enhancer activity maps identified by STARR-seq**. *Science* 2013, **339**:1074–1077.

43. Bergman DT, Jones TR, Liu V, Ray J, Jagoda E, Siraj L, Kang HY, Nasser J, Kane M, Rios A, *et al.*: **Compatibility rules of human enhancer and promoter sequences**. *Nature* 2022, **607**:176–184.

44. Seimetz J, Arif W, Bangru S, Hernaez M, Kalsotra A: **Cell-type specific polysome profiling from mammalian tissues**. *Methods* 2019, **155**:131–139.

45. Ingolia NT, Ghaemmaghami S, Newman JR, Weissman JS: **Genome-wide analysis in vivo of translation with nucleotide resolution using ribosome profiling**. *Science* 2009, **324**:218–223.

46. Consortium EP, Moore JE, Purcaro MJ, Pratt HE, Epstein CB, Shoresh N, Adrian J, Kawli T, Davis CA, Dobin A, *et al.*: **Expanded encyclopaedias of DNA elements in the human and mouse genomes**. *Nature* 2020, **583**:699–710.

47. Mansisidor AR, Risca VI: **Chromatin accessibility: methods, mechanisms, and biological insights**. *Nucleus* 2022, **13**:236–276.

48. Buenrostro JD, Giresi PG, Zaba LC, Chang HY, Greenleaf WJ: **Transposition of native chromatin for fast and sensitive epigenomic profiling of open chromatin, DNA-binding proteins and nucleosome position**. *Nat Methods* 2013, **10**:1213–1218.

49. Zhou J, Troyanskaya OG: **Predicting effects of noncoding variants with deep learning-based sequence model**. *Nat Methods* 2015, **12**:931–934.

50. Kelley DR, Snoek J, Rinn JL: **Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks**. *Genome Res* 2016, **26**:990–999.

51. Avsec Z, Weilert M, Shrikumar A, Krueger S, Alexandari A, Dalal K, Fropf R, McAnany C, Gagneur J, Kundaje A, *et al.*: **Base-resolution models of transcription-factor binding reveal soft motif syntax**. *Nat Genet* 2021, **53**:354–366.

52. Zhou J, Theesfeld CL, Yao K, Chen KM, Wong AK, Troyanskaya OG: **Deep learning sequence-based ab initio prediction of variant effects on expression and disease risk**. *Nat Genet* 2018, **50**:1171–1179.

53. * Chen KM, Wong AK, Troyanskaya OG, Zhou J: **A sequence-based global map of regulatory activity for deciphering human genetics**. *Nat Genet* 2022, **54**:940–949.

In an standout example of training ML models via large numbers of chromatin profiles for multi-modal output predictions, the authors show that Sei, their ML algorithm, can integrate chromatin marks and TF binding data to identify enhancers and predict effects of mutations in regulatory non-coding regions of the genome

54. Avsec Z, Agarwal V, Visentin D, Ledsam JR, Grabska-Barwinska A, Taylor KR, Assael Y, Jumper J, Kohli P, Kelley DR: **Effective gene expression prediction from sequence by integrating long-range interactions**. *Nat Methods* 2021, **18**:1196–1203.

55. * LaFleur TL, Hossain A, Salis HM: **Automated model-predictive design of synthetic promoters to control transcriptional profiles in bacteria**. *Nat Commun* 2022, **13**:5159.

Demonstrates application of interpretable machine learning for understanding regulatory behavior of multiple promoter elements and map non-linear interactions between these elements. The authors use a logistic regression model that can predict transcription initiation rate from a given promoter sequence. The model excels at explaining individual contributions of sequence based motifs to initiation rate.

56. Kotopka BJ, Smolke CD: **Model-driven generation of artificial yeast promoters**. *Nat Commun* 2020, **11**:2113.

57. Sahu B, Hartonen T, Pihlajamaa P, Wei B, Dave K, Zhu F, Kaasinen E, Lidschreiber K, Lidschreiber M, Daub CO, *et al.*: **Sequence determinants of human gene regulatory elements**. *Nat Genet* 2022, **54**:283–294.

58. Gosai SJ, Castro RI, Fuentes N, Butts JC, Kales S, Noche RR, Mouri K, Sabeti PC, Reilly SK, Tewhey R: **Machine-guided design of synthetic cell type-specific cis-regulatory elements**. *bioRxiv* 2023.

59. Bogard N, Linder J, Rosenberg AB, Seelig G: **A deep neural network for predicting and engineering alternative polyadenylation**. *Cell* 2019, **178**:91–106 e123.

60. Cuperus JT, Groves B, Kuchina A, Rosenberg AB, Jojic N, Fields S, Seelig G: **Deep learning of the regulatory grammar of yeast 5' untranslated regions from 500,000 random sequences**. *Genome Res* 2017, **27**:2015–2024.

61. ** Vaishnav ED, de Boer CG, Molinet J, Yassour M, Fan L, Adiconis X, Thompson DA, Levin JZ, Cubillos FA, Regev A: **The evolution, evolvability and engineering of gene regulatory DNA**. *Nature* 2022, **603**:455–463.

This study illustrate how using very large libraries of synthetic DNA to train sophisticated language models can generate extremely accurate sequence-to-function predictions, can can be used to explore the complex fitness landscape of regulatory sequences in yeast.

62. de Boer CG, Taipale J: **Hold out the genome: a roadmap to solving the cis-regulatory code**. *Nature* 2024, **625**:41–50.

63. Wang Y, Wang H, Wei L, Li S, Liu L, Wang X: **Synthetic promoter design in Escherichia coli based on a deep generative network**. *Nucleic Acids Res* 2020, **48**:6403–6412.

64. Taskiran II, Spanier KI, Dickmanken H, Kempynck N, Pancikova A, Eksi EC, Hulselmans G, Ismail JN, Theunis K, Vandepoel R, *et al.*: **Cell-type-directed design of synthetic enhancers**. *Nature* 2024, **626**:212–220.

65. DaSilva LF, Senan S, Patel ZM, Janardhan Reddy A, Gabbita S, Nussbaum Z, Valdez Cordova CM, Wenteler A, Weber N, Tunjic TM, *et al.*: **DNA-diffusion: leveraging generative models for controlling chromatin accessibility and gene expression via synthetic regulatory elements**. *bioRxiv* 2024.

66. Angenent-Mari NM, Garruss AS, Soenksen LR, Church G, Collins JJ: **A deep learning approach to programmable RNA switches**. *Nat Commun* 2020, **11**:5057.

67. Hollerer S, Papaxanthos L, Gumpinger AC, Fischer K, Beisel C, Borgwardt K, Benenson Y, Jeschek M: **Large-scale DNA-based phenotypic recording and deep learning enable highly accurate sequence-function mapping**. *Nat Commun* 2020, **11**:3551.

68. Sample PJ, Wang B, Reid DW, Presnyak V, McFadyen IJ, Morris DR, Seelig G: **Human 5' UTR design and variant effect prediction from a massively parallel translation assay**. *Nat Biotechnol* 2019, **37**:803–809.

69. Hair SC, Fedak S, Wang B, Linder J, Havens K, Certo M, Seelig G: **Optimizing 5'UTRs for mRNA-delivered gene editing using deep learning**. *bioRxiv* 2023. 2023.2006.2015.545194.

70. * Nguyen E, Poli M, Durrant MG, Thomas AW, Kang B, Sullivan J, Ng MY, Lewis A, Patel A, Lou A, *et al.*: **Sequence modeling and design from molecular to genome scale with Evo**. *bioRxiv* 2024. 2024.2002.2027.582234.

In an impressive demo of the power of training ML models at scale, the authors develop Evo, a foundational model for prokaryotes capable of predicting behavior and generating sequences across multiple genetic engineering and scientific domains. Evo can design genes optimized for expression, fitness or other properties, or can be fine-tuned for specific tasks such as design of CRISPR-gRNA complexes targetted to specific loci.

71. Gligorijevic V, Renfrew PD, Kosciolek T, Leman JK, Berenberg D, Vatanen T, Chandler C, Taylor BC, Fisk IM, Vlamakis H, *et al.*:

Structure-based protein function prediction using graph convolutional networks. *Nat Commun* 2021, **12**:3168.

72. de Boer CG, Vaishnav ED, Sadeh R, Abeyta EL, Friedman N, Regev A: **Deciphering eukaryotic gene-regulatory logic with 100 million random promoters**. *Nat Biotechnol* 2020, **38**:56–65.

73. Hartwell LH, Hopfield JJ, Leibler S, Murray AW: **From molecular to modular cell biology**. *Nature* 1999, **402**:C47–C52.

74. Nielsen AA, Der BS, Shin J, Vaidyanathan P, Paralanov V, Strychalski EA, Ross D, Densmore D, Voigt CA: **Genetic circuit design automation**. *Science* 2016, **352**, aac7341.

75. Gallup O, Ming H, Ellis T: **Ten future challenges for synthetic biology**. *Eng Biol* 2021, **5**:51–59.

76. Wong AS, Choi GC, Cheng AA, Purcell O, Lu TK: **Massively parallel high-order combinatorial genetics in human cells**. *Nat Biotechnol* 2015, **33**:952–961.

77. Mutalik VK, Novichkov PS, Price MN, Owens TK, Callaghan M, Carim S, Deutschbauer AM, Arkin AP: **Dual-barcoded shotgun expression library sequencing for high-throughput characterization of functional traits in bacteria**. *Nat Commun* 2019, **10**:308.

78. ** Daniels KG, Wang S, Simic MS, Bhargava HK, Capponi S, Tonai Y, Yu W, Bianco S, Lim WA: **Decoding CAR T cell phenotype using combinatorial signaling motif libraries and machine learning**. *Science* 2022, **378**:1194–1200.
In one of the first demonstrations of ML applied to genetic part compositions, the authors construct libraries of combinatorial intracellular signaling domains in chimeric antigen receptors, and use arrayed measurements to train an ML model to predict output phenotypes. The trained model enables the authors to create new CAR designs that enhance desirable phenotypic behaviors in engineered T cells.

79. Capponi S, Daniels KG: **Harnessing the power of artificial intelligence to advance cell therapy**. *Immunol Rev* 2023, **320**:147–165.

80. ** O'Connell RW, Rai K, Piepergerdes TC, Wang Y, Samra KD, Wilson JA, Lin S, Zhang TH, Ramos E, Sun A, *et al.*: **Ultra-high throughput mapping of genetic design space**. *bioRxiv* 2023.
Describes a highly generalizable platform for the combinatorial assembly and measurement of large gene circuit libraries in an inexpensive and scalable manner. Data generated from these libraries is used to train ML models that predict circuit behavior across entire design spaces, enabling comprehensive understanding of part composability rules and identification of behavior-optimized variants.

81. Kalvapalle PB, Staubus A, Dysart MJ, Gambill L, Gamas KR, Lu LC, Silberg JJ, Stadler LB, Chappell J: **Information storage across a microbial community using universal RNA memory**. *bioRxiv* 2023. 2023.2004.2016.536800.

82. Stone A, Youssef A, Rijal S, Zhang R, Tian XJ: **Context-dependent redesign of robust synthetic gene circuits**. *Trends Biotechnol* 2024.

83. Sherman E, Nobles C, Berry CC, Six E, Wu Y, Dryga A, Malani N, Male F, Reddy S, Bailey A, *et al.*: **INSPIIRED: a pipeline for quantitative analysis of sites of new DNA integration in cellular genomes**. *Mol Ther Methods Clin Dev* 2017, **4**:39–49.

84. Gilliot PA, Gorochowski TE: **Transfer learning for cross-context prediction of protein expression from 5'UTR sequence**. *Nucleic Acids Res* 2024.

85. Valeri JA, Collins KM, Ramesh P, Alcantar MA, Lepe BA, Lu TK, Camacho DM: **Sequence-to-function deep learning frameworks for engineered riboregulators**. *Nat Commun* 2020, **11**:5058.

86. ** Linder J, Srivastava D, Yuan H, Agarwal V, Kelley DR: **Predicting RNA-seq coverage from DNA sequence as a unifying model of gene regulation**. *bioRxiv* 2023. 2023.2008.2030.555582.
The authors develop Borzoi, a multi-modal foundation model that incorporates >500 kb of sequence input to predict the relationship between sequence and cis-regulatory code, including chromatin marks, TF binding, and specifically RNA-seq coverage directly. The authors show that Borzoi is competitive with, and often outperforms state-of-the-art models trained on specific regulatory functions.

87. Agarwal V, Inoue F, Schubach M, Martin BK, Dash PM, Zhang Z, Sohota A, Noble WS, Yardimci GG, Kircher M, *et al.*: **Massively parallel characterization of transcriptional regulatory elements in three diverse human cell types**. *bioRxiv* 2023.

88. de Almeida BP, Schaub C, Pagani M, Secchia S, Furlong EEM, Stark A: **Targeted design of synthetic enhancers for selected tissues in the Drosophila embryo**. *Nature* 2024, **626**:207–211.

89. Huang J, Child R, Rao V, Liu H, Satheesh S, Coates A: *Active learning for speech recognition: the power of gradients*. arXiv preprint; 2016.

90. Hu R, Mac Namee B, Delany SJ: **Active learning for text classification with reusability**. *Expert Syst Appl* 2016, **45**:438–449.

91. Huang K, Lopez R, Hütter J-C, Kudo T, Rios A, Regev A: **Sequential optimal experimental design of perturbation screens guided by multi-modal priors**. *bioRxiv* 2023. 2023.2012.2012.571389.

92. * Friedman RZ, Ramu A, Lichtarge S, Myers CA, Granas DM, Gause M, Corbo JC, Cohen BA, White MA: **Active learning of enhancer and silencer regulatory grammar in photoreceptors**. *bioRxiv* 2023.
An outstanding early demonstration of applying active learning to develop an ML model that can understand native regulation. The authors iteratively design variants to train a model to classify cis-regulatory sequence arrays as enhancers vs silencers in developing mouse retina, showing ~2-fold improvement in predictive power and accuracy using active learning guided design over random sampling.

93. Yin CH, Castillo Hair S, Woo Byeon G, Bromley P, Meuleman W, Seelig G: **Iterative deep learning-design of human enhancers exploits condensed sequence grammar to achieve cell type-specificity**. *bioRxiv* 2024. 06.

94. Shapley LS: **17. A value for n-person games**. In Harold William K, Albert William T. *Contributions to the theory of games (AM-28)*, vol. II. Princeton University Press; 1953:307–318.

95. Simonyan K, Vedaldi A, Zisserman A: *Deep inside convolutional networks: visualising image classification models and saliency maps*. arXiv preprint; 2013. arXiv:1312.6034.

96. Ribeiro MT, Singh S, Guestrin C: **Why should i trust you?" Explaining the predictions of any classifier**. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*; 2016:1135–1144.

97. Singh A, Sengupta S, Lakshminarayanan V: **Explainable deep learning models in medical image analysis**. *J Imaging* 2020, **6**.

98. Murdoch WJ, Singh C, Kumbier K, Abbasi-Asl R, Yu B: **Definitions, methods, and applications in interpretable machine learning**. *Proc Natl Acad Sci U S A* 2019, **116**:22071–22080.

99. Hillerton T, Secilmis D, Nelander S, Sonnhammer ELL: **Fast and accurate gene regulatory network inference by normalized least squares regression**. *Bioinformatics* 2022, **38**:2263–2268.

100. Radivojevic T, Costello Z, Workman K, Garcia Martin H: **A machine learning Automated Recommendation Tool for synthetic biology**. *Nat Commun* 2020, **11**:4879.

101. Rapp JT, Bremer BJ, Romero PA: **Self-driving laboratories to autonomously navigate the protein fitness landscape**. *Nat Chem Eng* 2024, **1**:97–107.

102. Burger B, Maffettone PM, Gusev VV, Aitchison CM, Bai Y, Wang X, Li X, Alston BM, Li B, Clowes R, *et al.*: **A mobile robotic chemist**. *Nature* 2020, **583**:237–241.