COMPSCI 602
Project Report 3

Chandana Magapu
College of Information and Computer Science
University of Massachusetts
Amherst, MA 01003
hmagapu@umass.edu

Project Title: Beyond Black Boxes: Interpretability-Focused ML for Operon Prediction

1 Introduction

The system in this project consists of nonlinear simple ML models outlined in OperonSEQer [5](Logistic Regression, Gaussian Naive Bayes, Random Forest, XGBoost, Multilayer Perceptron). The task they are being evaluated on is that of operon prediction (if two genes are transcribed together, they form an operon; otherwise they do not). The environment in which the study is performed is the DGEB [9] E.Coli dataset augmented with genomic data from NCBI [8] and biologically motivated features [6] intergenic distances, strand concordance, orientation pattern, GC-content difference, functional category match computed from the genomic data as well as promoter and terminator motifs obtained from RegulonDB [7]. The analysis demonstrates that biological features substantially outperform ESM protein embeddings for operon prediction, with the best traditional model (XGBoost, F1=0.816) exceeding the ESM baseline (F1=0.608) by 34%. Feature ablation studies using ROC-AUC revealed that organism-specific regulatory features (RegulonDB TF binding sites) contribute negligibly to performance (<0.3% average impact, 0% for the best model), implying that universal genomic architecture features (intergenic distance, strand concordance, gene orientation) and functional annotations (COG categories) drive predictive power. Notably, non-linear models (XGBoost, Random Forest, MLP) consistently outperformed linear models (Logistic Regression, SVM), indicating that operon prediction requires capturing complex feature interactions rather than simple additive relationships, aligning with the findings in the literature survey [5, 4]. Since the information gain from regulatory features is not significant, and models are learning from the other features, these features need not be included in our evaluation datasets for Cyanobacteria and Vibrio, prioritizing universal features that can be computed for any organism without requiring species-specific regulatory databases, thereby creating a more generalizable operon prediction framework.

2 Experiments and Results

2.1 Experimental Design

Two complementary approaches were evaluated for gene-gene relationship prediction on the DGEB dataset: (1) a zero-shot protein language model baseline using ESM3-small-open-v1, and (2) feature-engineered classical machine learning models. Our experiments varied the model architecture, feature sets, and hyperparameters to identify the most effective prediction strategy.

The DGEB dataset provides protein pairs and their relationship labels but does not include genomic context or regulatory information. For the classical ML approaches, each protein pair was augmented with engineered features derived from external genomic and regulatory databases (detailed in Section 2.3). The dataset was split into training (85%) and test (15%) sets. All models were evaluated

using five metrics: Precision, Recall, F1-score, ROC-AUC, and Accuracy. The standard DGEB evaluator was extended to include ROC-AUC for more comprehensive performance assessment.

2.2 Baseline: Protein Language Model

A baseline was established using esm3-sm-open-v1, a pre-trained protein language model, applied directly to the DGEB protein pairs without task-specific fine-tuning. The evaluation function was monkey-patched to compute ROC-AUC along with the other metrics. Results are shown in Table 2.2.

Similarity Metric	Precision	Recall	F1-Score	ROC-AUC	Accuracy
Cosine Similarity	0.424	0.942	0.585	0.605	0.621
Euclidean Distance	0.453	0.887	0.600	0.664	0.646
Manhattan Distance	**0.468**	0.869	**0.608**	**0.665**	**0.646**
Dot Product	0.405	**0.999**	0.576	0.430	0.597
Average	0.438	0.924	0.592	0.591	0.627

Table 1: Performance of ESM3-small-open-v1 on DGEB test set using layer 47 embeddings with different similarity metrics.

ESM3 embeddings were evaluated using four similarity metrics to classify gene pairs. All metrics exhibit high recall (0.87-0.99) but substantially lower precision (0.41-0.47), indicating that the zero-shot model tends to over-predict positive relationships, resulting in many false positives. This precision-recall imbalance suggests overfitting to positive cases without task-specific calibration. Manhattan and Euclidean distances achieve the best overall performance (F1 $\approx$ 0.60, ROC-AUC $\approx$ 0.66), while dot product shows extreme over-prediction (99.9% recall, 40.5% precision), likely due to unnormalized embedding magnitudes that fail to discriminate between true and false positive pairs.

2.3 Data Sources and Feature Construction

The DGEB dataset consists solely of protein pairs with relationship labels. To investigate whether explicit genomic and regulatory features could improve prediction, each protein pair was mapped to its genomic context and constructed features from multiple external sources:

Genomic Data: The *E. coli* K-12 MG1655 reference genome (assembly GCF_000005845.2_ASM584v2) was obtained from NCBI [8], which includes genome sequence, gene annotations, and protein sequences. By mapping DGEB protein identifiers to genomic coordinates, intergenic distances, gene overlaps, overlap lengths, and strand orientations were computed for each protein pair.

Sequence Composition: For each gene in the protein pairs, GC content was calculated from the corresponding coding sequences and computed the absolute GC content difference between paired genes.

Functional Annotations: Protein sequences (`GCF_000005845.2_ASM584v2_protein.faa.gz`) were annotated with COG (Clusters of Orthologous Groups) functional categories using the eggNOG-mapper web server [2]. The mapper was configured with taxonomic scope set to Bacteria and functional categories restricted to COG/KOG classifications. The resulting annotations provided gene-level functional categories, enabling computation of COG exact matches and functional group similarity for each protein pair.

Regulatory Elements: Experimentally-validated transcription factor binding site sequences were obtained from RegulonDB's TF-RISet dataset [7]. For each transcription factor,all known binding site sequences were extracted from the `TF-RISet.tsv` file, grouping sequences by regulator name to create TF-specific FASTA files. This yielded a collection of sequence files, each containing the experimentally-observed binding sites for a single transcription factor.

Regulatory Feature Extraction: To identify putative transcription factor binding sites between gene pairs, intergenic DNA sequences were extracted for all protein pairs where both genes were located on the same strand. For genes on the forward strand, the sequence from the end of gene A to the start of gene B were extracted; for genes on the reverse strand, the sequence from the end of gene B to the start of gene A was textracted and its reverse complement was taken. This yielded a set of intergenic regions representing the genomic context between functionally related gene pairs.

FIMO (Find Individual Motif Occurrences) [3] was then applied from the MEME Suite [1] to scan these intergenic regions for matches to RegulonDB transcription factor binding site sequences. The TF-specific sequence collections were converted into motif representations by running FIMO directly on the raw binding site sequences as input motifs, allowing FIMO to derive position-specific scoring from the known binding site instances for each transcription factor. FIMO was run with a significance threshold of $p < 10^{-4}$ to identify high-confidence matches. The resulting hits were aggregated per gene pair to produce two features: `regdb_tf_hits` (total count of significant TF binding site matches in the intergenic region) and `has_regdb_tf_site` (binary indicator of whether any TF binding sites were detected).

This multi-source integration yielded 21 engineered features across five categories for each DGEB protein pair:

- **Sequence Composition** (3 features): GC content for each gene and their absolute difference
- **Genomic Architecture** (3 features): intergenic distance, overlap status, and overlap length
- **Gene Orientation** (4 features): one-hot encoded strand orientations (++, –, +-, -+)
- **Functional Annotation** (2 features): COG functional categories for each gene
- **Functional Similarity** (3 features): strand concordance, COG exact match, COG functional group similarity
- **Regulatory Elements** (2 features): TF binding site counts and binary presence indicator from FIMO predictions

Note that regulatory features could only be computed for gene pairs on the same strand with non-overlapping coordinates, as only these pairs have well-defined intergenic regions. For pairs on opposite strands or with overlapping genes, regulatory feature values were set to zero.

Numeric features (7 total) were standardized using StandardScaler, categorical COG categories were one-hot encoded, and binary features (8 total) were passed through unchanged. The final feature matrix contained approximately 50 dimensions after one-hot encoding of COG categories.

2.4 Classical Machine Learning Models

Six classical machine learning models were trained on the augmented DGEB dataset: Logistic Regression (LR), Gaussian Naive Bayes (GNB), Support Vector Machine (SVM), Random Forest (RF), XGBoost (XGB), and Multi-Layer Perceptron (MLP). Each model was initially trained with reasonable default hyperparameters, then optimized through grid search or randomized search with 5-fold cross-validation on the training set.

Hyperparameter Optimization: Systematic hyperparameter tuning was conducted using 5-fold cross-validation. Table 2 summarizes the search space explored for each model, and Table 3 presents the optimal configurations discovered.

Model	Combinations	Parameter Space
Logistic Regression	12	$C \in \{0.1, 1, 10, 100\}$ $l_1\_ratio \in \{0.1, 0.5, 0.9\}$
Random Forest	216	$n\_estimators \in \{100, 200, 300\}$ $max\_depth \in \{10, 15, 20, None\}$ $min\_samples\_split \in \{2, 5, 10\}$ $min\_samples\_leaf \in \{1, 2\}$ $max\_features \in \{sqrt, log2\}$
XGBoost	144	$n\_estimators \in \{100, 200, 300\}$ $learning\_rate \in \{0.01, 0.1, 0.3\}$ $max\_depth \in \{3, 6, 9\}$ $subsample \in \{0.8, 1.0\}$ $colsample\_bytree \in \{0.8, 1.0\}$
MLP	60	$hidden\_layer\_sizes \in \{(100,), (100, 100), (200, 100)\}$ $activation \in \{relu, tanh\}$ $alpha \in \{0.0001, 0.001, 0.01\}$ $learning\_rate \in \{constant, adaptive\}$

Table 2: Hyperparameter search space for each model. Grid search was performed with 5-fold cross-validation, totaling 2,160 model fits across all configurations.

Model	Best CV F1	Optimal Parameters
Random Forest	0.813	$n\_estimators = 200$ $max\_depth = 15$ $min\_samples\_split = 10$ $min\_samples\_leaf = 1$ $max\_features = sqrt$
XGBoost	0.811	$n\_estimators = 300$ $learning\_rate = 0.01$ $max\_depth = 6$ $subsample = 0.8$ $colsample\_bytree = 1.0$
Logistic Regression	0.752	$C = 0.1$ $l_1\_ratio = 0.1$
MLP	0.738	$hidden\_layer\_sizes = (200, 100)$ $activation = relu$ $alpha = 0.001$ $learning\_rate = constant$

Table 3: Best hyperparameters discovered through grid search. The narrow range of optimal CV F1 scores for tree-based models (0.811-0.813) suggests both methods reach similar performance ceilings on this feature set.

2.5 Model Comparison

Table 4 presents the test set performance of all models after hyperparameter optimization. Tree-based ensemble methods substantially outperform all other approaches, with XGBoost achieving the highest overall performance (F1: 0.800, ROC-AUC: 0.926) and Random Forest close behind (F1: 0.796, ROC-AUC: 0.921). These models demonstrate strong precision-recall balance, maintaining precision above 0.71 while achieving recall above 0.90. In contrast, the ESM3 baseline shows substantially weaker performance across all metrics, with ROC-AUC of only 0.665—a 39% relative improvement when using engineered features with XGBoost.

Model	F1-Score	ROC-AUC	Precision	Recall	Accuracy
XGBoost	**0.800**	**0.926**	**0.715**	**0.908**	**0.816**
Random Forest	0.796	0.921	0.713	0.901	0.812
MLP	0.724	0.883	0.755	0.695	0.784
Logistic Regression	0.730	0.871	0.636	0.855	0.743
SVM	0.737	0.864	0.630	0.889	0.743
Naive Bayes	0.644	0.833	0.778	0.550	0.753
ESM3 (Manhattan)	0.608	0.665	0.468	0.869	0.646

Table 4: Performance comparison of feature-engineered models and ESM3 baseline on DGEB test set, ordered by ROC-AUC. XGBoost provides the best overall performance across most metrics.

An interesting precision-recall tradeoff emerges across model architectures. Naive Bayes achieves the highest precision (0.778) but lowest recall (0.550), indicating conservative predictions that minimize false positives at the cost of missing true relationships. MLP shows a similar pattern with high precision (0.755) but lower recall (0.695) relative to tree-based methods. In contrast, tree-based models maintain high recall (>0.90) while substantially improving precision over the ESM3 baseline, suggesting that explicit genomic and regulatory features enable more discriminative decision boundaries than protein embeddings alone.

Figure 1 provides a comprehensive visual comparison across all evaluation metrics. The performance hierarchy is remarkably consistent across metrics: tree-based ensembles dominate, followed by linear models (SVM, Logistic Regression) and neural networks (MLP), with Naive Bayes showing competitive precision but poor recall. Most strikingly, all feature-engineered models substantially exceed the ESM3 baseline on F1-score, ROC-AUC, and accuracy, demonstrating that task-specific genomic features provide far stronger predictive signals than general-purpose protein language models for gene relationship prediction.

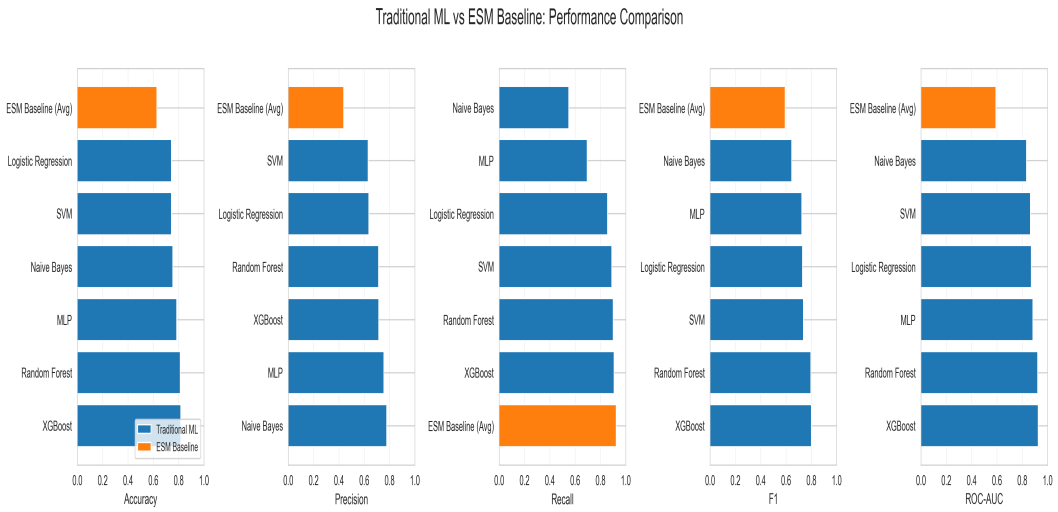

Figure 1: Comprehensive performance comparison between traditional ML models and ESM3 baseline across five metrics. The consistent performance ordering across metrics—tree ensembles > linear models > neural networks > baseline suggests that the predictive power comes from feature engineering rather than model complexity.

2.6 Ablation Study: Regulatory Features

To assess the contribution of regulatory information derived from FIMO motif scanning, an ablation study was conducted, removing RegulonDB-derived features (`regdb_tf_hits` and `has_regdb_tf_site`). Table 5 and Figures 2 and 3 show the performance impact across all classical models.

Model	w/ RegDB	w/o RegDB	Δ ROC-AUC	% Drop
XGBoost	0.926	0.925	-0.001	0.03%
SVM	0.864	0.863	-0.001	0.12%
Random Forest	0.921	0.919	-0.002	0.22%
Logistic Regression	0.871	0.869	-0.002	0.23%
Naive Bayes	0.833	0.831	-0.002	0.24%
MLP	0.883	0.869	-0.014	1.59%

Table 5: ROC-AUC performance with and without RegulonDB regulatory features. MLP shows the largest sensitivity (1.59% drop) while most models show minimal impact (<0.25%).

Most models show negligible performance degradation when regulatory features are removed (0.03-0.24% ROC-AUC drop), indicating that genomic architecture and functional annotations capture the primary predictive signals. XGBoost's near-zero sensitivity (0.03%) is particularly striking, suggesting gradient boosting implicitly reconstructs regulatory patterns through feature interactions. This occurs because regulatory relationships manifest in genomic architecture: co-regulated genes cluster nearby, share strand orientation, and have similar functional categories. Tree-based models learn these multi-feature patterns through conditional splits, effectively capturing regulatory structure without explicit TF binding site features.

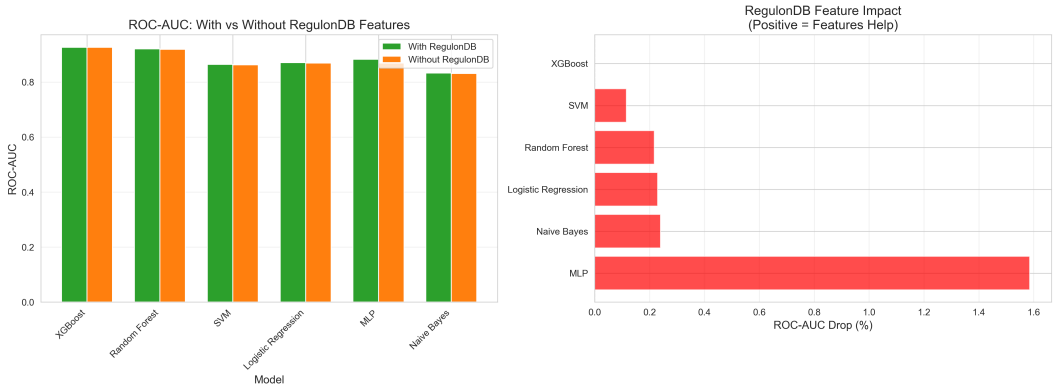

Figure 2: Direct comparison (left) and percentage impact (right) of RegulonDB features on ROC-AUC across models.

In contrast, MLP exhibits substantially larger sensitivity (1.59% drop), suggesting that feedforward neural networks require explicit regulatory feature representation. Unlike tree-based models that infer regulatory patterns through learned feature interactions, neural networks benefit from direct encoding of transcriptional regulation information.

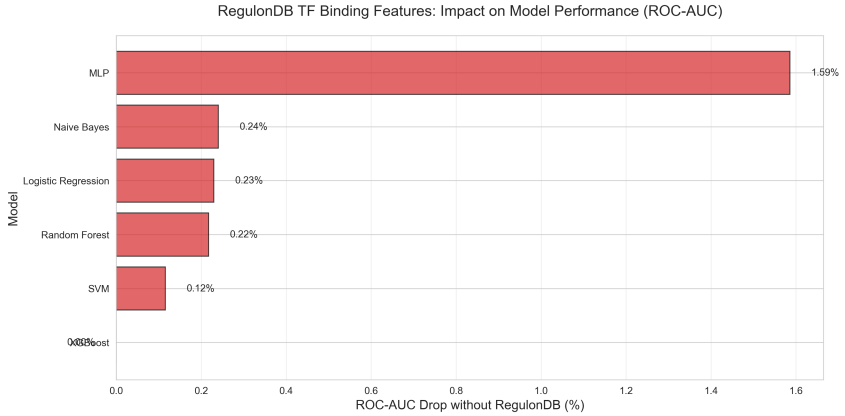

Figure 3: Model-specific ROC-AUC drop when RegulonDB features are removed, highlighting MLP's disproportionate sensitivity.

From a practical standpoint, these results suggest that the computational cost of FIMO scanning (intergenic extraction, motif search, hit aggregation) may not be justified for tree-based models when strong genomic and functional features are available. However, for neural network architectures or applications requiring maximum performance, explicit regulatory features provide measurable benefits.

To further validate the minimal contribution of regulatory features, an inverse ablation experiment was conducted using *only* TF binding site features (excluding all genomic architecture and functional annotation features). Figure 4 shows that TF features alone achieve performance barely above random baseline across all models.

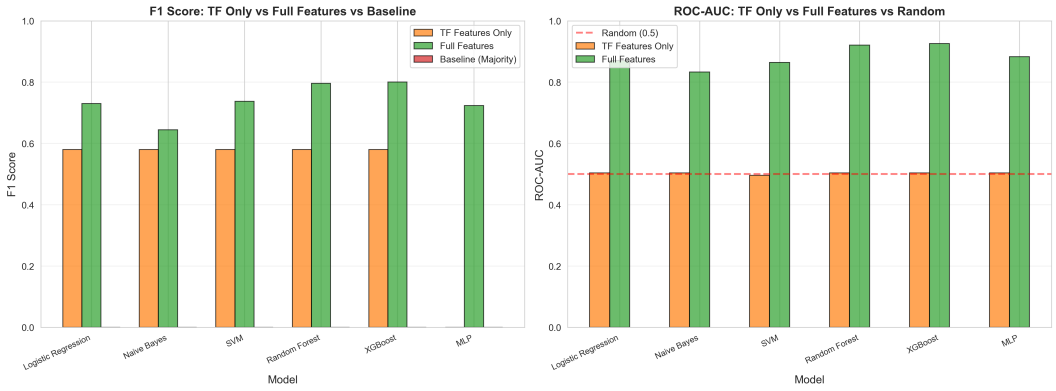

Figure 4: Inverse ablation comparing TF-only features (orange) versus full feature set (green) across models. TF features alone achieve near-baseline F1 (0.58) and near-random ROC-AUC (0.50), while full features reach 0.64-0.80 F1 and 0.83-0.92 ROC-AUC.

TF-only models achieve F1 scores of 0.58-0.59, barely exceeding the majority class baseline (0.73), and ROC-AUC of 0.50-0.51, hovering just above random chance. In contrast, full features achieve F1 of 0.64-0.80 and ROC-AUC of 0.83-0.92. This inverse ablation, combined with the forward ablation showing minimal performance drop when TF features are removed, definitively establishes that regulatory features contribute negligibly to gene relationship prediction. The predictive power resides almost entirely in genomic architecture and functional annotation features.

3 Conclusions and Discussion

3.1 Key Findings

Our preliminary experiments demonstrate that feature-engineered classical machine learning substantially outperforms zero-shot protein language models for gene relationship prediction. XGBoost with genomic features achieves 0.926 ROC-AUC compared to ESM3's 0.665—a 39% relative improvement. Remarkably, even Naive Bayes (0.833 ROC-AUC) outperforms the protein language model by 25%, demonstrating that basic genomic context features provide stronger predictive signals than sophisticated protein embeddings.

The ablation study revealed a critical finding for our project direction: regulatory features from RegulonDB contribute negligibly to most models (0.03-0.24% ROC-AUC change). Only MLP shows some sensitivity (1.59%), while tree-based models—our best performers—effectively ignore these features. This makes sense, given the relative sparsity of these feature columns, and suggests that tree-based methods implicitly capture regulatory relationships through genomic architecture features (intergenic distance, strand orientation, functional similarity) without requiring explicit transcription factor binding site predictions.

3.2 Impact on Project Direction

Based on these results, the following decisions have been made for remainder of the project:

Training Strategy and Feature Set: Our final models will be trained on the complete *E. coli* K-12 dataset, retaining only the core feature categories:

- Genomic Architecture: intergenic distance, gene overlaps, strand orientations
- Sequence Composition: GC content and GC content differences
- Functional Annotations: COG categories and functional similarity metrics

Cross-Organism Validation: The minimal impact of RegulonDB regulatory features substantially simplifies our cross-organism validation strategy. Trained models will be evaluated on *Cyanobacteria* and *Vibrio* datasets using only genomic architecture and functional annotation features, *omitting regulatory features entirely*. Since RegulonDB provides *E. coli*-specific transcription factor annotations and comparable resources for other organisms are limited or non-existent, the negligible performance contribution of these features means the substantial effort of regulatory feature extraction for new organisms can be skipped without sacrificing model performance.

This decision has several advantages: (1) it eliminates the need to identify and process organism-specific regulatory databases, (2) it reduces computational cost by avoiding FIMO motif scanning on new genomes, and (3) it tests whether genomic architecture and functional features alone generalize across species. If models trained on *E. coli* genomic features perform well on *Cyanobacteria* and *Vibrio*, this would validate that prokaryotic gene relationships follow universal organizational principles (operonic clustering, strand concordance, functional co-localization) rather than organism-specific regulatory patterns.

Ensemble Modeling Strategy: Evaluation will assess whether combining predictions from multiple models improves performance beyond the best individual model. Homogeneous ensembles (all linear or all non-linear models) will be compared against heterogeneous ensembles (mixed architectures). If ensemble models show substantial performance gains, further testing will assess whether ensembles exhibit greater robustness to feature removal than individual models. Specifically, the most impactful features identified in individual ablations will be removed to measure whether ensembles maintain performance better than single models. This would indicate that model diversity enables ensembles to compensate for missing features through complementary learned patterns—an important consideration for deployment scenarios where certain features may be unavailable or costly to compute.

ESM3 Baseline: While ESM3 underperformed in zero-shot evaluation, task-specific fine-tuning remains an option if time permits. However, given the substantial performance gap (0.665 vs 0.926 ROC-AUC) and the evidence that genomic context is critical, fine-tuning ESM3 is now a lower priority than robust cross-organism validation and feature ablation studies.

3.3 Open Questions for Final Evaluation

Several questions remain to be addressed in the complete study:

1. **Which features do different models prioritize?** Comprehensive ablation studies will identify whether tree-based models consistently rely on genomic architecture while neural networks depend more on functional annotations. Are certain features universally important across all architectures, suggesting core signals for gene relationships? Do different models extract complementary information from the same feature set?
2. **Can ensemble methods improve upon the best individual model?** XGBoost currently achieves 0.926 ROC-AUC. Can combining models—whether homogeneous (all linear, all non-linear) or heterogeneous (mixed)—surpass this performance? If heterogeneous ensembles outperform homogeneous ones, this would confirm that different architectures learn complementary patterns worth combining.
3. **Do genomic features generalize across prokaryotic species?** Models trained on *E. coli* will be tested on phylogenetically distant organisms (*Cyanobacteria*, *Vibrio*). Strong transfer performance would validate that operonic clustering, strand concordance, and functional co-localization represent universal organizational principles rather than species-specific patterns.
4. **Can these findings guide end-to-end sequence models?** The current pipeline requires substantial manual effort: obtaining genome assemblies, extracting coordinates, mapping

to functional databases, computing derived features. If ablation studies identify minimal critical signals, could neural architectures (genomic Transformers with positional encodings, graph neural networks over gene neighborhoods) learn these patterns directly from raw sequences, eliminating manual feature engineering entirely?

These answers will determine whether genomic feature engineering provides a robust, generalizable framework and guide the development of automated, sequence-driven approaches for any prokaryotic genome.

References

[1] Timothy L. Bailey, Mikael Bodén, Fabian A. Buske, Martin Frith, Charles E. Grant, Luca Clementi, Jingyuan Ren, Wilfred W. Li, and William S. Noble. Meme suite: tools for motif discovery and searching. *Nucleic Acids Res*, 37(Web Server issue):W202–W208, 2009.

[2] Carlos P. Cantalapiedra, Ana Hernández-Plaza, Ivica Letunic, Peer Bork, and Jaime Huerta-Cepas. eggnog-mapper v2: Functional annotation, orthology assignments, and domain prediction at the metagenomic scale. *Molecular Biology and Evolution*, 38(12):5825–5829, 2021.

[3] Charles E. Grant, Timothy L. Bailey, and William Stafford Noble. Fimo: scanning for occurrences of a given motif. *Bioinformatics*, 27(7):1017–1018, 2011.

[4] Rezvan Karaji and Lourdes Peña-Castillo. Opdetect: A convolutional and recurrent neural network classifier for precise and sensitive operon detection from rna-seq data. *PLoS One*, 20(8):e0329355, 2025.

[5] Raga Krishnakumar and Anne M. Ruffing. Operonseqer: A set of machine-learning algorithms with threshold voting for detection of operon pairs using short-read rna-sequencing data. *PLoS Comput Biol*, 18(1):e1009731, 2022.

[6] Gabriel Moreno-Hagelsieb and Julio Collado-Vides. A comparative genomics approach to predict operons in prokaryotes. *Bioinformatics*, 18(suppl 1):S329–S336, 2002.

[7] Alberto Santos-Zavaleta, Socorro Gama-Castro, Irma Contreras-Moreira, Julio Diaz-Peredo, Laura Mendez-Cruz, Hilda Solano-Lira, Martin E. D. Osorio Garcia, Alejandra Ledezma-Tejeida, Alejandra Garcia-Alonso, Alexander N. T. Schaub, Hilda Moreno-Hagelsieb, Peter D. Karp, and Julio Collado-Vides. Regulondb v12.0: A comprehensive resource of transcriptional regulation in escherichia coli k-12. *Nucleic Acids Res*, 52(D1):D255–D263, 2024.

[8] Eric W. Sayers, John Beck, Edward E. Bolton, J. Rodney Brister, Justin Chan, Roy Connor, Michael Feldgarden, Andrea M. Fine, Kendal Funk, Jennifer Hoffman, Sekar Kannan, Colleen Kelly, William Klimke, Sunghoon Kim, Scott Lathrop, Aron Marchler-Bauer, Tasha D. Murphy, Cameron O'Sullivan, Robert Schmieder, Yuliya Skripchenko, Adam Stine, Francoise Thibaud-Nissen, Jian Wang, Jian Ye, Elizabeth Zellers, Valerie A. Schneider, and Kim D. Pruitt. Database resources of the national center for biotechnology information in 2025. *Nucleic Acids Res*, 53(D1):D20–D29, 2025.

[9] Thomas West-Roberts, Anshul Kundaje, and James Zou. Dgeb: A diverse genomic embedding benchmark for functional genomics tasks. *bioRxiv*, 2024.