Identifying regularization schemes that make feature attributions faithful

Julius Adebayo[1] **Samuel Stanton**[1] **Simon Kelow**[1] **Michael Maser**[1] **Richard Bonneau**[1]
Vladimir Gligorijević[1] **Kyunghyun Cho**[1,3,4] **Stephen Ra**[1] **Nathan Frey**[1]
[1]Prescient Design, Genentech
[3]Department of Computer Science, New York University
[4]Center for Data Science, New York University

Abstract

Feature attribution methods assign a score to each input dimension as a measure of the relevance of that dimension to a model's output. Despite wide use, the feature importance rankings induced by gradient-based feature attributions are unfaithful, that is, they do not correlate with the input-perturbation sensitivity of the model—unless the model is trained to be adversarially robust. Here we demonstrate that these concerns translate to models trained for protein function prediction tasks. Despite making a model's gradient-based attributions faithful to the model, adversarial training has low real-data performance. We find that independent Gaussian noise corruption is an effective alternative, to adversarial training, that confers faithfulness onto a model's gradient-based attributions without performance degradation. On the other hand, we observe no meaningful faithfulness benefits from regularization schemes like dropout and weight decay. We translate these insights to a real-world protein function prediction task, where the gradient-based feature attributions of noise-regularized models, correctly indicate low sensitivity to irrelevant gap tokens in a protein's sequence alignment.

1 Introduction

A *faithful* feature attribution, of a function's output, assigns a score to each input feature that indicates the magnitude of the change, in the function's output, when that feature is ablated. Inspecting heat maps of feature attributions, especially those derived from gradient-based methods, is a popular approach for 'explaining' models trained on protein sequences for function prediction. However, the setting under which feature attributions are faithful remains unclear. Hooker et al. [2018] found that the rankings induced by several, widely used, gradient-based feature attribution methods are no more faithful than a random ranking of the input features. However, Shah et al. [2021] showed that gradient-based feature attributions of adversarially robust models, unlike those of unregularized models, are faithful. Yet, it is widely observed that robust models have low non-worst case—real data—performance. For example, a state-of-art L2-norm robust ($\epsilon = 0.5$) classifier, on ImageNet, has accuracy 71 percent compared to 80.4 percent for an unregularized model [Carlini et al., 2022]. Motivated by these findings, in this work, we ask:

- Are the input-gradients of unregularized protein property prediction models also unfaithful to the model?
- Can we identify alternative regularization schemes that donot incur low real-data performance, but also confer faithfulness onto gradient-based attributions?

The NeurIPS 2023 Workshop on New Frontiers of AI for Drug Discovery and Development (AI4D3 2023), New Orleans, LA, USA, 2023.

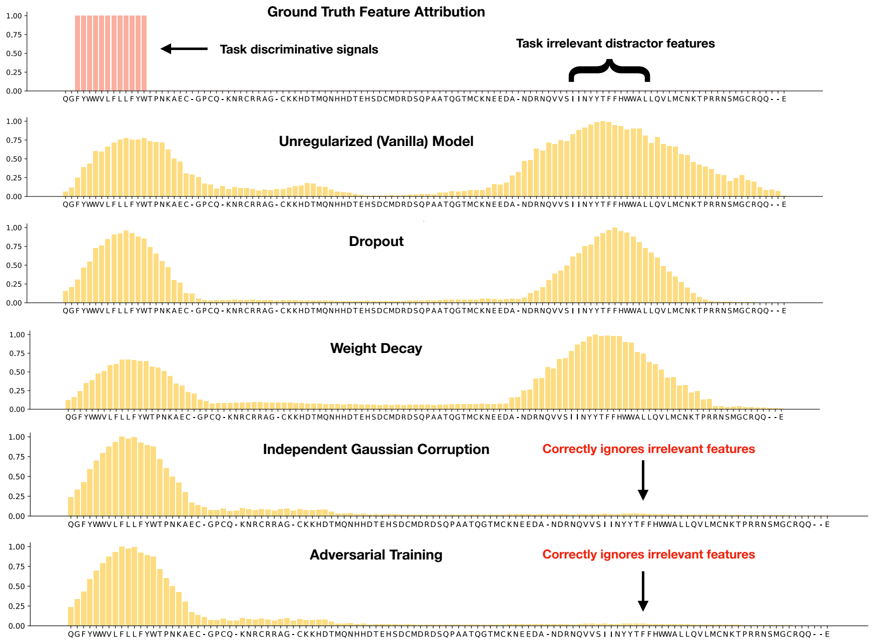

Figure 1: **The effect of regularization on the faithfulness of model's input-gradients.** We construct a task where the signals that determine the model output, for a particular input, are localized to specific input dimensions, and known a priori. In addition, we inject a 'distractor' signal—a short subsequence that is correlated with the signal but is present across all inputs, so independent of the label. We then train models, under different regularization schemes, on this task, to **100 percent test set accuracy on a dataset that does not include the distractor signal**. Further, any change or perturbation to the distractor regions does not change the model's prediction (and logits). Consequently, we expect any feature attribution method to only assign relevance to the input dimensions that contain the task signals. Of the regularization schemes we test, we find that only independent noise corruption of inputs during training matches the effect of adversarial training on a model's input-gradients.

- Under the appropriate regularization scheme, do input-gradient attributions remain effective for real-world protein function prediction tasks?

We design synthetic tasks that allow us to train models with verifiable feature ablation ranking . Since the model feature ablation is known, a priori, the faithfulness of a feature attribution method can computed as its similarity to the model feature ablation. We use this setup to measure the effect of various regularization schemes on the faithfulness of several gradient-based feature attribution methods. We find that:

- the input-gradient and smoothgrad feature attributions of unregularized protein function models are indeed unfaithful. On the contrary, integrated gradients and a gradient approximation of the shapley value are faithful attributions, even for unregularized models.

- independent noise addition, during training, is an effective alternative to adversarial training, for making the input-gradient attribution of a model faithful.

Taken together, our results help clarify the settings under which the output of gradient-based feature attributions can be used to *explain* a model's output for protein function prediction tasks.

2 Background

We now define key terms, and give an overview of the methods that we study. In addition, we discuss the effect of adversarial training on a model's feature attribution, and the alternative regularization schemes whose effect on a model's feature attribution we investigate.

2.1 Basic Concepts

Setting. We consider the classification setting, where we are given a dataset of feature-label pairs $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=0}^{n-1}$, with a corresponding feature space $\mathcal{X} \subset \mathbb{R}^d$ and label space $\mathcal{Y} \subset \mathbb{N}$. We wish to understand how a *ground-truth* function, $f : \mathcal{X} \mapsto \Delta(\mathcal{Y})$ mediates between $\mathbf{x}$ and y.

Ground-Truth Feature Ablation: one of the simplest ways to understand the relationship between $\mathbf{x}$ and y is *perturbation analysis*, which consists in specifying a set of perturbations $\mathcal{P} = \{\mathbf{p}_0, \dots, \mathbf{p}_{J-1}\}$ and evaluating the change in f when those perturbations are applied to a specific input $\varepsilon_j(\mathbf{x}) = D(f(\mathbf{x}), f(\mathbf{x} - \mathbf{p}_j)),\ j \in \{0, \dots, J-1\}$, where D is some divergence measure on $\Delta(\mathcal{Y})$, e.g. KL divergence. Often we are more interested in the *rank* of $\varepsilon_j(\mathbf{x})$ relative to the other elements of $\mathcal{P}$ than the magnitude of ε_j itself. A particularly noteworthy set of perturbations is a *feature ablation* study, where $\mathcal{P}(\mathbf{x}) = \{\mathbf{x} \odot \mathbf{e}_0, \dots, \mathbf{x} \odot \mathbf{e}_{d-1}\}$ (so $J = d$), which corresponds to setting the j-th feature of $\mathbf{x}$ to 0 and holding all other features constant. For example, we could set each pixel in an image to 0 and ask human annotators about the label.

Function Approximation: When f is expensive to evaluate (e.g. by averaging human annotations or running an experiment), ground-truth feature ablation is not practical, requiring us to introduce an estimate of f via regularized empirical risk minimization (ERM): $\arg\min_{\hat{f}_\theta} 1/n \sum_i^n \ell(\hat{f}_\theta(\mathbf{x}_i), y_i) + \lambda \mathcal{R}(\hat{f}_\theta)$, where ℓ is a loss function, $\mathcal{R}$ is a regularizer, and $\lambda > 0$. We say that $\hat{f}_\theta$ is *congruent* with f if there is substantial agreement between f and $\hat{f}_\theta$, i.e. the generalization error of $\hat{f}_\theta$ is small everywhere in $\mathcal{X}$.

$$D(f(\mathbf{x}), \hat{f}_\theta(\mathbf{x})) < \epsilon, \quad \forall \mathbf{x} \in \mathcal{X} \tag{1}$$

Model Feature Ablation: obtaining an estimator $\hat{f}_\theta$ allows us to cheaply approximate $f(\mathbf{x})$ and the effect of perturbations about $\mathbf{x}$, $\hat{\varepsilon}_j(\mathbf{x}) = D(\hat{f}_\theta(\mathbf{x}), \hat{f}_\theta(\mathbf{x} - \mathbf{p}_j))$. A congruent model will result in model feature ablations that are congruent with ground-truth feature ablations, meaning there will be consistency between the features $\hat{f}_\theta$ is sensitive to and the features f is sensitive to. When our primary interest is the relative sensitivity of the true function to different features, we can relax congruency to *similarity*. We say $\hat{\varepsilon}$ is similar to ε if the following condition is satisfied:

$$\begin{gathered}\hat{\varepsilon}_j(\mathbf{x}) < \hat{\varepsilon}_{j'}(\mathbf{x}) \iff \varepsilon_j(\mathbf{x}) < \varepsilon_{j'}(\mathbf{x}), \\ \forall \mathbf{x} \in \mathcal{X},\ \forall j, j' \in \{0, \dots, J-1\}.\end{gathered} \tag{2}$$

Continuing our example, instead of using human annotators we could evaluate a model on a set of images where each element has a different pixel set to 0.

Model Feature Attribution: When $|\mathcal{P}|$ is large even a model feature ablation can be impractical to evaluate by brute force. A *feature attribution* $\hat{r}(\mathbf{x})$ outputs a vector in $\mathbb{R}^J$ that induces an ordering on $\mathcal{P}$ given $\mathbf{x}$ and $\hat{f}_\theta$. A feature attribution $\hat{r}$ is similar to $\hat{\varepsilon}$ if it satisfies the following condition:

$$\begin{gathered}\hat{r}_j(\mathbf{x}) < \hat{r}_{j'}(\mathbf{x}) \iff \hat{\varepsilon}_j(\mathbf{x}) < \hat{\varepsilon}_{j'}(\mathbf{x}), \\ \forall \mathbf{x} \in \mathcal{X},\ \forall j, j' \in \{0, \dots, J-1\}.\end{gathered} \tag{3}$$

One example of a feature attribution method is the input gradient $\nabla_{\mathbf{x}} \log p(\hat{y}_i = y_i \mid \hat{f}_\theta(\mathbf{x}_i))$. An input gradient can be computed with a single forwards and backwards pass on the model, which is much cheaper when $J >> 2$. A feature attribution $\hat{r}$ that is similar to $\hat{\varepsilon}$ is sometimes called a *faithful* feature attribution. Note that attribution faithfulness (i.e. similarity between $\hat{r}$ and $\hat{\varepsilon}$) is an interaction between the attribution method and the model, whereas similarity between $\hat{\varepsilon}$ and ε is an interaction between the model and the ground-truth function.

Relating Ground-Truth Feature Ablations and Model Feature Attributions: if $\hat{\varepsilon}$ and ε are not similar, then a faithful model feature attribution will not inform us about ground-truth feature sensitivity, only model feature sensitivity. If $\hat{\varepsilon}$ and ε are similar but $\hat{r}$ is not similar to $\hat{\varepsilon}$, then even if the model is sensitive to the correct features we may fail to identify them. If both conditions are satisfied, then by transitivity $\hat{r}$ is similar to ε, and only then we can use $\hat{r}$ to draw conclusions about ε.

2.2 Common Feature Attribution Methods

We now introduce a range of methods from the literature which produce attributions $\hat{r}$ of varying degrees of similarity to $\hat{\varepsilon}$ when $\mathcal{P}(\mathbf{x}) = \{\mathbf{x} \odot \mathbf{e}_0, \dots, \mathbf{x} \odot \mathbf{e}_{d-1}\}$.

Occlusion/Feature Ablation. A computationally expensive approach to estimate a sample's feature attribution iteratively nulls out each input dimension and measures the change in function output [Zeiler and Fergus, 2014].

Input-Gradient (Grad) which is also termed logit-gradient or saliency map, ranks, by magnitude, the gradient of output with respect to input: $\nabla_{x_i} \hat{f}_\theta$ [Baehrens et al., 2010, Simonyan et al., 2013].

Gradient-based Alternatives. A challenge with the input-gradient is that a function's gradient can be zero at a point, but the function's output can still undergo a change when compared to the output at the baseline input, which is undesirable [Shrikumar et al., 2016]. We consider two approaches that address this issue: *integrated gradients (IG)* [Sundararajan et al., 2017], and *smoothgrad (SG)* [Smilkov et al., 2017]. IG is the sum the input-gradients of intermediate interpolants between the baseline and the original input: $(x_i - x^j_{i,b}) \int_{\alpha=0}^{1} \frac{\partial \hat{f}_\theta(x_i + \alpha(x_i - x^j_{i,b}))}{\partial x_i} \, d\alpha$. SG is the average of noisy input-gradients: $1/n \sum_{i=1}^{n} \nabla_{x_i+\epsilon} \hat{f}_\theta \left(x_i + \epsilon\right)$, where ϵ is the user-defined independent Gaussian noise.

Approximate Shapley. We also consider the gradient approximation to the Shapley values feature attributions [Lundberg and Lee, 2017]—a method that unifies additive feature attributions.

2.3 Common Model Regularization Methods

While there are too many regularization methods to cover exhaustively, here we introduce some of those most frequently used in literature. The effect of model regularization on $\hat{r}$ can be difficult to articulate because we must disentangle its effect on model smoothness (which influences the similarity between $\hat{r}$ and $\hat{\varepsilon}$ for many choices of attribution method) and its effect on model generalization (which influences the similarity between $\hat{\varepsilon}$ and ε). These effects are singularly difficult to disentangle because some degree of smoothness is often a necessary but insufficient condition for good generalization. We now discuss the regularization schemes that we study in this work:

- **Unregularized (UR)**: we take as baseline models obtained via unregularized maximum likelihood.
- **Dropout (DT)**: stochastically drops a fraction of neurons in training [Srivastava et al., 2014].
- **Weight Decay (WD)**: encourages a minimum norm solution via an L-2 norm penalty on model parameters [Krogh and Hertz, 1991, Bos and Chug, 1996, Gupta and Lam, 1998].
- **Additive Gaussian Noise (IN)**: corrupts inputs, additively, with Gaussian noise [An, 1996].
- **Adversarial training (AT)**: requires minimizing the task loss on a combination of normal inputs and worst-case adversarial inputs. We consider the fast gradient sign method (FGSM) [Goodfellow et al., 2014] and iterative projected gradient descent (PGD) [Madry et al., 2017].

3 Related Work

Evaluating the Faithfulness of Feature Attribution Methods. There has been significant scholarship on empirically assessing the faithfulness of feature attribution methods across various settings, including vision [Samek et al., 2016, Yeh et al., 2019, Hooker et al., 2018, Bhatt et al., 2020, Fel et al., 2022, Zhou et al., 2022, Denain and Steinhardt, 2022, Karimi et al., 2022], text classification [DeYoung et al., 2019], times series [Ismail et al., 2020], and backdoor detection [Casper et al., 2023]. The emerging consensus is that gradient-based attribution methods are unfaithful. However, Bastings et al. [2021] found that gradient-based attributions, for BERT models, become faithful. Yet, it is unclear what factor drives such improvement.

Adversarial Training and Perceptually Aligned Gradients. Beyond conferring robustness, adversarial training produces models whose input-gradients are *perceptually-aligned*, visually similar, to the input [Santurkar et al., 2019, Engstrom et al., 2019a,b]. Penalizing a model's input-gradient, during training, also improves robustness [Ross and Doshi-Velez, 2018]. To clarify the connection between input-gradient penalization and adversarial training, Chalasani et al. [2020] showed that, for integrated-gradients on a 1-hidden layer MLP, both are equivalent. However, the link between robustness and faithfulness remained unclear until Shah et al. [2021] showed that the gradient-based attributions of an adversarially trained model become faithful attributions. Here we show that these insights translate beyond vision models to those trained on protein and natural language sequences.

Recently, Srinivas et al. [2023] identified *off-manifold robustness*—the degree to which a model's output is invariant to off-manifold perturbations—as the key driver of perceptual alignment. Here, we seek schemes that confer faithfulness to a model's input-gradients without performance degradation.

Regularization and Stability of Feature Attributions. Beyond challenges with faithfulness, feature attribution methods are also unstable [Alvarez-Melis and Jaakkola, 2018, Ghorbani et al., 2019, Dombrowski et al., 2019, Anders et al., 2020], i.e., slight perturbations to the input, that do not change the model's output, result in large changes in the feature attribution. Dombrowski [2023] showed that regularization helps inmprove the stability of the feature attributions. Similarly, Srinivas et al. [2022] uses a soft-plus activation and constrained batch normalization layers to train low curvature models that exhibit improved robustness and feature attribution stability compared to unconstrained training.

4 Experimental Design

We now describe the setup that allows us to train models with pre-specified feature ablation rankings. A common design [Bastings et al., 2021, Zhou et al., 2022, Adebayo et al., 2020] ties an input's label to specific dimensions, and represents the other dimensions with high entropy noise. A function that successfully generalizes on this task, necessarily, relies on the pre-specified input dimensions.

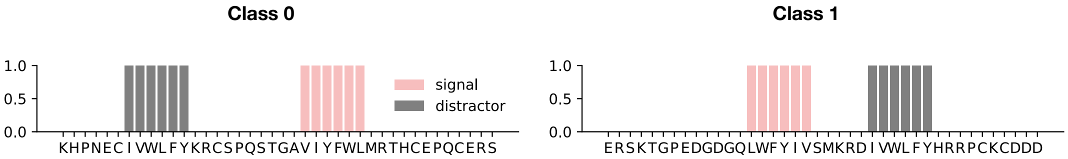

Figure 2: **Signal and Distractor decomposition for the Helix classification synthetic task.**

4.1 Design

Data Generating Process (DGP). We let an input be comprised of three groups of features that partition its dimensions: signal s_i, noise η_i, and distractor d_i so that $x_i \overset{\text{def}}{=} \texttt{concatenate}(s_i, \eta_i, d_i)$. We design the DGP so that a function estimated from data, that is derived from the DGP, has model feature ablation corresponding to the signal vector. A labelling function, $f(s_i)$, determines the label of x_i based, *solely*, on s_i. The noise dimensions are determined by high entroy noise that is independent of the label. Similarly, the distractor is a feature that is present in all inputs.

We generate an input x_i, via an oracle, $\mathcal{O}$, that uses masking functions to set the signal, noise, and distractor dimensions. The signal masking function, $\mathcal{M}_s : \vec{0} \mapsto \{0, 1\}^d$, maps an input into a binary vector that corresponds to the signal dimensions. The noise and distractor masking functions, $\mathcal{M}_\eta$ & $\mathcal{M}_d$, set the noise and distractor dimensions respectively. The oracle, $\mathcal{O}(\vec{0}, \mathcal{M}_s, \mathcal{M}_\eta, \mathcal{M}_d)$, maps the all zero vector to an x_i with signal, distractor, and noise tuple: (s_i, d_i, η_i). Using the oracle, we generate a training set: $\mathcal{D}_{\text{train}} \overset{\text{def}}{=} \{(x_i, s_i, d_i, \eta_i, y_i)\}_{i=1}^n$, and a test set: $\mathcal{D}_{\text{test}} \overset{\text{def}}{=} \{(x_i, s_i, d_i, \eta_i, y_i)\}_{i=1}^m$.

Model Verification. Given training data, we obtain an estimate, $\hat{f}_\theta$, of the labelling function, f, via regularized ERM. To ensure that the model's feature ablation corresponds to s_i, we require the function's performance to meet certain requirements: first, its test performance should be close to 100 percent. Second, on a modified test set, where the distractor dimensions have been ablated, the model's performance should remain unchanged.

4.2 Tasks

Protein Property Prediction. We instantiate the experimental design across three protein property prediction tasks. We predict protein properties from sequence only. Proteins are made up of (one of 22) chains of amino acid residues. Given a 40 residue long sequence, we consider a binary classification task that predicts the level of following properties:

1. **Helix Fraction:** amino acids in the protein sequence that are one of Valine (V), Isoleucine (I), Tyrosin (Y), Phenylalanine (F), Tryptophan (W), and Leucine (L);

2. **Turn fraction:** amino acids in the protein sequence that are one of Asparagine (N), Proline (P), Glycine (G), and Serine (S); and
3. **Sheet fraction:** amino acids in the protein sequence that are one of Glutamic acid (E), Methionine (M), Alanine (A), and Leucine (L).

- Signal, Noise, & Distractor: Across all settings, a sample belongs to a high proportion class if it contains a subsequence, of length 6 or greater, with residues that directly encode the output property of interest. For example, in the Helix fraction prediction task, if a sequence contains any subsequence, with length greater than 6, of consecutive residues from the set: {VIYFWL}, then we it is a high helix fraction sequence. Otherwise, it is a low helix fraction sequence. Here, the signal dimensions are the 6 or more consequence helix residue sequences for inputs that belong to the high helix fraction sequence. For each input, the noise dimensions are set to randomly sampled, non-helix fraction, amino acid residues. Across all samples, the distractor feature corresponds, also, to a 6 sequence long amino acid residue where the helix fraction tokens are interspersed with non-helix fraction tokens. We make analogous design decisions across all the other protein property prediction tasks.
- Metric: We measure attribution faithfulness with the `Precision@k` metric, which is the precision between the top-k ranked dimensions of an attribution vector and the pre-specified signal dimensions (length k) [Bastings et al., 2021].
- Models: We consider a SeqCNN and MLP model.

5 The Input-Gradient of an unregularized protein sequence model is not a faithful attribution

Overview and Question. Previous assessments of feature attributions mostly focused on the image and natural language settings [Shah et al., 2021, Hooker et al., 2018]. However, it is unclear whether insights from these modalities carry over to the protein property prediction setting. Here we ask whether the input-gradient of unregularized models is an unfaithful feature attribution for protein classification tasks. We train SeqCNN and MLP models on the synthetic protein classification tasks discussed in Section 4. To assess the faithfulness of a feature attribution method, we compare the output of common gradient-based feature attributions to the model's feature ablation and occlusion attribution. A high similarity between the output of a feature attribution method and the model feature ablation (and occlusion) indicates that the method is a faithful attribution; i.e., reflects the perturbation sensitivity behavior of the model.

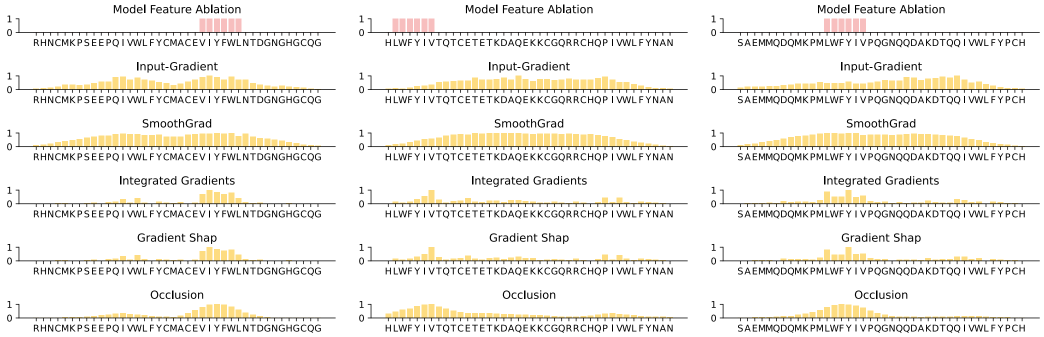

Figure 3: Comparing gradient-based feature attributions, for three samples, to the model feature ablation and occlusion attribution of an unregularized model.

Result. In Figure 3, we show three inputs, the expected model ablation for that input, and all the feature attribution methods that we consider. First, we find that the input-gradient and smoothgrad feature attribution methods are indeed unfaithful to the model; with low similarity to the model feature ablation. The input-gradient and smoothgrad attributions have mean average precision@k of 0.12, and 0.09 respectively when compared with the model feature ablation.

On the other hand, we find that integrated gradients, and gradient-shap—gradient approximation of the shapley value attribution—have a high similarity, 0.85, and 0.86 respectively, with the model feature ablation. These findings complement those of Bastings et al. [2021], who also find that integrated gradients attributions are faithful for BERT and LSTM models trained for natural language classification tasks. A possible explanation for the effectiveness of integrated-gradients and gradient-shap is the incorporation of a baseline. Both approaches 'compare' (and sum) input-gradient attribution of a input to that of a baseline.

6 The Effect of Regularization on the Faithfulness of Feature Attribution

Overview & Experimental Setup. Using the experimental protocol setup in §4, we now assess the impact of various regularization schemes on the faithfulness of the resulting model's feature attributions. Specifically, given a regularization scheme, and a model architecture, we follow the experimental protocol to obtain datasets and models for that setting. To assess faithfulness, we then compare the attributions to the model feature ablation. For each setting, we train 5 models and compute the precision metric for 50 samples in the test set. We report the average precision score under each setting. We present a summary of these average precision scores in Table 1.

		Helix				Turn				Sheet			
		Grad	SGrad	IG	GShap	Grad	SGrad	IG	GShap	Grad	SGrad	IG	GShap
SeqCNN	UNREGULARIZED	.41	.44	.86	.87	.43	.42	.83	.81	.45	.47	.83	.84
	DROPOUT	.39	.86	.89	.85	.32	.33	.81	.88	.32	.33	.81	.84
	WEIGHT DECAY	.45	.85	.86	.57	.52	.51	.88	.87	.50	.50	.86	.87
	NOISE-LOW	.81	.80	.85	.83.	.82	.77	.84	.85	.79	.84	.87	.86
	NOISE-MED	.81	.99	.85	.83.	.82	.77	.84	.85	.79	.84	.87	.86
	ADV-LOW	.99	.99	.89	.88	.99	.99	.89	.91	.99	.99	.89	.89
	ADV-MED	.99	.99	.89	.88	.99	.99	.89	.91	.99	.99	.89	.89

Table 1: Mean Precision@k similarity for the feature attribution methods across different regularization schemes on the three synthetic tasks. We find that similar to adversarial training, independent Gaussian noise corruption, during training, improves the faithfulness of a model's input-gradient attribution. On the contrary, we find that weight decay, and dropout do not confer faithfulness benefits.

The Impact of Regularization. First, we seek to identify regularization schemes, beyond adversarial training, that also confer faithfulness onto the feature attributions of the resulting model. We now take each regularization scheme in turn, and discuss the key results:

- **Unregularized:** Across all tasks and settings, we find that the standard input-gradient, and smoothgrad feature attribution methods achieve a low precision score (below 0.5). Similar to previous findings [Shah et al., 2021, Zhou et al., 2022, Bordt et al., 2022], the input-gradient and smoothgrad attributions of unregularized protein sequence models are not faithful to the model. However, contrary to Shah et al. [2021]'s finding in the image setting, we find that integrated gradients and gradient-shap attributions are faithful attributions even for the unregularized model.
- **Weight Decay & Dropout:** Similar to unregularized models, for both weight decay and dropout, we do not observe any meaningful improvement in the faithfulness of the input-gradients and smoothgrad attributions of resulting models. However, integrated gradients and gradient-shap remain effective.
- **Independent Additive Gaussian Noise:** Different from other regularization schemes, we observe a distinctive improvement, in precision scores across feature attribution methods, sometimes rivaling adversarial training. We observe that the feature attributions of models derived from Gaussian noise corruption no longer indicate high relevance for the distractor dimensions. In addition, we do not observe significant performance degradation from noise addition; therefore, it could serve as a replacement for adversarial training.
- **Adversarial Training:** As expected and previously observed Shah et al. [2021], we find that adversarial training results in models whose gradients are indeed sensitive to the model output. Across all training schemes tested, the gradient-based feature attributions of adversarially models have the highest precision score (greater than 0.95). Surprisingly, even relatively low values of $\epsilon = 0.01$, was still effective at conferring task discriminativity.

Taken together, we can find that independent Gaussian additive noise corruption, during training, is an effective alternative to adversarial training for improving the faithfulness of a model's gradients. To a lesser extent, we also find that input masking and joint generative & classifier training also improves a model's gradient sensitivity to the model output. On the other hand, we observe no meaningful improvement from dropout and weight decay regularization.

Limited effect of model architecture. We do not observe any systematic differences due to the underlying model architecture. This finding might suggest that the choice of the regularization scheme and feature attribution method is more important than the architecture for the protein sequence setting.

7 Faithful Feature Attributions on real-world Protein-Property Prediction Tasks

In previous section, we identified independent Gaussian noise addition, during training, as a viable alternative to adversarial training for improving the faithfulness of feature attributions. Here we train regularized models for a protein property prediction task—protein binding affinity—and show that regularization leads to faithful feature attributions on real-world biological tasks. To understand the effect of sequence-level mutations on binding affinity, we train models to predict whether two proteins will bind, on a collection of 36 thousand sequences, publicly available from Mason et al. [2021]. These sequences are mutated variants of the hu4D5 protein.

Overview and Question. A fundamental property of proteins is their ability to bind to one another. However, whether two proteins will bind is a challenging task to predict. Towards addressing this challenge, we train high performing models for the protein-protein binding task. We then investigate the feature attributions of these models. Training on aligned protein sequences is a popular technique [Rao et al., 2021, Gruver et al., 2023, Frey et al., 2023] that introduces a strong evolutionary and structural prior into discriminative and generative modeling of proteins. Here, we align antibodies according to the AHo [Honegger and PluÈckthun, 2001] alignment scheme. This leads to better model performance and explainability, as it is possible to recognize standard regions of antibodies in the aligned sequences. However, the alignment tokens ("-") can also be thought of as distractor features, which do not encode for the label.

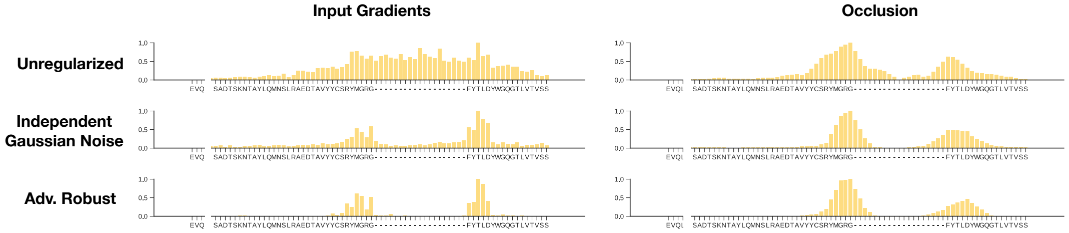

Figure 4: **The effect of regularization on the faithfulness of the model's input-gradients.**

Result. We see (Figure 4) that input gradients of unregularized models assign significant weight across the alignment tokens, while regularized models are invariant to the distractor signal. We conjecture that the alignment tokens allow the model to identify the regions and tokens that are actually task discriminative (improving model performance), while regularization ensures that feature attributions are faithful. The results in Fig. 4 extend our findings from the synthetic, toy task above to a real-world biological task.

8 Conclusion

We investigate the effect of regularization on the faithfulness of the resulting model's feature attributions, particularly for model trained for protein property prediction tasks. Specifically, we seek alternatives to adversarial training that confer faithfulness onto a model's gradient-based feature attributions. To address this challenge, we designed controlled classification tasks to quantify the effect of regularization on the faithfulness of a model's gradient-based feature attribution. We find that independent Gaussian noise input-corruption, during training, confers similar benefits as adversarial training on a model's input-gradients. Finally, we translate these results to a real-world protein function, binding affinity, prediction task where the gradient-based feature attributions of noise-regularized models correctly indicate low sensitivity to irrelevant gap tokens in the input protein's sequence alignment.

References

Julius Adebayo, Michael Muelly, Ilaria Liccardi, and Been Kim. Debugging tests for model explanations. *arXiv preprint arXiv:2011.05429*, 2020. 5

David Alvarez-Melis and Tommi S Jaakkola. On the robustness of interpretability methods. *arXiv preprint arXiv:1806.08049*, 2018. 5

Guozhong An. The effects of adding noise during backpropagation training on a generalization performance. *Neural computation*, 8(3):643–674, 1996. 4, 12

Christopher Anders, Plamen Pasliev, Ann-Kathrin Dombrowski, Klaus-Robert Müller, and Pan Kessel. Fairwashing explanations with off-manifold detergent. In *International Conference on Machine Learning*, pages 314–323. PMLR, 2020. 5

David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. How to explain individual classification decisions. *The Journal of Machine Learning Research*, 11:1803–1831, 2010. 4

Pierre Baldi and Peter J Sadowski. Understanding dropout. *Advances in neural information processing systems*, 26, 2013. 12

Jasmijn Bastings, Sebastian Ebert, Polina Zablotskaia, Anders Sandholm, and Katja Filippova. " will you find these shortcuts?" a protocol for evaluating the faithfulness of input salience methods for text classification. *arXiv preprint arXiv:2111.07367*, 2021. 4, 5, 6, 7

Umang Bhatt, Adrian Weller, and José MF Moura. Evaluating and aggregating feature-based model explanations. *arXiv preprint arXiv:2005.00631*, 2020. 4

Sebastian Bordt, Uddeshya Upadhyay, Zeynep Akata, and Ulrike von Luxburg. The manifold hypothesis for gradient-based explanations. *arXiv preprint arXiv:2206.07387*, 2022. 7

Siegfried Bos and E Chug. Using weight decay to optimize the generalization ability of a perceptron. In *Proceedings of International Conference on Neural Networks (ICNN'96)*, volume 1, pages 241–246. IEEE, 1996. 4, 12

Nicholas Carlini, Florian Tramer, Krishnamurthy Dj Dvijotham, Leslie Rice, Mingjie Sun, and J Zico Kolter. (certified!!) adversarial robustness for free! *arXiv preprint arXiv:2206.10550*, 2022. 1

Stephen Casper, Yuxiao Li, Jiawei Li, Tong Bu, Kevin Zhang, and Dylan Hadfield-Menell. Benchmarking interpretability tools for deep neural networks. *arXiv preprint arXiv:2302.10894*, 2023. 4

Prasad Chalasani, Jiefeng Chen, Amrita Roy Chowdhury, Xi Wu, and Somesh Jha. Concise explanations of neural networks using adversarial training. In *International Conference on Machine Learning*, pages 1383–1391. PMLR, 2020. 4

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *international conference on machine learning*, pages 1310–1320. PMLR, 2019. 12

Jean-Stanislas Denain and Jacob Steinhardt. Auditing visualizations: Transparency methods struggle to detect anomalous behavior. *arXiv preprint arXiv:2206.13498*, 2022. 4

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. Eraser: A benchmark to evaluate rationalized nlp models. *arXiv preprint arXiv:1911.03429*, 2019. 4

Ann-Kathrin Dombrowski. A geometrical perspective on explanations for deep neural networks. 2023. 5

Ann-Kathrin Dombrowski, Maximillian Alber, Christopher Anders, Marcel Ackermann, Klaus-Robert Müller, and Pan Kessel. Explanations can be manipulated and geometry is to blame. *Advances in neural information processing systems*, 32, 2019. 5

Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Brandon Tran, and Aleksander Madry. Adversarial robustness as a prior for learned representations. *arXiv preprint arXiv:1906.00945*, 2019a. 4

Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Brandon Tran, and Aleksander Madry. Learning perceptually-aligned representations via adversarial robustness. *arXiv preprint arXiv:1906.00945*, 2(3):5, 2019b. 4

Thomas Fel, David Vigouroux, Rémi Cadène, and Thomas Serre. How good is your explanation? algorithmic stability measures to assess the quality of explanations for deep neural networks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 720–730, 2022. 4

Nathan C Frey, Daniel Berenberg, Karina Zadorozhny, Joseph Kleinhenz, Julien Lafrance-Vanasse, Isidro Hotzel, Yan Wu, Stephen Ra, Richard Bonneau, Kyunghyun Cho, et al. Protein discovery with discrete walk-jump sampling. *arXiv preprint arXiv:2306.12360*, 2023. 8

Amirata Ghorbani, Abubakar Abid, and James Zou. Interpretation of neural networks is fragile. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3681–3688, 2019. 5

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 4, 12

Nate Gruver, Samuel Stanton, Nathan C Frey, Tim GJ Rudner, Isidro Hotzel, Julien Lafrance-Vanasse, Arvind Rajpal, Kyunghyun Cho, and Andrew Gordon Wilson. Protein design with guided discrete diffusion. *arXiv preprint arXiv:2305.20009*, 2023. 8

Amit Gupta and Siuwa M Lam. Weight decay backpropagation for noisy data. *Neural networks*, 11 (6):1127–1138, 1998. 4, 12

Annemarie Honegger and Andreas PluÈckthun. Yet another numbering scheme for immunoglobulin variable domains: an automatic modeling and analysis tool. *Journal of molecular biology*, 309(3): 657–670, 2001. 8

Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. Evaluating feature importance estimates. 2018. 1, 4, 6

Aya Abdelsalam Ismail, Mohamed Gunady, Hector Corrada Bravo, and Soheil Feizi. Benchmarking deep learning interpretability in time series predictions. *Advances in neural information processing systems*, 33:6441–6452, 2020. 4

Amir-Hossein Karimi, Krikamol Muandet, Simon Kornblith, Bernhard Schölkopf, and Been Kim. On the relationship between explanation and prediction: A causal view. *arXiv preprint arXiv:2212.06925*, 2022. 4

Anders Krogh and John Hertz. A simple weight decay can improve generalization. *Advances in neural information processing systems*, 4, 1991. 4

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017. 4

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 4, 12

Derek M Mason, Simon Friedensohn, Cédric R Weber, Christian Jordi, Bastian Wagner, Simon M Meng, Roy A Ehling, Lucia Bonati, Jan Dahinden, Pablo Gainza, et al. Optimization of therapeutic antibodies by predicting antigen specificity from antibody sequence via deep learning. *Nature Biomedical Engineering*, 5(6):600–612, 2021. 8

Roshan M Rao, Jason Liu, Robert Verkuil, Joshua Meier, John Canny, Pieter Abbeel, Tom Sercu, and Alexander Rives. Msa transformer. In *International Conference on Machine Learning*, pages 8844–8856. PMLR, 2021. 8

Andrew Ross and Finale Doshi-Velez. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 4

Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE transactions on neural networks and learning systems*, 28(11):2660–2673, 2016. 4

Shibani Santurkar, Andrew Ilyas, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Image synthesis with a single (robust) classifier. *Advances in Neural Information Processing Systems*, 32, 2019. 4

Harshay Shah, Prateek Jain, and Praneeth Netrapalli. Do input gradients highlight discriminative features? *Advances in Neural Information Processing Systems*, 34:2046–2059, 2021. 1, 4, 6, 7

Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*, 2016. 4

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013. 4

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017. 4

Suraj Srinivas, Kyle Matoba, Himabindu Lakkaraju, and François Fleuret. Efficient training of low-curvature neural networks. *Advances in Neural Information Processing Systems*, 35:25951–25964, 2022. 5

Suraj Srinivas, Sebastian Bordt, and Hima Lakkaraju. Which models have perceptually-aligned gradients? an explanation via off-manifold robustness. *arXiv preprint arXiv:2305.19101*, 2023. 5

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014. 4, 12

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017. 4

Greg Yang, Tony Duan, J Edward Hu, Hadi Salman, Ilya Razenshteyn, and Jerry Li. Randomized smoothing of all shapes and sizes. In *International Conference on Machine Learning*, pages 10693–10705. PMLR, 2020. 12

Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Suggala, David I Inouye, and Pradeep K Ravikumar. On the (in) fidelity and sensitivity of explanations. *Advances in Neural Information Processing Systems*, 32, 2019. 4

Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*, pages 818–833. Springer, 2014. 4

Yilun Zhou, Serena Booth, Marco Tulio Ribeiro, and Julie Shah. Do feature attribution methods correctly attribute features? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 9623–9633, 2022. 4, 5, 7

A Models, Compute, & Architecture

We consider three different types of architectures for the models considered in this work.

CNN. We use a 4 layer CNN where the first two layers are followed by a ReLU activation, and a pooling operation. After these two layers, we follow with two fully-connected layers with dropout is optionally applied.

MLP: This model consists of four fully connected layers.

Optimizer. Across all models we use the AdamW Optimizer.

Loss Function. All the tasks we consider in this work are binary classification tasks, so we default to the cross-entropy loss function.

Compute. All of this work was performed on a single GPU (A-100) available via a SLURM compute cluster.

B Regularization Scheme(s)

We now discuss the various regularization schemes that we consider in this work. For each regularization type, we train models under different 3 hyper-parameter settings that correspond to varying the regularization strength. Out of the three settings, we pick the the largest setting for which we still obtain perfect performance on the test test.

- **Unregularized:** Here, for each model architecture, we simply turn off any form of data augmentation and regularization.
- **Dropout**: involves stochastically zeroing a selection of neurons during training Srivastava et al. [2014], Baldi and Sadowski [2013]. We vary the dropout hyper-parameters for three different settings: [0.25, 0.5, 0.75].
- **Weight Decay**: correspond to an L-2 norm penalty on model parameters during training Gupta and Lam [1998], Bos and Chug [1996]. We consider the range: [0.0001, 0.1]. We found that a weight decay penalty above 0.1 hurt the model performance.
- **Independent Gaussian Corruption** corrupts training inputs with independent additive Gaussian noise during. Gaussian noise addition helps increase smoothness [An, 1996], hence improving generalization, and robustness to adversarial examples [Cohen et al., 2019, Yang et al., 2020]. We take as the corruption zero-mean Gaussian noise where we vary the covariance for three settings: [0.01, 0.1, 0.5].
- **Adversarial Training** requires minimizing the task loss on a combination of normal inputs and worst-case adversarial inputs. We adopt two popular schemes based on the fast gradient sign method (FGSM) [Goodfellow et al., 2014] and iterative projected gradient descent (PGD) [Madry et al., 2017]. On the synthetic task, the maximal perturbation that still allowed 100 percent test performance was 0.09. We did not observe any gains from using PGD instead of FGSM, so we report results only for FGSM.

C Additional Visualization and Results

Similar to Figure 4 in the main draft, we now present additional samples for the CNN, MLP, and Transformer models. We show additional examples for a CNN model, and MLP model.

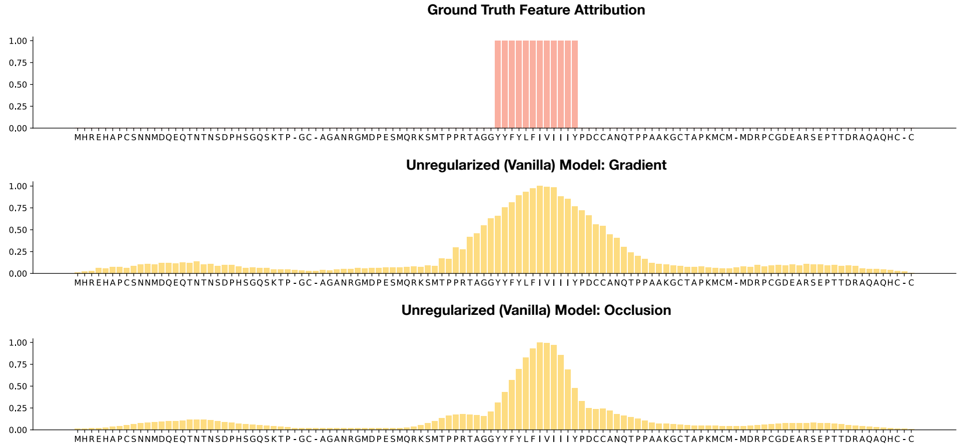

Figure 5: Gradient and Occlusion feature attributions for an unregularized model trained on the toy dataset without a distractor signal. We observe that the gradient and occlusion attributions highlight the key residues that are responsible for the model's output prediction.

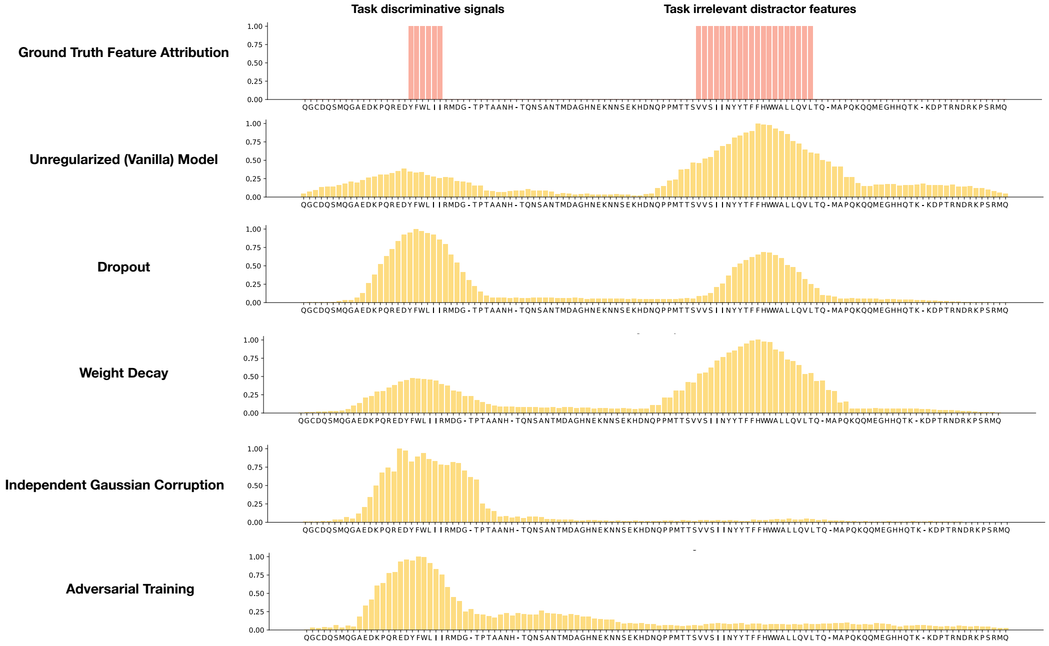

Figure 6: **CNN Model.** Input-Gradient attributions across models trained on various regularization schemes for a sample on the test set.

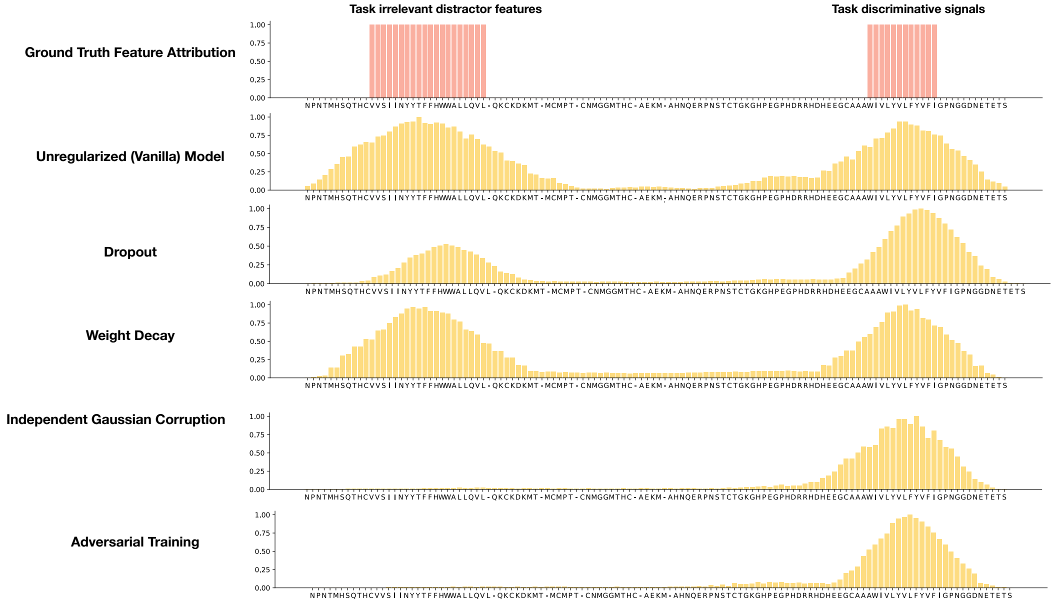

Figure 7: **CNN Model.** Input-Gradient attributions across models trained on various regularization schemes for a sample on the test set.

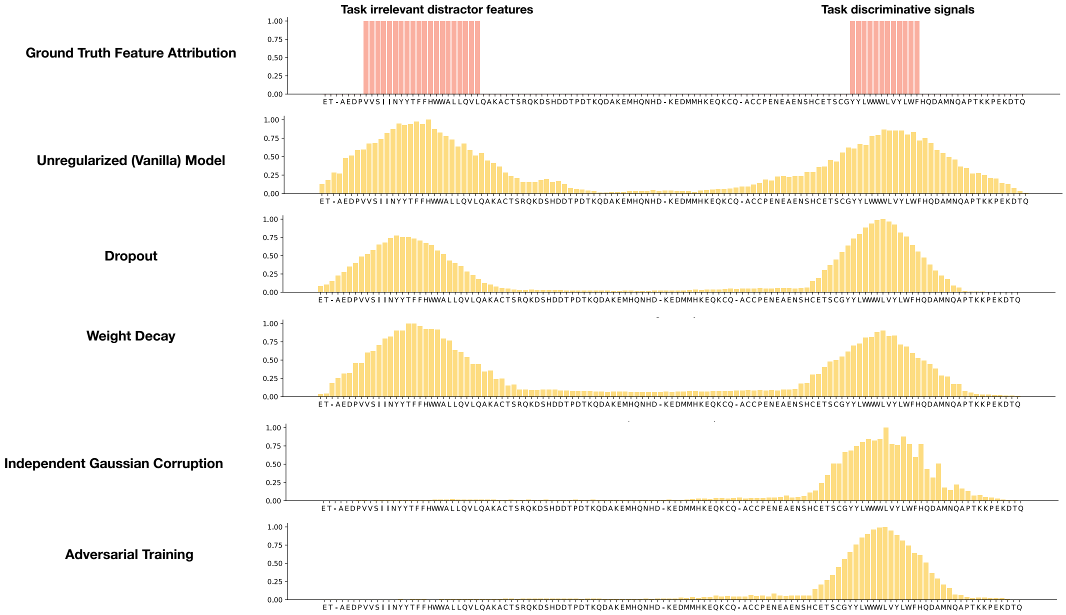

Figure 8: **MLP Model.** Input-Gradient attributions across models trained on various regularization schemes for a sample on the test set.

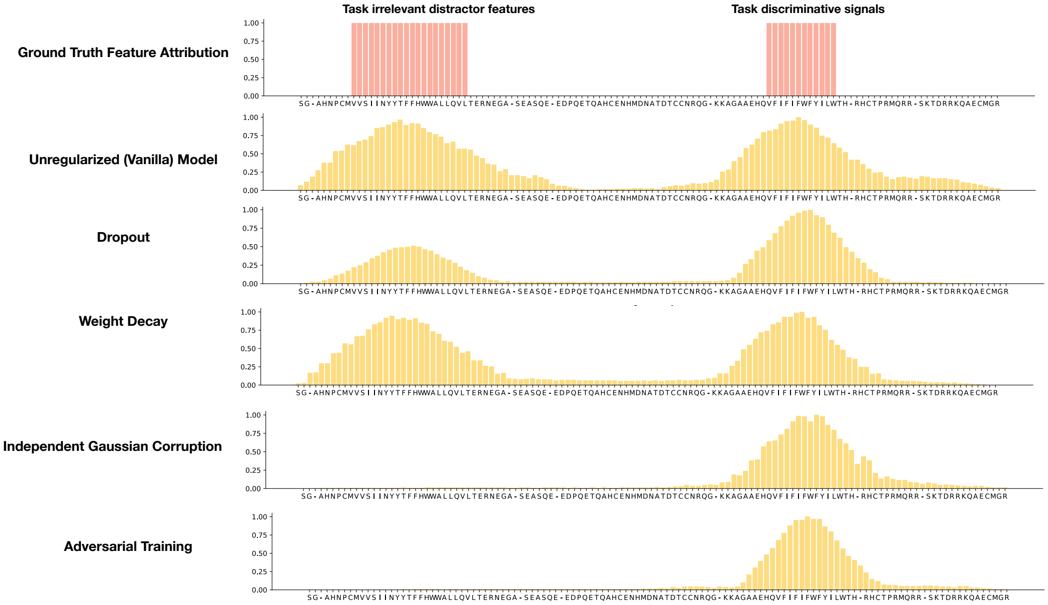

Figure 9: **MLP Model.** Input-Gradient attributions across models trained on various regularization schemes for a sample on the test set.