Towards Faithful Explanations for Text Classification with Robustness Improvement and Explanation Guided Training

Anonymous ACL submission

Abstract

Feature attribution methods highlight the important input tokens as explanations to model predictions, which have been widely applied to deep neural networks towards trustworthy AI. However, recent works show that explanations provided by these methods face challenges of being faithful and robust. In this paper, we propose a method with **R**obustness improvement and **E**xplanation **G**uided training towards more faithful **EX**planations (**REGEX**). First, we improve model robustness by input gradient regularization technique and virtual adversarial training. Secondly, we use salient ranking to mask noisy tokens and maximize the similarity between model attention and feature attribution, which can be seen as a self-training procedure without importing other external information. We conduct extensive experiments on six datasets with five attribution methods, and also evaluate the faithfulness in the out-of-domain setting. The results show that REGEX improves the fidelity metrics of the explanation in all settings. Moreover, we show that using highlight explanations produced by REGEX to train select-then-predict models results in comparable task performance to the end-to-end method. These findings provide evidence that we can improve the faithfulness of highlight explanations by considering model robustness. [1]

1 Introduction

As broad adoption of Pre-trained Language Models (PLMs) (Devlin et al., 2019; Liu et al., 2019; Raffel et al., 2020) requires humans to trust their output, we need to be able to understand the rationale behind the output and even ask questions regarding how the model comes to its decision (Lipton, 2018). Recently, explanation methods for interpreting why a model makes certain decisions are proposed and become more crucial. For example, feature attribution methods assign scores to tokens and highlight the important ones as explanations (Sundararajan et al., 2017; Jain et al., 2020; DeYoung et al., 2020).

[1] We will publicly release our code.

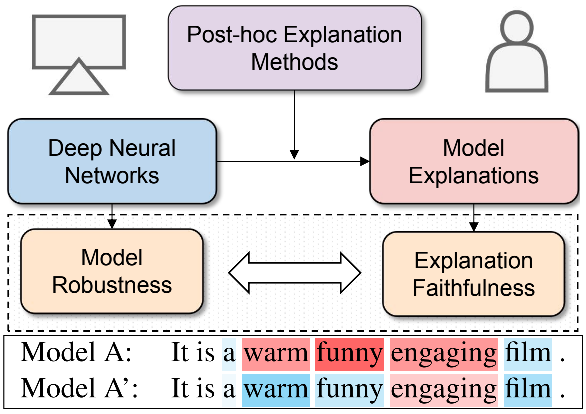


Figure 1: Visualization of positive and negative highlights produced by post-hoc explanation methods (e.g., feature attribution). However, these explanations suffer from unfaithfulness problems and can be further fooled by adversarial manipulation without changing model output (Ghorbani et al., 2019).

However, recent studies show that these explanations face challenges of being faithful and robust (Yeh et al., 2019; Sinha et al., 2021; Ivankay et al., 2022), illustrated in Figure 1. The *faithfulness* means the explanation accurately represents the reasoning behind model predictions (Jacovi and Goldberg, 2020). Though some works are proposed to use higher-order gradient information (Smilkov et al., 2017), by incorporating game-theoretic notions (Hsieh et al., 2021) and learning from priors (Chrysostomou and Aletras, 2021a), how to improve the faithfulness of highlight explanations still remains an open research problem. Moreover, the explanation should be stable between functionally equivalent models trained from different initializations (Zafar et al., 2021). Intuitively, the potential causes of these challenges could be (i) the model is not robust and mostly leads to unfaithful and fragile explanations (Alvarez-Melis and Jaakkola, 2018; Li et al., 2022) and (ii) those explanation methods themselves also lack robustness to imperceptible perturbations of the input (Ghorbani et al., 2019); hence we need to develop better explanation methods. In this paper, we focus on the former and argue that there are connections between model robustness and explainability, any progress in one part may represent progress in both.

To this end, we propose a method with **R**obustness improvement and **E**xplanation **G**uided training to improve the faithfulness of **EX**planations (**REGEX**) while also preserving the task performance for text classification. First, we apply the input gradient regularization technique and virtual adversarial training to improve model robustness. While previous works found that these mechanisms can improve the adversarial robustness and interpretability of deep neural networks (Ross and Doshi-Velez, 2018; Li et al., 2022), to the best of our knowledge, the faithfulness of model explanations by applying them has not been explored. Secondly, our method leverages token attributions aggregated by the explanation method, which provides a local linear approximation of the model's behavior (Baehrens et al., 2010). We mask input tokens with low feature attribution scores to generate perturbed text and then maximize the similarity between new attention and attribution scores. Moreover, we minimize the Kullback–Leibler (KL) divergence between model attention of original input and attributions. The main idea is to allow attention distribution of the model to learn from input importance during training to reduce the effect of noisy information.

To verify the effectiveness of REGEX, we consider a variety of classification tasks across six datasets with five attribution methods. Additionally, we conduct extensive empirical studies to examine the faithfulness of five feature attribution approaches in out-of-domain settings. The results show that REGEX improves the faithfulness of the highlight explanations measured by sufficiency and comprehensiveness (DeYoung et al., 2020) in all settings while outperforms or performs comparably to the baseline. Moreover, we show that using the explanations output from REGEX to train select-then-predict models results in comparable task performance to the end-to-end method, where the former trains an independent classifier using only the rationales extracted by the pre-trained extractor (Jain et al., 2020). These findings provide empirical support that improving model robustness can provide more faithful explanations produced by feature attribution methods – one desideratum in trustworthy AI.

2 Related Work

Model Robustness and Explainability As it has recently been shown that deep neural networks are vulnerable to adversarial attacks even with PLMs, several works are proposed to ensure that AI systems are trustworthy and reliable, which include quantifying the vulnerability and designing new attacks and better defense technologies (Hendrycks et al., 2020; Wang et al., 2021). For example, adversarial training-based methods are proposed to improve the robustness of the model and have been proven effective in various tasks (Goodfellow et al., 2015; Madry et al., 2018). However, as the debug tools for black-box models, explanation methods also lack robustness to imperceptible and targeted perturbations of the input (Heo et al., 2019; Camburu et al., 2019; Meister et al., 2021; Hsieh et al., 2021). While significantly different explanations are provided for similar models (Zafar et al., 2021), how to elicit more reliable explanations is a promising direction towards interpretation robustness.

Explanation Faithfulness The faithfulness of explanations is important for NLP tasks, especially when humans refer to model decisions (Kindermans et al., 2017; Girardi et al., 2018). Jacovi and Goldberg (2020) firstly propose to evaluate the faithfulness of NLP methods by separating the two definitions between faithfulness and plausibility and provide guidelines on how evaluation of explanations methods should be conducted. Recently, some works focus on faithfulness measurements of NLP model explanations and improve the faithfulness of specific explanations (Wiegreffe et al., 2021; Yin et al., 2021; Bastings et al., 2021; Chrysostomou and Aletras, 2021b). Among them, Ding and Koehn (2021) propose two specific consistency tests intending to measure if the post-hoc explanations remain consistent with similar models.

Incorporate Explanations into Learning While most previous explanation methods have been developed for explaining deep neural networks, some works also explore the potential to leverage these explanations to help build better models (Liu and Avci, 2019; Rieger et al., 2020; Jayaram and Allaway, 2021; Ju et al., 2021; Bhat et al., 2021; Han

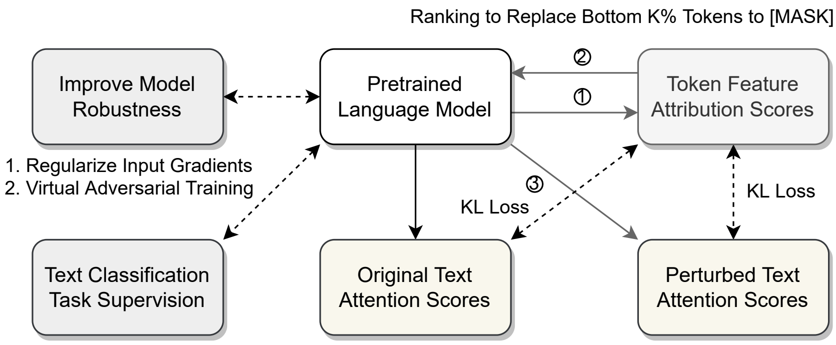


Figure 2: The overall framework of proposed REGEX method. REGEX consists of two components for robustness improvement and explanations guided training respectively. For latter, we iteratively mask input tokens with low attribution scores and then minimize the KL divergence between attention of masked input and feature attributions.

and Tsvetkov, 2021; Ismail et al., 2021; Chrysostomou and Aletras, 2021a; Stacey et al., 2022; Ye and Durrett, 2022). Hase and Bansal (2021) propose a framework to understand the role of explanations in learning, and find that explanations are suitably used in a retrieval-based modeling approach. Similarly, Adebayo et al. (2022) investigate whether post-hoc explanations are effective for detecting model reliance on spurious training signals but the answer seems to be negative. While how to effectively incorporate explanations into learning remains an open problem, we focus on incorporating model explanations in a self-training way to improve its faithfulness.

3 Method

3.1 Problem Formulation

First, we consider the setting of multi-label text classification problem with n input examples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$. The input space embedded into vectors is $\boldsymbol{x} \subseteq \mathbb{R}^{l \times d}$ and the output space is $\mathcal{Y}$. A neural classifier is $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ where $f_\theta(\boldsymbol{x})$ parameterized by θ which denotes the output class for one example $\boldsymbol{x} = (x_1, \ \cdots, x_l) \in \mathcal{X}$, where l represents the length of the sequence. The optimization of the network is to minimize the cross-entropy loss $\mathcal{L}$ over the training set as follows:

$$\mathcal{L}_{classify} = \sum_{i=1}^{n} \log p_\theta(y_i | \boldsymbol{x}_i). \tag{1}$$

Then, given an input $\boldsymbol{x}_i = (x_1, \ \cdots, x_l)$ and its particular prediction $f_\theta(\boldsymbol{x}_i) = y_i$, the goal of feature attribution is to assign each token with a normalized score that then can be used to extract a compact set of relevant sub-sequences with respect to the prediction. Formally, an attribution of the prediction at input $\boldsymbol{x}_i$ is a vector $\boldsymbol{a}_i = (a_{i1}, \ \cdots, a_{il})$ and a_{ij} is defined as the attribution of x_{ij}. After that, we denote the set of extracted tokens (i.e., highlight explanations or rationales) provided by taking top-k values from $\boldsymbol{x}_i$ as $\boldsymbol{r}_i$, and use $\overline{\boldsymbol{r}}_i = \boldsymbol{x}_i \setminus \boldsymbol{r}_i$, as the complementary set of $\boldsymbol{r}_i$ to denote the set of irrelevant tokens.

3.2 Robustness Improvement

Adversarial attacks are inputs that are intentionally constructed to mislead neural networks (Szegedy et al., 2013; Goodfellow et al., 2015). Given the f_θ and an input $\boldsymbol{x} \in \mathcal{X}$ with the label $y \in \mathcal{Y}$, an adversarial example $\boldsymbol{x}_{adv}$ satisfies

$$f_\theta(\boldsymbol{x}_{adv}) \neq y \quad s.t. \quad \|\boldsymbol{x} - \boldsymbol{x}_{adv}\| \leq \epsilon, \tag{2}$$

where $\| \cdot \|$ is the distance metric and generally measured by the ℓ_p-norm. To address these attack problems, several defense methods have been proposed to increase the robustness of deep neural networks to adversarial attacks. We adopt two popular methods: *virtual adversarial training* (Miyato et al., 2015) which leverages a regularization loss to promote the smoothness of the model distribution, and *input gradient regularization* (Ross and Doshi-Velez, 2018) which regularizes the gradient of the cross-entropy loss.

As shown in Figure 2, we aim to improve the robustness of deep neural networks intrinsically. Instead of adopting adversarial training objective, we follow Jiang et al. (2019) to regularize the standard objective using virtual adversarial training (Miyato et al., 2018):

$$\mathcal{L}_{at}(\boldsymbol{x}, y, \theta) = \max_{\delta} l(f(\boldsymbol{x} + \delta; \theta), f(\boldsymbol{x}; \theta)). \tag{3}$$

The goal of this approach is the enhancement of label smoothness in the embedding neighborhood. Specially, we run C projected gradient steps to find the perturbation δ with violation of local smoothness to maximize the adversarial loss. On the other hand, input gradient regularization trains neural networks by minimizing not just the "energy" of the network but the rate of change of that energy with respect to the input features (Drucker and LeCun, 1992). The goal of this approach is to ensure that if any input changes slightly, the KL divergence between the predictions and the labels will not change significantly. Formally, it takes the original loss term and penalizes the ℓ_2 norm of its gradient and parameters:

$$\mathcal{L}_{gr}(\boldsymbol{x}, y, \theta) = \|\frac{\partial}{\partial \boldsymbol{x}} \mathcal{L}(\boldsymbol{x}, y, \theta)\|_2 + \|\theta\|_2. \quad (4)$$

It can also be interpreted as applying a particular projection to the Jacobian of the logits and regularizing it (Ross and Doshi-Velez, 2018).

3.3 Explanation Guided Training

The main idea of Explanation Guided Training (EGT) is that the irrelevant tokens should have low feature attribution scores if post-hoc explanations faithfully quantify the model predictions. Hence, we propose to leverage the existing interpretations to guide the model for reducing feature attribution scores of irrelevant tokens without sacrificing the model performance. Inspire by Ismail et al. (2021) that uses the Saliency method (i.e., gradient of the target class with respect to the input) (Simonyan et al., 2014), we firstly apply the *Integrated Gradients* (IG) method (Sundararajan et al., 2017) that is more faithful via axiomatic proofs to calculate the token importance. It integrates the gradient along the path from an uninformative baseline to the original input. This baseline input is used to make a high-entropy prediction that represents uncertainty. As it takes a straight path between baseline and input, it requires computing gradients several times. The motivation for using path integral rather than vanilla gradient is that the gradient might have been saturated around the input while the former can alleviate this problem. Formally, given an input $\boldsymbol{x}$ and baseline $\boldsymbol{x'}$, the integrated gradient along the i^{th} dimension is defined as follows:

$$IG_i(\boldsymbol{x}, \boldsymbol{x'}) ::= (x_i - x'_i) \int_{\alpha=0}^{1} \frac{\partial f_\theta(\boldsymbol{x'} + \alpha \times (\boldsymbol{x} - \boldsymbol{x'}))}{\partial x_i} \, d\alpha, \quad (5)$$

where $\frac{\partial f_\theta(\boldsymbol{x})}{\partial x_i}$ represents the gradient of f along the i^{th} dimension at $\boldsymbol{x}$ which is the concatenated embedding of the input sequence, and the attribution of each token is the sum of the attributions of its embedding. Note that we attribute the output of the model with ground-truth labels during training. We also test other feature attribution methods in §6.2.

After calculating the token's importance score by ℓ_2 aggregation over embedding dimensions, we sort tokens of $\boldsymbol{x}$ based on these scores and mask the bottom $K\%$ words according to that sorting. We define the sorting function as $\mathbf{s}(\cdot)$ and the masking function as $\mathbf{m}(\cdot)$. For example, $\mathbf{s}_i(\boldsymbol{x})$ is the i^{th} smallest element in $\boldsymbol{x}$, and $\mathbf{m}_k(\mathbf{s}(\boldsymbol{x}), \boldsymbol{x})$ replaces all $x_i \in \{\mathbf{s}_i(\boldsymbol{x})\}_{e=0}^{ceil(1,Kl)}$ with a mask distribution, i.e., $\mathbf{m}_k(\mathbf{s}(\boldsymbol{x}), \boldsymbol{x})$ removes the $K\%$ lowest features from $\boldsymbol{x}$ based on the order provided by $\mathbf{s}(\boldsymbol{x})$. During training, we generate a new input $\widetilde{\boldsymbol{x}}$ for each example $\boldsymbol{x}$ by masking the features with low attribution scores as follows:

$$\widetilde{\boldsymbol{x}} = \mathbf{m}_k(\mathbf{s}_{IG}(\boldsymbol{x}), \boldsymbol{x}). \quad (6)$$

$\widetilde{\boldsymbol{x}}$ is then passed through the network which results in an attention scores $att(\widetilde{\boldsymbol{x}})$. Following Jain et al. (2020), the attention scores are taken as the mean self-attention weights induced from the first token index to all other indices. Then we maximize the similarity between $att(\boldsymbol{x})$ and $att(\widetilde{\boldsymbol{x}})$ to ensure that the model produces similar output probability distributions over labels for both masked and unmasked inputs. The optimization objective for the EGT is:

$$\begin{aligned}\mathcal{L}_{kl}(\boldsymbol{x}, y, \theta) = D_{KL}\left(att(\boldsymbol{x}; \theta);\ IG(x)\right) + \\ D_{KL}\left(att(\widetilde{\boldsymbol{x}}; \theta);\ IG(x)\right),\end{aligned} \quad (7)$$

where D_{KL} is the KL divergence function between two distributions. The motivation behind two KL divergence terms is to encourage the model to focus on high salient words and ignore low salient words during training, and generate similar outputs for the original input $\boldsymbol{x}$ and masked input $\widetilde{\boldsymbol{x}}$, which can be seen as a special adversarial example. On the other hand, as the calculation of the mask input is batch-wise, the model should learn to assign low gradient values to irrelevant tokens for the predicted label in an iterative way.

3.4 Training

We define the final weighted loss as follows,

$$\mathcal{L} = \lambda_1 \mathcal{L}_{classify} + \lambda_2 \mathcal{L}_{gr} + \lambda_3 \mathcal{L}_{at} + \lambda_4 \mathcal{L}_{kl}, \quad (8)$$

where λ_1, λ_2, λ_3 and λ_4 are hyper-parameters. Note that mixing these losses brings an increase in computational complexity, requiring multiple forward and backward propagations, but improves the generalization and robustness of the model, and in this process we do not introduce external knowledge, only use salient ranking as self-training. At inference, we calculate the label probability and use different explanation methods in §4.2 to generate highlight explanations.

3.5 Erasure-based Faithfulness Evaluation

To evaluate post-hoc explanations, we adopt *sufficiency* that measures the degree to which the highlight explanation is adequate for a model to make predictions, and *comprehensiveness* that measures the influence of explanations to predictions (DeYoung et al., 2020). These two metrics are usually used to evaluate faithfulness as it does not require re-training and the main idea is to estimate the effect of changing parts of inputs on model output. Let $p_\theta(y^j|\boldsymbol{x}_i)$ be the output probability of the j-th class for the i-th example, and rationale $\boldsymbol{r}_i$ extracted according to attribution scores. Formally, the sufficiency we used is as follows:

$$\mathrm{S}(\boldsymbol{x}_i, y^j, \boldsymbol{r}_i) = 1 - max(0, p_\theta(y^j|\boldsymbol{x}_i) - p_\theta(y^j|\boldsymbol{r}_i)), \quad (9)$$

$$\text{sufficiency}(\boldsymbol{x}_i, y^j, \boldsymbol{r}_i) = \frac{\mathrm{S}(\boldsymbol{x}_i, y^j, \boldsymbol{r}_i) - \mathrm{S}(\boldsymbol{x}_i, y^j, \boldsymbol{0})}{1 - \mathrm{S}(\boldsymbol{x}_i, y^j, \boldsymbol{0})}, \quad (10)$$

where higher sufficiency values are better as we normalize and reverse it between 0 and 1, and $\mathrm{S}(\boldsymbol{x}_i, y^j, \boldsymbol{0})$ is the sufficiency of the original input where no token is erased. Similarly, we define the comprehensiveness as follows:

$$\mathrm{C}(\boldsymbol{x}_i, y^j, \boldsymbol{r}_i) = max(0, p_\theta(y^j|\boldsymbol{x}_i) - p_\theta(y^j|\overline{\boldsymbol{r}}_i)), \quad (11)$$

$$\text{comprehensiveness}(\boldsymbol{x}_i, y^j, \boldsymbol{r}_i) = \frac{\mathrm{C}(\boldsymbol{x}_i, y^j, \boldsymbol{r}_i)}{1 - \mathrm{S}(\boldsymbol{x}_i, y^j, \boldsymbol{0})}, \quad (12)$$

where higher comprehensiveness values are better. As choosing the appropriate rationale length is dataset dependent, we use the Area Over the Perturbation Curve (AOPC) metrics for sufficiency and comprehensiveness. It defines bins of tokens to be erased and calculates the average measures across bins. Here, we keep the top 1%, 5%, 10%, 20%, 50% tokens into bins in the order of decreasing attribution scores.

4 Experiments

Our baseline is a text classification model fine-tuned on the training set while the same pre-trained language model is applied to REGEX. In other words, the baseline is optimized by Eqn. 1 without robustness improvement and explanation guided training mechanisms.

4.1 Dataset

We consider six datasets to evaluate explanations and the data statistics are as follows.

SST: The Stanford Sentiment Treebank (SST) dataset (Socher et al., 2013) includes review sentences (positive/negative) for analysis of the compositional effect of sentiment. The training set, development set, and test set consist of 6920, 872, and 1821 examples.

IMDB: The IMDB dataset (Maas et al., 2011) consists of 25k movies reviews from IMDB website labeled by sentiment (positive/negative). The training set, development set, and test set consist of 20k, 2.5k, and 2.5k examples.

Yelp: The Yelp dataset (Zhang et al., 2015) includes highly polar movie reviews and is transformed to a binary classification task (positive/negative). The training set, development set, and test set consist of 476k, 84k, and 38k examples.

Amazon Reviews: The amazon reviews dataset (Ni et al., 2019) is constructed by personalized justification from existing review data. We choose the 3-class review and product metadata for three categories: Digital Music, Prime Pantry and Musical Instruments (Chrysostomou and Aletras, 2022). These examples are then divided into three subsets: **AmazDigiMu** (122k/21k/25k examples), **AmazPantry** (99k/17k/20k examples) and **AmazInstr** (16k/29k/3k examples).

4.2 Post-hoc Explanation Methods

We consider five feature attribution methods and a random attribution method:

Random (Rand) (Chrysostomou and Aletras, 2022): Token importance is assigned at random.

Attention (α) (Jain et al., 2020): Normalized attention scores are used to calculate token importance.

Scaled Attention ($\alpha\nabla\alpha$) (Serrano and Smith, 2019):Normalized attention scores α_i scaled by the corresponding gradients $\nabla\alpha_i = \frac{\partial\hat{y}}{\partial\alpha_i}$.

InputXGrad ($x\nabla x$) (Shrikumar et al., 2016; Kindermans et al., 2016): The input attribution importance is generated by multiplying the gradient $\nabla x_i = \frac{\partial \hat{y}}{\partial x_i}$ with the input.

Integrated Gradients (IG) (Sundararajan et al., 2017): See §3.3 for details.

DeepLift (Shrikumar et al., 2017): The difference between each neuron activation and a reference vector is used to rank words.

5 Results

5.1 Post-hoc Explanations Faithfulness

We conduct experiments on the faithfulness metrics (i.e., normalized sufficiency and normalized comprehensive) to compare the fidelity of different post-hoc explanation methods between the baseline and REGEX models. We extract rationale r from a model by selecting the top-k most important tokens measured by these post-hoc explanation methods. Following Chrysostomou and Aletras (2022), we also evaluate explanation faithfulness in out-of-domain settings without retraining models (i.e., zero-shot), and we follow their settings with six dataset pairs and a random attribution baseline. Specially, the model is first trained on the source datasets and then we evaluate its performance on test set of the target datasets.

As shown in Table 1, REGEX improves the explanation faithfulness with all five attribution methods by a large gap under most in-domain and out-of-domain settings. Among them, scaled attention and DeepLift perform better than others. For example, REGEX surpasses the baseline in the sufficiency metric for the explanation extracted by DeepLift under all scenarios, while the comprehensiveness decreases when the model is trained in the AmazDigiMu dataset and tested in the AmazInstr dataset. It shows that REGEX improves the fidelity of post-hoc explanations measured by sufficiency and comprehensiveness. Nevertheless, we observe a decrease in the comprehensiveness metrics for attention and IG on specific datasets. For example, considering the uncertainty of attention as an interpretable method (Serrano and Smith, 2019), the fidelity metrics of attention attribution are inferior to the baseline on all three Amazon Reviews datasets.

Overall, feature attribution approaches outperform the random attributions in most cases of in-domain and out-of-domain settings. Moreover, results show that post-hoc explanation sufficiency and comprehensiveness are higher in in-domain test-sets than out-of-domain except for the Yelp dataset. On the other hand, as shown in Table 2, REGEX improves the performance or achieves similar task performance than baseline on most out-of-domain datasets,

5.2 Quantitative Evaluation by FRESH Method

We further compare the average macro F1 of the FRESH classifier (Jain et al., 2020) across five random seeds in the in-domain and out-of-domain settings. In short, FRESH is a select-then-predict framework, and the general process is that an extractor is first trained where the labels are induced by arbitrary feature importance scores over token inputs; then, an independent classifier is trained exclusively on rationales provided by the extractor which are assumed to be inherently faithful. Here, rationales extracted by the top-k most important tokens are used as input to the classifier for training and test. As shown in Table 2, the best two methods are DeepLift and scaled attention, which achieve a similar performance as the original text input model in the in-domain and out-of-domain settings and is consistent with the faithfulness evaluation. For example, the FRESH classifier applying the DeepLift attribution method is higher than the baseline and outperforms the model with the full text input (97.1 vs. 96.9) on the Yelp dataset. It also illustrates that the performance of the FRESH method depends on the choice of the feature attribution method.

6 Analysis

6.1 Explanation Robustness

Following Zafar et al. (2021), we test *implementation invariance* of feature attributions by Untrained Model Test (UIT) and Different Initialization Test (DIT). The UIT and DIT measure the consistency and calculate the Jaccard similarity between feature attributions generated by the post-hoc explanation method. We use Jaccard similarity for explanations extracted by top 25% important tokens using the scaled attention method. If the two attributions are more similar, the Jaccard metric is higher. We compare the REGEX and baseline by comparing two identical models trained from different initializations. As shown in Table 3, REGEX achieves an improved performance than baseline. For example, REGEX gets 0.56 while baseline gets 0.36 for

Train	Test	Normalized Sufficiency (↑)						Normalized Comprehensiveness (↑)					
		RAND	$\alpha\nabla\alpha$	α	DeepLift	$x\nabla x$	IG	RAND	$\alpha\nabla\alpha$	α	DeepLift	$x\nabla x$	IG
SST	SST	.30(.38)	.68(.51)	.48(.42)	.71(.42)	.49(.40)	.49(.41)	.22(.19)	.56(.39)	.41(.22)	.52(.25)	.43(.26)	.43(.26)
	IMDB	.25(.31)	.54(.53)	.45(.39)	.46(.32)	.40(.31)	.40(.32)	.19(.23)	.75(.54)	.66(.34)	.61(.27)	.58(.27)	.58(.28)
	Yelp	.24(.32)	.51(.56)	.38(.40)	.45(.35)	.35(.33)	.36(.34)	.22(.21)	.70(.48)	.57(.28)	.59(.24)	.48(.24)	.47(.25)
IMDB	IMDB	.34(.32)	.82(.55)	.51(.46)	.80(.36)	.54(.36)	.53(.36)	.17(.16)	.71(.48)	.39(.31)	.62(.25)	.31(.23)	.32(.24)
	SST	.30(.24)	.72(.35)	.42(.28)	.68(.28)	.46(.27)	.45(.27)	.21(.27)	.59(.46)	.28(.32)	.51(.33)	.32(.33)	.33(.33)
	Yelp	.32(.35)	.81(.48)	.53(.41)	.79(.36)	.48(.36)	.47(.36)	.20(.21)	.71(.45)	.42(.32)	.64(.26)	.33(.26)	.34(.26)
Yelp	Yelp	.35(.23)	.82(.32)	.59(.31)	.82(.29)	.53(.24)	.53(.25)	.10(.12)	.64(.20)	.39(.14)	.63(.16)	.24(.15)	.23(.16)
	SST	.33(.41)	.76(.45)	.49(.43)	.75(.44)	.60(.41)	.60(.41)	.16(.17)	.57(.24)	.31(.18)	.55(.21)	.40(.22)	.40(.22)
	IMDB	.38(.18)	.83(.34)	.59(.32)	.82(.25)	.61(.22)	.61(.22)	.13(.19)	.74(.34)	.43(.29)	.70(.23)	.31(.23)	.31(.24)
AmazDigiMu	AmazDigiMu	.50(.34)	.73(.56)	.55(.34)	.66(.31)	.60(.41)	.62(.39)	.18(.13)	.60(.32)	.12(.14)	.21(.10)	.26(.16)	.24(.17)
	AmazInstr	.60(.29)	.75(.54)	.67(.32)	.67(.31)	.66(.33)	.68(.32)	.16(.19)	.62(.47)	.18(.23)	.15(.19)	.24(.22)	.23(.23)
	AmazPantry	.53(.33)	.70(.55)	.60(.33)	.64(.31)	.60(.37)	.62(.36)	.19(.21)	.61(.46)	.13(.22)	.18(.17)	.24(.23)	.22(.25)
AmazPantry	AmazPantry	.55(.25)	.79(.46)	.56(.36)	.82(.19)	.54(.28)	.52(.27)	.15(.20)	.50(.42)	.14(.31)	.52(.15)	.16(.25)	.17(.25)
	AmazDigiMu	.54(.24)	.78(.47)	.56(.37)	.82(.19)	.52(.27)	.50(.26)	.14(.19)	.50(.41)	.16(.32)	.52(.15)	.14(.23)	.15(.24)
	AmazInstr	.55(.17)	.81(.42)	.53(.30)	.82(.15)	.51(.20)	.50(.20)	.14(.24)	.60(.52)	.13(.40)	.60(.23)	.15(.30)	.16(.30)
AmazInstr	AmazInstr	.52(.16)	.82(.34)	.58(.18)	.82(.21)	.59(.18)	.58(.17)	.16(.26)	.58(.52)	.22(.26)	.56(.29)	.18(.28)	.19(.29)
	AmazDigiMu	.56(.21)	.82(.38)	.58(.21)	.82(.22)	.60(.24)	.59(.22)	.12(.23)	.48(.46)	.16(.20)	.46(.22)	.15(.24)	.15(.25)
	AmazPantry	.56(.22)	.81(.39)	.58(.21)	.81(.23)	.59(.24)	.58(.23)	.13(.27)	.50(.51)	.16(.22)	.47(.25)	.16(.27)	.17(.29)

Table 1: Normalized sufficiency and comprehensiveness in the in-domain and out-of-domain settings for five feature attribution approaches and a random attribution. REGEX vs. baseline (shown in brackets).

Train	Test	Full-text F1	$\alpha\nabla\alpha$	α	DeepLift	$x\nabla x$	IG
SST (20%)	SST	89.7(90.1)	**88.9(87.7)**	83.0(81.1)	87.3(84.4)	77.8(76.3)	77.8(76.8)
	IMDB	83.4(84.3)	**86.3(81.8)**	65.3(52.6)	81.1(64.0)	53.2(55.0)	53.2(56.3)
	Yelp	87.8(87.9)	**90.2(88.1)**	76.5(72.6)	80.4(75.4)	64.4(59.6)	64.4(63.9)
IMDB (2%)	IMDB	91.3(91.1)	**88.9(87.9)**	79.2(80.4)	87.6(87.2)	59.1(59.8)	59.1(59.7)
	SST	88.0(85.8)	**80.6(80.9)**	71.8(71.8)	72.9(70.1)	65.7(69.6)	65.7(70.7)
	Yelp	90.3(91.0)	**90.4(87.8)**	72.7(82.0)	86.5(79.4)	70.5(69.0)	70.5(69.1)
Yelp (10%)	Yelp	96.1(96.9)	96.3(94.0)	87.1(90.4)	**97.1(93.6)**	71.2(70.5)	71.2(71.9)
	SST	85.3(86.8)	**82.0(59.3)**	58.1(69.8)	69.9(67.2)	67.6(67.7)	67.6(69.3)
	IMDB	86.2(88.6)	**86.7(78.0)**	51.5(64.5)	79.1(66.6)	48.0(53.0)	48.0(55.8)
AmazDigiMu (20%)	AmazDigiMu	72.4(70.6)	**67.9(66.1)**	62.5(63.4)	67.5(65.8)	48.3(51.9)	48.3(65.8)
	AmazInstr	60.3(61.2)	**60.9(58.0)**	50.0(57.2)	**60.9(57.4)**	39.0(46.0)	39.0(57.2)
	AmazPantry	61.0(64.6)	**60.1(59.1)**	46.3(56.5)	59.0(56.5)	38.8(44.8)	38.8(44.8)
AmazPantry (20%)	AmazPantry	71.3(70.2)	67.8(67.3)	59.6(62.6)	**68.0(67.2)**	50.3(48.6)	50.3(48.7)
	AmazDigiMu	60.1(59.5)	**58.5(57.7)**	51.5(54.6)	58.4(56.2)	42.7(41.2)	42.7(57.7)
	AmazInstr	65.7(64.5)	64.9(63.8)	54.9(58.0)	**65.5(63.6)**	43.3(40.1)	43.3(40.3)
AmazInstr (20%)	AmazInstr	72.9(71.5)	69.5(69.8)	63.1(62.1)	**70.7(69.7)**	47.5(45.6)	47.5(48.6)
	AmazDigiMu	60.7(61.3)	58.6(60.0)	51.6(53.2)	**58.9(57.8)**	43.7(43.8)	43.7(60.0)
	AmazPantry	67.9(68.2)	65.0(64.5)	55.8(56.3)	**65.6(63.1)**	45.2(44.6)	45.2(47.6)

Table 2: Average macro F1 results of Full-text and FRESH models with a prescribed rationale length. REGEX vs. baseline (shown in brackets), all the values are averaged across 5 seeds and full results are in Appendix C.

Jaccard@25%	Init#1	Init#2	Init#3	Init#4	#Untrained
Init#1	1.0	.44(.33)	.54(.34)	.56(.34)	.28(.27)
Init#2	.44(.33)	1.0	.45(.44)	.41(.34)	.16(.17)
Init#3	.54(.34)	.45(.44)	1.0	.56(.36)	.22(.21)
Init#4	.56(.34)	.41(.34)	.56(.36)	1.0	.12(.16)
#Untrained	.28(.27)	.16(.17)	.22(.21)	.12(.16)	1.0

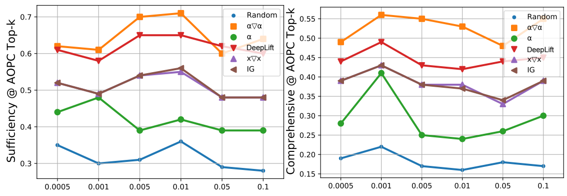
Table 3: Jaccard@25% between the feature attributions (REGEX vs. baseline, here we use scaled attention) for models with same architecture, with same data, and same learning schedule, except for their randomly chosen initial parameters.

`Init#3` and `Init#4`. As we expected, the similarity between explanations of the trained and untrained models is low, e.g., 0.12 between `Init#4` and `#Untrained`. It shows that improving faithfulness of explanations can strengthen interpretation robustness. However, the overall results between the two feature attributions are still low as 50% of similarity comparisons are less than 0.5.

6.2 Ablation Study

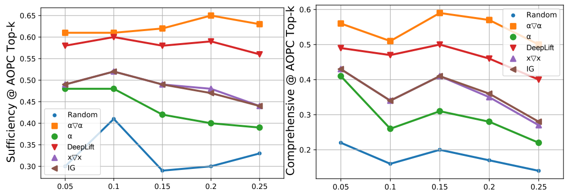
We perform ablation studies to explore the effect of robustness improvement and explanation guided training for faithfulness evaluations shown in Table 4 and Table 9, and investigate the effect of different hyper-parameters on experimental results. We further compare the effect of the two aggregation methods (i.e., mean and ℓ_2) during explanation guided training and the effect of using different feature attribution in §3.3 on the faithfulness of highlight explanations after training.

Robustness improvement is important for improving sufficiency and comprehensiveness. Compared with REGEX without explanation guided training, sufficiency and comprehensiveness of REGEX without robustness improvement



(a) Sufficiency (b) Comprehensiveness

Figure 3: Comparisons between different explanation guided training λ_4 on the SST dataset.



(a) Sufficiency (b) Comprehensiveness

Figure 4: Comparisons between different mask ratio K on the SST dataset.

decrease more (0.14 vs. 0.02, 0.23 vs. 0.02, 0.29 vs. 0.07, 0.35 vs. 0.08).

The performance of the attention method varies more across different hyper-parameters. In Figure 3, we compare different λ_4 in Eqn. 8 and observe that all methods achieve best sufficiency at 0.01 and best comprehensiveness at 0.001. In Figure 4, we compare different mask ratios in §3.3 and find that the mask ratio between 0.15 and 0.2 is useful as larger values can bring noise.

The choice of aggregation method and feature attribution method in §3.3 has a large effect on the faithfulness evaluation. We find that for most attribution methods, ℓ_2 aggregation has higher fidelity performance. For example, Saliency with ℓ_2 aggregation is better than Saliency with mean aggregation with more sufficiency improvement (0.70 vs. 0.55). Though there is no best method for explanation guided training, gradient-based methods (e.g., IG, 0.71) may be good choices in line with Atanasova et al. (2020).

6.3 Case Study

Table 5 presents two randomly-chosen examples of the test set of the IMDB dataset. For example, the top-k important tokens returned by REGEX are *wonderfully*, *wonderful*, *wonderful*, *excellent* and *great* in the first example. We observe that these highlight explanations seem intuitive to humans and reasonably plausible. Though faithfulness and plausibility may not be necessarily correlative (Jacovi and Goldberg, 2020), we find that the highlights extracted by REGEX contain more sentiment-related words, which should be helpful for review-based text classification.

Methods	Suff.		Comp.	
	$\alpha\nabla\alpha$	DeepLift	$\alpha\nabla\alpha$	DeepLift
w/o robustness improvement	.54	.42	.33	.17
w/o explanation guided training	.66	.64	.54	.44
Saliency (Mean)	.52	.48	.48	.42
InputXGrad (Mean)	.52	.53	.37	.39
DeepLift (Mean)	.61	.58	.52	.49
IG (Mean)	.47	.45	.49	.51
Saliency (ℓ_2)	**.70**	.65	.55	.43
InputXGrad (ℓ_2)	.58	.54	**.58**	.49
DeepLift (ℓ_2)	.69	.68	.53	.47
REGEX	.68	**.71**	.56	**.52**

Table 4: Ablation study with different aggregation methods and feature attribution methods in §3.3.

Label: Positive **Prediction**: Positive **Dataset**: IMDB **ID**: Test 1364
…,is the fact that the wonderful RAYMOND MASSEY is relegated to the last twenty or so minutes in the trial scene. … David NIVEN and KIM HUNTER are wonderfully cast as the young lovers…. French accented MARIUS GORING is a delight (he even gets in a remark about Technicolor) as the heavenly messenger sent to reclaim Niven when his wartime death goes unreported due to an oversight. Seeing this tonight on TCM for the first time in twenty or so years, I think it's a supreme example of what a wonderful year 1946 was for films. The Technicolor photography, somewhat subdued and not garish at all, is excellent and the way it shifts into B&W for the heavenly sequences is done with great imagination and effectiveness….
Label: Negative **Prediction**: Negative **Dataset**: IMDB **ID**: Test 1373
…but pompous horror icon Christopher Lee squirming in the midst of it all (the gracefully-aged star has pathetically asserted a number of times in interviews that he hasn't appeared in horror-oriented fare since his last picture for Hammer Films back in 1976!). Anyway, this film should have borne the subtitle "Your Movie Is A Turd" - being astoundingly inept in all departments (beginning with the all-important werewolf make-up)! The plot (and dialogue) is not only terrible, but it has the limpest connection with Dante's film - strangely enough, the author of the original novel Gary Brandner co-wrote this himself! Still, one of the undeniable highlights (er...low points) of the film is the pointless elliptical editing -

Table 5: We randomly pick two examples from test set of IMDB dataset, and highlight the Top-k important tokens using DeepLift method (REGEX vs. Baseline).

7 Conclusion

We explore whether the fidelity of explanations can be further optimized by techniques to improve model robustness and conduct extensive empirical studies on six datasets in both in-domain and out-of-domain settings. Experimental results show that our method REGEX improves both fidelity metrics and performance of select-then-predict models. These findings provide evidence that improving model robustness may produce more faithful explanations. In the future, we would like to investigate more PLMs architectures and more faithfulness metrics under the standard evaluation protocol.

References

Julius Adebayo, Michael Muelly, Harold Abelson, and Been Kim. 2022. Post hoc explanations may be ineffective for detecting unknown spurious correlation. In *Proc. of ICLR*.

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

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2020. A diagnostic study of explainability techniques for text classification. In *Proc. of EMNLP*.

David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. 2010. How to explain individual classification decisions. *J. Mach. Learn. Res.*

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

Meghana Moorthy Bhat, Alessandro Sordoni, and Subhabrata Mukherjee. 2021. Self-training with few-shot rationalization: Teacher explanations aid student in few-shot NLU. In *Proc. of EMNLP*.

Oana-Maria Camburu, Eleonora Giunchiglia, Jakob Foerster, Thomas Lukasiewicz, and Phil Blunsom. 2019. Can I trust the explainer? verifying post-hoc explanatory methods. *arXiv preprint arXiv:1910.02065*.

George Chrysostomou and Nikolaos Aletras. 2021a. Enjoy the salience: Towards better transformer-based faithful explanations with word salience. In *Proc. of EMNLP*.

George Chrysostomou and Nikolaos Aletras. 2021b. Improving the faithfulness of attention-based explanations with task-specific information for text classification. In *Proc. of ACL*.

George Chrysostomou and Nikolaos Aletras. 2022. An empirical study on explanations in out-of-domain settings. *Proc. of ACL*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL*.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. 2020. ERASER: A benchmark to evaluate rationalized NLP models. In *Proc. of ACL*.

Shuoyang Ding and Philipp Koehn. 2021. Evaluating saliency methods for neural language models. In *Proc. of NAACL*.

Harris Drucker and Yann LeCun. 1992. Improving generalization performance using double backpropagation. *IEEE Trans. Neural Networks*.

Amirata Ghorbani, Abubakar Abid, and James Y. Zou. 2019. Interpretation of neural networks is fragile. In *Proc. of AAAI*.

Ivan Girardi, Pengfei Ji, An-phi Nguyen, Nora Hollenstein, Adam Ivankay, Lorenz Kuhn, Chiara Marchiori, and Ce Zhang. 2018. Patient risk assessment and warning symptom detection using deep attention-based neural networks. In *EMNLP workshop LOUHI 2018*.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *Proc. of ICLR*.

Xiaochuang Han and Yulia Tsvetkov. 2021. Influence tuning: Demoting spurious correlations via instance attribution and instance-driven updates. In *Proc. of EMNLP Findings*.

Peter Hase and Mohit Bansal. 2021. When can models learn from explanations? A formal framework for understanding the roles of explanation data. *arXiv preprint arXiv:2102.02201*.

Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, and Dawn Song. 2020. Pretrained transformers improve out-of-distribution robustness. In *Proc. of ACL*.

Juyeon Heo, Sunghwan Joo, and Taesup Moon. 2019. Fooling neural network interpretations via adversarial model manipulation. In *Proc. of NeurIPS*.

Cheng-Yu Hsieh, Chih-Kuan Yeh, Xuanqing Liu, Pradeep Kumar Ravikumar, Seungyeon Kim, Sanjiv Kumar, and Cho-Jui Hsieh. 2021. Evaluations and methods for explanation through robustness analysis. In *Proc. of ICLR*.

Aya Abdelsalam Ismail, Héctor Corrada Bravo, and Soheil Feizi. 2021. Improving deep learning interpretability by saliency guided training. In *Proc. of NeurIPS*.

Adam Ivankay, Ivan Girardi, Chiara Marchiori, and Pascal Frossard. 2022. Fooling explanations in text classifiers. In *Proc. of ICLR*.

Alon Jacovi and Yoav Goldberg. 2020. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In *Proc. of ACL*.

Sarthak Jain, Sarah Wiegreffe, Yuval Pinter, and Byron C. Wallace. 2020. Learning to faithfully rationalize by construction. In *Proc. of ACL*.

Sahil Jayaram and Emily Allaway. 2021. Human rationales as attribution priors for explainable stance detection. In *Proc. of EMNLP*.

Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao. 2019. Smart: Robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization. *arXiv preprint arXiv:1911.03437*.

Yiming Ju, Yuanzhe Zhang, Zhixing Tian, Kang Liu, Xiaohuan Cao, Wenting Zhao, Jinlong Li, and Jun Zhao. 2021. Enhancing multiple-choice machine reading comprehension by punishing illogical interpretations. In *Proc. of EMNLP*.

Pieter-Jan Kindermans, Sara Hooker, Julius Adebayo, Maximilian Alber, Kristof T. Schütt, Sven Dähne, Dumitru Erhan, and Been Kim. 2017. The (un)reliability of saliency methods. *Explainable AI 2019*.

Pieter-Jan Kindermans, Kristof Schütt, Klaus-Robert Müller, and Sven Dähne. 2016. Investigating the influence of noise and distractors on the interpretation of neural networks. *arXiv preprint arXiv:1611.07270*.

Dongfang Li, Baotian Hu, Qingcai Chen, Tujie Xu, Jingcong Tao, and Yunan Zhang. 2022. Unifying model explainability and robustness for joint text classification and rationale extraction. *Proc. of AAAI*.

Zachary C. Lipton. 2018. The mythos of model interpretability. *ACM Queue*.

Frederick Liu and Besim Avci. 2019. Incorporating priors with feature attribution on text classification. In *Proc. of ACL*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proc. of ACL*.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards deep learning models resistant to adversarial attacks. In *Proc. of ICLR*.

Clara Meister, Stefan Lazov, Isabelle Augenstein, and Ryan Cotterell. 2021. Is sparse attention more interpretable? In *Proc. of ACL*.

Takeru Miyato, Shin ichi Maeda, Masanori Koyama, Ken Nakae, and Shin Ishii. 2015. Distributional smoothing with virtual adversarial training. *arXiv preprint arXiv: 1507.00677*.

Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. 2018. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp. In *Proc. of EMNLP: System Demonstrations*.

Jianmo Ni, Jiacheng Li, and Julian J. McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proc. of EMNLP*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*

Marco Túlio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with checklist. In *Proc. of ACL*.

Laura Rieger, Chandan Singh, W. James Murdoch, and Bin Yu. 2020. Interpretations are useful: Penalizing explanations to align neural networks with prior knowledge. In *Proc. of ICML*.

Andrew Slavin Ross and Finale Doshi-Velez. 2018. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In *Proc. of AAAI*.

Sofia Serrano and Noah A. Smith. 2019. Is attention interpretable? In *Proc. of ACL*.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. 2017. Learning important features through propagating activation differences. In *Proc. of ICML*.

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

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2014. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *Proc. of ICLR*.

Sanchit Sinha, Hanjie Chen, Arshdeep Sekhon, Yangfeng Ji, and Yanjun Qi. 2021. Perturbing inputs for fragile interpretations in deep natural language processing. In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*.

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

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proc. of EMNLP*.

Joe Stacey, Yonatan Belinkov, and Marek Rei. 2022. Supervising model attention with human explanations for robust natural language inference. In *Proc. of AAAI*.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proc. of ICML*.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Xuezhi Wang, Haohan Wang, and Diyi Yang. 2021. Measure and improve robustness in NLP models: A survey. *arXiv preprint arXiv:2112.08313*.

Sarah Wiegreffe, Ana Marasovic, and Noah A. Smith. 2021. Measuring association between labels and free-text rationales. In *Proc. of EMNLP*.

Xi Ye and Greg Durrett. 2022. Can explanations be useful for calibrating black box models? In *Proc. of ACL*.

Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Sai Suggala, David I. Inouye, and Pradeep Ravikumar. 2019. On the (in)fidelity and sensitivity of explanations. In *Proc. of NeurIPS*.

Fan Yin, Zhouxing Shi, Cho-Jui Hsieh, and Kai-Wei Chang. 2021. On the faithfulness measurements for model interpretations. *arXiv preprint arXiv:2104.08782*.

Muhammad Bilal Zafar, Michele Donini, Dylan Slack, Cédric Archambeau, Sanjiv Das, and Krishnaram Kenthapadi. 2021. On the lack of robust interpretability of neural text classifiers. In *Proc. of ACL Findings*.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Proc. of NeurIPS*.

Metric	Attack Results
Number of successful attacks:	12(45)
Original accuracy(%):	93.0(96.0)
Accuracy under attack(%):	90.0(84.8)
Attack success rate(%):	3.23(11.71)
Average perturbed word(%):	39.06(27.02)
Average num. words per input:	244.73(244.73)
Avg num queries:	408.47(339.69)

Table 6: Attack results of REGEX and baseline by CHECKLIST attack recipe.

Methods	Full-text F1
w/o robustness improvement	90.57±0.52
w/o explanation guided training	85.19±2.80
Saliency (Mean)	87.81±3.64
InputXGrad (Mean)	91.21±0.23
DeepLift (Mean)	87.99±0.48
IG (Mean)	91.60±0.08
Saliency (ℓ_2)	83.52±1.29
InputXGrad (ℓ_2)	90.83±0.29
DeepLift (ℓ_2)	87.62±0.53
REGEX	89.73±0.05

Table 7: Macro F1 and standard deviations with different aggregation methods and feature attribution methods in §3.3.

A Experiment Settings

We use Spacy [2] to pre-tokenize the sentence and apply the BERT-base model to encode text (Devlin et al., 2019). We use AdamW optimizer with a batch size of 8 for model training. The initial learning rate is 1×10^{-5} for fine-tuning BERT parameters and 1×10^{-4} for the classification layer. The maximum sequence length, the dropout rate, the gradient accumulation steps, the training epoch and the hidden size d are set to 256, 0.1, 10%, 10, 768 respectively. We clip the gradient norm within 1.0. The learning parameters are selected based on the best performance on the development set. Our model is trained with NVIDIA Tesla A100 40GB GPUs (PyTorch & Huggingface/Transformers [3] & Captum [4]). Following Jiang et al. (2019), we set the perturbation size $\epsilon = 1 \times 10^{-5}$, the step size $\eta = 1 \times 10^{-3}$, ascent iteration step $C = 2$ and the variance of normal distribution $\sigma = 1 \times 10^{-5}$. The weight parameters λ_1, λ_2, λ_3, λ_4 are set to 1.0, 0.01, 0.5, 0.01 respectively. The mask ration K is set to 0.15. The number of steps used by the approximation method in IG is 50, and we use zero scalar corresponding to each input tensor as IG baselines. For the baseline and FRESH model, we use the same transformer-based models as mentioned previously to encode tokens and we choose rationale length by following Chrysostomou and Aletras (2022). The model is trained for 10 epochs, and we keep the best models with respect to macro F1 scores on the development sets.

B Text Classification to Attacks

We conduct the behavioral testing with CHECKLIST (Ribeiro et al., 2020) and `TextAttack` (Morris et al., 2020) to attack REGEX text classification models. We randomly choose 400 examples from IMDB test set as original attack examples, and the attack recipe greedily search adversarial examples to change the predicted lable by contracting, extending, and substituting name entities in the sentence. The results are shown in the Table 6 and the attack success rate which is used to evaluate the effectiveness of the attacks is 3.23%.

C Full Results

Table 7 presents the Full-text F1 of variants in ablation study. Table 8 lists the full results for FRESH (select-then-predict) models. Table 9 lists the full results of ablation study.

D Limitations

Possible limitations include the single PLM architecture (size) and faithfulness evaluation metrics are not necessarily comprehensive. Considering if we can intrinsically know or derive the golden faithful explanations (Bastings et al., 2021), the exploration of model robustness and explainability will be alternatively investigated for revealing the internal reasoning processes.

[2]https://spacy.io/models/en
[3]https://github.com/huggingface/transformers
[4]https://captum.ai/

Train	Test	$\alpha\nabla\alpha$	α	DeepLift	$x\nabla x$	IG
SST	SST	88.88±0.7	83.00±0.3	87.31±0.5	77.84±0.5	77.84±0.5
	IMDB	86.27±0.2	65.32±1.9	81.18±0.6	53.22±0.6	53.22±0.6
	Yelp	90.15±0.1	76.45±0.6	80.35±2.1	64.38±0.5	64.38±0.5
IMDB	IMDB	88.88±0.3	79.16±0.2	87.60±0.2	59.14±1.0	59.14±1.0
	SST	80.60±1.6	71.75±0.3	72.91±0.6	65.68±2.2	65.68±2.2
	Yelp	90.37±0.5	72.71±1.0	86.51±0.4	70.54±0.9	70.54±0.9
Yelp	Yelp	96.27±0.1	87.13±0.1	97.05±0.0	71.22±0.1	71.22±0.1
	SST	82.03±0.5	58.13±0.6	69.89±0.4	67.58±0.6	67.58±0.6
	IMDB	83.68±0.4	51.51±0.4	79.10±1.2	47.99±1.8	47.99±1.8
AmazDigiMu	AmazDigiMu	67.87±0.4	62.53±0.9	67.52±1.0	48.30±2.2	48.30±2.2
	AmazInstr	60.95±0.1	49.98±0.8	60.92±0.5	39.02±0.2	39.02±0.2
	AmazPantry	60.05±0.3	46.27±0.9	59.01±1.0	38.83±1.0	38.83±1.0
AmazPantry	AmazPantry	67.83±1.0	59.62±0.8	67.99±1.6	50.33±1.2	50.33±1.2
	AmazDigiMu	58.49±0.8	51.48±1.0	58.40±0.5	42.71±0.8	42.71±0.8
	AmazInstr	64.91±0.5	54.92±1.7	65.55±1.0	43.31±0.9	43.31±0.9
AmazInstr	AmazInstr	69.52±0.7	63.06±0.6	70.73±0.2	47.47±1.0	47.47±1.0
	AmazDigiMu	58.59±0.8	51.64±0.4	58.93±0.5	43.68±0.7	43.68±0.7
	AmazPantry	64.95±0.9	55.82±0.6	65.58±0.2	45.24±0.8	45.24±0.8

Table 8: Macro F1 and standard deviations of FRESH models with Top-k explanations. RED means REGEX outperforms the baseline.

Methods	**Normalized Sufficiency (↑)**						**Normalized Comprehensiveness (↑)**					
	RAND	$\alpha\nabla\alpha$	α	**DeepLift**	$x\nabla x$	**IG**	RAND	$\alpha\nabla\alpha$	α	**DeepLift**	$x\nabla x$	**IG**
w/o robustness improvement	**.38**	.54	.43	.42	.42	.42	.12	.33	.18	.17	.20	.20
w/o virtual adversarial training	.27	.47	.32	.31	.33	.33	.14	.39	.21	.19	.24	.24
w/o input gradient regularization	.23	.54	.30	.32	.40	.40	.19	.57	.25	.28	.40	.40
w/o explanation guided training	.32	.66	.40	.64	.54	.54	.16	.54	.27	.44	.39	.39
Saliency (Mean)	.32	.52	.32	.48	.44	.45	.25	.48	.53	.42	.40	.38
InputXGrad (Mean)	.40	.52	.43	.53	.42	.42	.18	.37	.19	.39	.22	.22
DeepLift (Mean)	.36	.61	.42	.58	.50	.51	.22	.52	**.66**	.49	.37	.37
IG (Mean)	.29	.47	.37	.45	.29	.27	.24	.49	.26	.51	.28	.33
Saliency (ℓ_2)	.32	**.70**	.36	.65	**.54**	**.54**	.17	.55	.20	.43	.37	.37
InputXGrad (ℓ_2)	.34	.58	.38	.54	.43	.43	**.29**	**.58**	.25	.49	.31	.30
DeepLift (ℓ_2)	.30	.69	.39	.68	.53	.53	.16	.53	.26	.47	.37	.37
REGEX	.30	.68	**.48**	**.71**	.49	.49	.22	.56	.41	**.52**	**.43**	**.43**

Table 9: Full results of ablation study with different aggregation methods and feature attribution methods in §3.3.