Context-Aware Estimation of Attribution Robustness In Text

Anonymous authors
Paper under double-blind review

Abstract

Explanations are crucial parts of deep neural network (DNN) classifiers. In high stakes applications, faithful and robust explanations are important to understand DNN classifiers and gain trust. However, recent work has shown that state-of-the-art attribution methods in text classifiers are susceptible to imperceptible adversarial perturbations that alter explanations significantly while maintaining the correct prediction outcome. If undetected, this can critically mislead the users of DNNs. Thus, it is crucial to understand the influence of such adversarial perturbations on the networks' explanations. In this work, we establish a novel definition of attribution robustness (AR) in text classification. Crucially, it reflects both attribution change induced by adversarial input alterations and perceptibility of such alterations. Moreover, we introduce a set of measures to effectively capture several aspects of perceptibility of perturbations in text, such as semantic distance to the original text, smoothness and grammaticality of the adversarial samples. We then propose our novel Context-AwareExplanationAttack (CEA), a strong adversary that provides a tight estimation for attribution robustness in text classification. CEA uses context-aware masked language models to extract word substitutions that result in fluent adversarial samples. Finally, with experiments on several classification architectures, we show that CEA consistently outperforms current state-of-the-art AR estimators, yielding perturbations that alter explanations to a greater extent while being less perceptible.

1 Introduction

Attribution methods aim to give insights into causal relationships between deep neural networks' (DNNs) inputs and their outcome prediction. They are fundamental to unravel the black-box nature of DNNs and are widely used both in the image and natural language domain. Commonly used attributions like Saliency Maps (Simonyan et al., 2013), Integrated Gradients (Sundararajan et al., 2017), DeepLIFT (Shrikumar et al., 2017) and Self-Attention (Bahdanau et al., 2015) highlight input features that are deemed important for the DNNs in the inference process. These methods are especially attractive and useful, as they provide on-the-fly explanations without requiring any domain-specific knowledge from users or extensive computation resources.

However, it has been shown recently that many of these attributions lack robustness towards adversarial perturbations (Ghorbani et al., 2019). Carefully crafted, *imperceptible* input alterations change the explanations significantly without modifying the output prediction of the DNNs. This violates the *prediction assumption* of faithful explanations (Jacovi & Goldberg, 2020), which states that similar inputs should have similar explanations for identical outputs. Figure 1 exemplifies this fragility of attributions. In many safety-critical natural language processing problems, such as EHR classification (Girardi et al., 2018), robustness is a key factor for DNNs to be deployed in real life. For instance, a medical professional assessing EHRs would neither understand nor trust a model that yields two significantly different explanations for seemingly identical input texts and predictions. Hence, it is fundamental to understand how the networks and attributions behave in the presence of input perturbations and how perceptible those alterations are to the user.

In this work, we focus on understanding the adversarial robustness of attribution maps (AR) in *text classification* problems. Specifically, we are interested in investigating and quantifying the extent to which small input perturbations can alter explanations in DNNs and how perceptible such alterations are. We do so by

Original sample	**CEA perturbed sample** (ours)	**TEF perturbed sample** (Ivankay et al., 2022)
peek at the week : **ben** vs. the **streak** | yet another risky **game** for that patriots winning streak , now at 21 . pittsburgh hasn # 39;t lost at home , and **rookie quarterback** ben roethlisberger hasn # 39;t lost , period . $F(\boldsymbol{s}, \text{“Sports”}) = 0.99$	**peek** at the playoffs : ben vs. the **steelers** | **yet** another risky **game** for that patriots winning **streak** , now at 21 **. pittsburgh** hasn # **34** lost at home , and rookie **quarterback ben** roethlisberger hasn # 39;t lost , period $\geq$ $F(\boldsymbol{s}, \text{“Sports”}) = 0.95$ **PCC**: 0.02 *SemS*: 0.97, r: 14.9	hoodwink at **the zou** : **suis vs.** the **wave** | **yet** another **risky** game **for** that **patriots** winning **streak** , now **at** 21 **. pittsburgh hasn** # 39;t lost at **home** , **and rookie quarterback ben** roethlisberger hasn # 39;t lost , period . $F(\boldsymbol{s}, \text{“Sports”}) = 1.0$ **PCC**: 0.22 *SemS*: 0.9, r: 3.4
press the delete key . $F(\boldsymbol{s}, \text{“Negative”}) = 0.99$	**hit** the delete **key .** $F(\boldsymbol{s}, \text{“Negative”}) = 0.95$ **PCC**: -0.05 *SemS*: 0.98, r: 30	**newspaper the delete key .** $F(\boldsymbol{s}, \text{“Negative”}) = 0.95$ **PCC**: 0.6 *SemS*: 0.8, r: 1.1
intel seen **readying** new **wi - fi** chips | intel corp . this week isexpected to introduce a chip that adds support for a relativelyobscure version of wi - **fi** , analysts said on monday , in a movethat could help ease congestion on **wireless** networks . $F(\boldsymbol{s}, \text{“Sci/Tech”}) = 0.78$	intel seen **readying wireless** wi - **fi** chips | intel corp . this week isexpected to launch a specification that added support for a relativelyobscure version of wi - fi , analysts said on monday , in a movethat could help ease congestion on wireless networks . $F(\boldsymbol{s}, \text{“Sci/Tech”}) = 0.95$ **PCC**: 0.27 *SemS*: 0.98, r: 20	**intel** seen readying nouveau **wi** - **fi** chips | intel corp . this week isexpected to **insert** a dies that summing support for a relativelyobscure version of wi - **fi** , **analysts** said on monday , in a movethat could help ease congestion on **wireless** networks . $F(\boldsymbol{s}, \text{“Sci/Tech”}) = 0.95$ **PCC**: 0.28 *SemS*: 0.91, r: 4

Figure 1: Three examples of fragile attribution maps in text sequence classifiers. In each row, careful alteration of the original sample results in significantly different attribution maps while maintaining the prediction confidence F in the correctly predicted class. Red words have positive attribution values, i.e. contribute *towards* the true class, while blue words with negative attributions *against* it. Our novel CEA attack yields perturbed samples that have lower *Pearson Correlation Coefficient* (PCC) values between the words highlighted by the attribution method in the original and perturbed inputs, as well as higher semantic similarity values (SemS) of the original and adversarial sentences, compared to the baseline TEF attack. This results in higher estimated robustness constants r (see Section 4), thus lower robustness of the classifiers against attacks.

focusing on methods to find perturbations that maximize the change in attribution while being as imperceptible as possible. Characterizing and quantifying the robustness of attribution methods is an important step towards training robust classifiers and attribution methods that can be deployed in a wide variety of critical real-life use cases. We summarize our contributions as follows:

- We are the first to introduce a definition of attribution robustness (AR) in text classification that takes both the attribution distance and perceptibility of perturbations into account.
- We propose a diverse set of metrics to effectively capture aspects like semantic distance to original, smoothness and grammaticality of perturbed inputs. This is key to understand the perceptibility of small adversarial input perturbations in text.
- We introduce a novel and powerful attack algorithm, CONTEXT-AWAREEXPLANATIONATTACK (CEA), which is shown to consistently outperform state-of-the-art adversaries and therefore allows us to more accurately estimate attribution robustness in text classifiers.

- We are the first to utilize masked language models (MLMs) for context-aware candidate extraction in attribution robustness estimation. This is important because domain-specific MLMs are becoming increasingly available, making them a progressively attractive alternative to less effective, custom synonym embeddings on which current estimation methods have to rely.
- We successfully speed up robustness estimation with the usage of distilled language models and batch masking.

2 Related work

The robustness aspect of faithful explanations (Jacovi & Goldberg, 2020) has recently been studied with increasing interest. The authors Ghorbani et al. (2019) were the first to show that attribution methods like Integrated Gradients (Sundararajan et al., 2017) and DeepLIFT (Shrikumar et al., 2017), amongst others, lack robustness to local, imperceptible perturbations in the input that lead to significantly altered attribution maps while maintaining the correct prediction of the image classifier. The works of Dombrowski et al. (2019), Chen et al. (2019), Moosavi-Dezfooli et al. (2019), Rigotti et al. (2022) and Ivankay et al. (2021) have further studied this phenomenon and established theoretical frameworks to understand and mitigate the lack of attribution robustness in the image domain.

However, explanation robustness in natural language processing has not been explored as deeply. The authors Jain & Wallace (2019) and Wiegreffe & Pinter (2020) show that similar inputs can lead to similar attention values but different predictions, and that models can be retrained to yield different attention values for identical inputs and outputs. This, however, does not directly contradict the prediction assumption of faithfulness (Jacovi & Goldberg, 2020) as discussed by Wiegreffe & Pinter (2020). Closer to our work, the works of Ivankay et al. (2022) and Sinha et al. (2021) are the first to prove that explanations in text classifiers are also susceptible to input changes in a very small local neighbourhood of the input. Ivankay et al. (2022) introduce TextExplanationFooler (TEF) as a baseline to alter attributions and estimate local robustness of attributions in text. However, the authors' definition of AR does not take semantic distances between original and adversarial samples into account. Moreover, it draws token substitution candidates from a separately trained custom synonym embedding. Thus, their attack results in out-of-context and non-fluent adversarial samples, rendering such perturbations easily detectable. Our work aims to improve the imperceptibility of input alterations and estimate AR with less detectable adversarial alterations that change attributions to a greater extent.

3 Preliminaries

A text dataset $\mathbb{S}$ is comprised of N text samples $\boldsymbol{s}_i$, each containing a series of tokens w_i from a vocabulary $\mathbb{W}$ and labels l_i drawn from the label set $\mathbb{L}$. A text classifier F is a function that maps each sample $\boldsymbol{s}_i$ to a label $y_i \in \mathbb{L}$. It consists of an embedding function E and a classifier function f. The embedding function $E : \mathbb{S} \to \mathbb{R}^{d \times p}$, $E(\boldsymbol{s}) = \boldsymbol{X}$ maps the text samples $\boldsymbol{s}_i$ to a continuous embedding $\boldsymbol{X}_i$, while the classifier function $f : \mathbb{R}^{d \times p} \to \mathbb{R}^{|\mathbb{L}|}$, $f(\boldsymbol{X}) = \boldsymbol{o}$ maps the embeddings to the output probabilities for each class.

An *attribution function* $A(\boldsymbol{s}, F, l) = \boldsymbol{a}$ assigns a real number to each token w_j in sample $\boldsymbol{s}$. This represents the tokens influence towards the classification outcome. A positive value represents a token that is deemed relevant *towards* the label l, a negative value *against* it. We consider the attribution methods Saliency (S) (Simonyan et al., 2013), Integrated Gradients (IG) (Sundararajan et al., 2017) and Self-Attention (A) (Bahdanau et al., 2015).

The *perplexity* (Brown et al.) of a text sample $\boldsymbol{s}$ with tokens w_j, given a language model L, measures how well the probability distribution given by L predicts the sample $\boldsymbol{s}$, as defined in Equation (1):

$$PP(\boldsymbol{s}|L) = 2^{-\sum_{w_j \in \boldsymbol{s}} p(w_j|L,\boldsymbol{s}) \log p(w_j|L,\boldsymbol{s})} \tag{1}$$

where PP denotes the perplexity of the text sample $\boldsymbol{s}$ and $p(w_j|L, \boldsymbol{s})$ the probability of token w_j given L and $\boldsymbol{s}$. Low perplexity values indicate that the model L has captured the true distribution of the text dataset $\mathbb{S}$ well.

Sentence encoders are embedding functions $E_s : \mathbb{S} \to \mathbb{R}^m,\ E_s(\boldsymbol{s}) = \boldsymbol{e}$ that assign a continuous embedding vector of dimension m to each text sample (Reimers & Gurevych, 2019). These embeddings are used to capture higher-level representations of sentences or short paragraphs that can be used to train downstream tasks effectively. As they are jointly trained on a diverse set of multi-task problems, they are argued to capture the semantic meaning of the text well Reimers & Gurevych (2019).

4 Attribution Robustness

In this section, we introduce our new definition of attribution robustness (AR) in text classifiers. We describe our attribution and text distance measures. Furthermore, we introduce the optimization problem of estimating AR, our threat model as well as our novel estimator algorithm.

4.1 Attribution Robustness in Text

Most related works define AR as the maximal attribution distance with a given locality constraint in the search space (Ivankay et al., 2022; Sinha et al., 2021). We argue that this is potentially problematic, as the extent of the input perturbation is not taken into account. Two adversarial samples with similarly altered attributions might in fact strongly differ in terms of how well they maintain semantic similarity to the original sample (see e.g. 3rd example in Figure 1). This suggests that a proper measure of attribution robustness should ascribe higher robustness to methods that are only vulnerable to larger perturbations while being impervious to imperceptible ones. Thus, we define attribution robustness for a given text sample $\boldsymbol{s}$ with true and predicted label l as functions of both resulting attribution distance and input perturbation size, written in Equation (2).

$$r(\boldsymbol{s}) = \max_{\tilde{\boldsymbol{s}} \in \mathcal{N}(\boldsymbol{s})} \frac{d\big[A(\tilde{\boldsymbol{s}}, F, l),\ A(\boldsymbol{s}, F, l)\big]}{d_s(\tilde{\boldsymbol{s}}, \boldsymbol{s})} \tag{2}$$

with the constraint that the predicted classes of $\tilde{\boldsymbol{s}}$ and $\boldsymbol{s}$ are equal, written in Equation (3).

$$\underset{i \in \{1 \dots |\mathbb{L}|\}}{\arg\max}\, F_i(\tilde{\boldsymbol{s}}) = \underset{i \in \{1 \dots |\mathbb{L}|\}}{\arg\max}\, F_i(\boldsymbol{s}) \tag{3}$$

Here, d denotes the distance between attribution maps $A(\tilde{\boldsymbol{s}}, F, l)$ and $A(\boldsymbol{s}, F, l)$, F the text classifier with output probability F_i for class i, and d_s the distance of input text samples $\tilde{\boldsymbol{s}}$ and $\boldsymbol{s}$. $\mathcal{N}(\boldsymbol{s})$ indicates a neighbourhood of $\boldsymbol{s}$: $\{\mathcal{N}(\boldsymbol{s}) = \tilde{\boldsymbol{s}} \mid d_s(\tilde{\boldsymbol{s}}, \boldsymbol{s}) < \varepsilon\}$ for a small ε. This definition is inspired by the robustness assumption of faithful explanations (Jacovi & Goldberg, 2020). The estimated robustness of an attribution method A on a model F then becomes the expected per-sample $r(\boldsymbol{s})$ on dataset $\mathbb{S}$, see Equation (4).

$$r(A, F) = \mathbb{E}_{\boldsymbol{s} \in \mathbb{S}}\big[r(\boldsymbol{s})\big] \tag{4}$$

We call this r the estimated *attribution robustness (AR) constant*. The robustness of attribution method A on the model F is *inversely proportional* to $r(A, F)$, as high values mean large attribution distances and small input perturbations, which indicates low robustness.

4.2 Distances in Text Data

In order to compute the attribution robustness constant r from Equation (4), the distance measures in the numerator and denominator of Equation (2) need to be defined. In explainable AI, it is often argued that only the relative rank between input features or tokens is important when explaining the outcome of a classifier, or even only the top few features. Users frequently focus on the features deemed most important to explain a decision and disregard the less important ones (Ghorbani et al., 2019; Ivankay et al., 2021; Dombrowski et al., 2019). Therefore, it is common practice (Sinha et al., 2021; Ivankay et al., 2022) to use correlation coefficients and top-k intersections as distance measures between attributions. For this reason, we utilize the Pearson correlation coefficient (PCC) (Pearson, 1895) as attribution distance $d\big[A(\tilde{\boldsymbol{s}}, F, l),\ A(\boldsymbol{s}, F, l)\big] = 1 - \dfrac{1 + \mathrm{PCC}\big[A(\tilde{\boldsymbol{s}}, F, l),\ A(\boldsymbol{s}, F, l)\big]}{2}$ of Equation (2).

The denominator in Equation (2) contains the distance between original and adversarial text samples. In textual input domains, measuring distance between inputs in the adversarial setting is not as straightforward as in the image domain, where ℓ_p-norm induced distances are common. String distance metrics (Navarro, 2001) can only be used limitedly, as two words can have similar characters but entirely different semantics. For this reason, we propose the following set of measures to effectively capture smoothness, semantic distance to original, and correctness of grammar of adversarial text inputs.

First, we utilize pretrained sentence encoders to measure the semantic textual similarity between the original and adversarial text samples. This can be computed by the cosine similarity between the sentence embeddings of the two text samples, given as

$$d_{\mathrm{s}}(\tilde{\boldsymbol{s}}, \boldsymbol{s}) = 1 - \frac{s_{cos}[E_s(\tilde{\boldsymbol{s}}), E_s(\boldsymbol{s})] + 1}{2} \tag{5}$$

where d_{s} denotes the semantic distance between samples $\tilde{\boldsymbol{s}}$ and $\boldsymbol{s}$, s_{cos} the cosine similarity, and $E_s(\tilde{\boldsymbol{s}})$ and $E_s(\boldsymbol{s})$ the sentence embeddings of the two input samples. The semantic textual similarity provides a measure how close the two inputs are in their semantic meaning. To this end, the Universal Sentence Encoder (Cer et al., 2018) is widely-used in adversarial text setups (Sun et al., 2020; Ivankay et al., 2022). However, this architecture is not state-of-the-art on the STSBenchmark dataset (Cer et al., 2017), a benchmark used to evaluate semantic textual similarity. Therefore, we utilize a second sentence encoder architecture trained by the authors Wang et al. (2020), MiniLM. This model achieves close to state-of-the-art performance on the benchmark while maintaining a low computational cost.

Our second input distance is derived from the perplexity of original and adversarial inputs $\tilde{\boldsymbol{s}}$ and $\boldsymbol{s}$. We capture the relative increase of perplexity when perturbing the original sentence $\boldsymbol{s}$, given the pretrained GPT-2 language model (Radford et al., 2019) (Equation 6).

$$d_s(\tilde{\boldsymbol{s}}, \boldsymbol{s}) = \frac{PP(\tilde{\boldsymbol{s}}|L) - PP(\boldsymbol{s}|L)}{PP(\boldsymbol{s}|L) + \varepsilon} \tag{6}$$

where d_{s} denotes the distance between inputs $\tilde{\boldsymbol{s}}$ and $\boldsymbol{s}$, PP the perplexity of the text sample given the GPT-2 language model L and ε is a small constant. Intuitively, this measure indicates how natural the resulting adversarial inputs are.

Lastly, we capture the increase of grammatical errors in the input samples using the LanguageTool API [1]. As grammatical errors are easily perceived by the human observer, they significantly contribute to the perceptibility of adversarial perturbations (Ebrahimi et al., 2018).

4.3 Context-Aware Robustness Estimation

Given our AR definition in Equation (2), in order to estimate the true robustness of an attribution method for a given model, all possible input sequences $\tilde{\boldsymbol{s}}$ within the neighborhood $\mathcal{N}$ of $\boldsymbol{s}$ would have to be checked, which is intractable. Therefore we restrict the search space to sequences $\tilde{\boldsymbol{s}}$ that only contain token substitutions from the predefined vocabulary set $\mathbb{W}$. Moreover, we restrict the ratio of substituted tokens in the original sequence to ρ_{max}, considering only $|\mathbb{C}|$ number of possible substitutions for each token in $\boldsymbol{s}$. The number $|\mathbb{C}|$ is chosen to yield high attribution distance while keeping the computation cost low, detailed in Section 5. This way, we reduce the total perturbation set from $|\mathbb{W}|^{|\boldsymbol{s}|}$ to $|\mathbb{C}|^{|\boldsymbol{s}|\cdot\rho_{max}}$ samples. The adversarial sequence $\boldsymbol{s}_{\mathrm{adv}}$ then becomes the perturbed sequence that maximizes $r(\boldsymbol{s})$ from Equation (2)

We estimate AR with our novel Context-AwareExplanationAttack (CEA). CEA is a black-box attack, only having access to the model's prediction and the accompanying attributions, no intermediate representations or gradients. CEA consists of the following two steps.

Step 1: Word importance ranking. The first step extracts a priority ranking of tokens in the input text sample $\boldsymbol{s}$. For each word w_i in $\boldsymbol{s}$, CEA computes $I_{w_i} = d\big[A(\boldsymbol{s}_{w_i \to 0}, F, l),\ A(\boldsymbol{s}, F, l)\big]$, where $\boldsymbol{s}_{w_i \to 0}$ denotes the token w_i in $\boldsymbol{s}$ set to the zero embedding vector and d denotes the attribution distance measure

---

[1] https://languagetool.org

Algorithm 1 `Context-AwareExplanationAttack`

Input: Input sentence $\boldsymbol{s}$ with label l, classifier F, attribution A, attribution distance d, DistilBERT-MLM L, number of candidates N, maximum perturbation ratio ρ_{max}, batch masking ratio ρ_b

Output: Adversarial sentence $\boldsymbol{s}_{\text{adv}}$

```
$\boldsymbol{s}_{\text{adv}} \leftarrow \boldsymbol{s}$, $d_{max} \leftarrow 0$, $n \leftarrow 0$
for $w_i \in \boldsymbol{s}$ do
    $I_{w_i} = d\big[A(\boldsymbol{s}_{w_i \to 0}, F, l),\ A(\boldsymbol{s}, F, l)\big]$
$\boldsymbol{s}_B \leftarrow \langle \boldsymbol{s}_{1 \ldots b}, \boldsymbol{s}_{b+1 \ldots 2b}, \ldots, \boldsymbol{s}_{|\boldsymbol{s}|-b+1 \ldots |\boldsymbol{s}|} \rangle$ with $I_{w_{b-1}} \geq I_{w_b}\ \forall j \in \{2, \ldots, |\boldsymbol{s}_B|\}$ and $\forall b \in \{1, .., |\boldsymbol{s}_j|\}$
for $\boldsymbol{s}_b \in \boldsymbol{s}_B$ do
    $\mathbb{C}_{\mathbf{b}} \leftarrow L(\boldsymbol{s}_{b \to [MASK]}, \boldsymbol{s}_{\text{adv}})$
    for $w_j \in \boldsymbol{s}_b$ do
        if $w_j \in \mathbb{S}_{\texttt{Stopwords}}$ then
            continue
        for $c_k \in \mathbb{C}_j$ do
            $\tilde{\boldsymbol{s}}_{w_j \to c_k} \leftarrow$ Replace $w_j$ in $\boldsymbol{s}_{\text{adv}}$ with $c_k$
            if $\arg\max_{i \in \{1:|\mathbb{L}|\}} F(\tilde{\boldsymbol{s}}_{w_j \to c_k}) \neq l$ then
                continue
            $\tilde{d} = d\big[A(\tilde{\boldsymbol{s}}_{w_i \to c_k}, F, l), A(\boldsymbol{s}, F, l)\big]$
            if $\tilde{d} > d_{max}$ then
                $\boldsymbol{s}_{\text{adv}} \leftarrow \tilde{\boldsymbol{s}}_{w_i \to c_k}$
                $d_{max} \leftarrow \tilde{d}$
                $n \leftarrow n + 1$
        if $\rho = \frac{n+1}{|\boldsymbol{s}|} > \rho_{max}$ then
            break
```

in Equation (2), described in the previous subsection. The tokens in $\boldsymbol{s}$ are then sorted by descending values of I_{w_i}. Thus, we estimate words that are *likely* to result in large attribution distances and prioritize those for substitutions towards building explanation attacks.

Step 2: Candidate selection and substitution. The second step substitutes each highest ranked token in $\boldsymbol{s}$, computed in **Step 1**, with a token from a candidate set $\mathbb{C}$, in descending importance order. Each highest ranked token has its separate candidate set $\mathbb{C}$. CEA extracts these sets by masking the words and querying a transformer-based masked language model (MLM). In order to keep the computational costs low, we utilize the DistilBERT pretrained masked language model (Sanh et al., 2019), a BERT-MLM with significantly fewer parameters and more computationally efficient. Also, at most $n = \lfloor |\boldsymbol{s}| \cdot \rho_{max} \rfloor$ words are substituted.

In order to further reduce computational cost, CEA uses batch masking. Thus, instead of masking each token separately, the first $n_b = |\boldsymbol{s}| \cdot \rho_b$ most important tokens are masked at once and the language model is queried for candidates for all of these masked tokens. Here, n_b denotes the number, ρ_b the ratio of tokens in $\boldsymbol{s}$ to be masked at once. For instance, during AR estimation of a 100 word text sample, given $\rho_{max} = 0.15$ and $\rho_b = 0.05$, the MLM is queried only $(100 \cdot 0.15)/(100 \cdot 0.05) = 3$ times with batch masking instead of $100 \cdot 0.15 = 15$ times without it. We compared the runtime of CEA using non-distilled (Devlin et al., 2019) and distilled (Sanh et al., 2019) BERT MLMs, with and without batch masking, and found considerable performance increase with batch masking and distillation. The results are reported in Section 5.

5 Experiments

In this section, we present our AR estimation experiments. Specifically, we describe the evaluation setup and results with our novel robustness definition. We show that CEA consistently outperforms our direct state-of-the-art competitor, TextExplanationFooler (TEF) in terms of the attribution robustness constant r described in Section 4. Thus, we convey that CEA extracts smoother adversarial samples that are able to

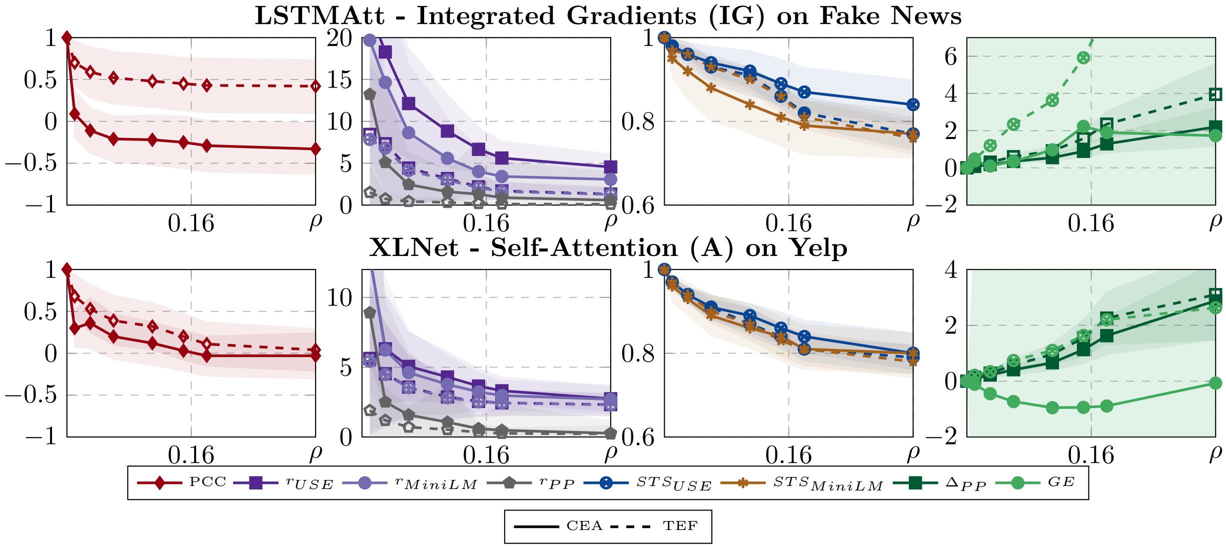


Figure 2: AR metrics as functions of the ratio of perturbed tokens ρ. We plot the mean and standard deviation of the Pearson correlations (PCC) between original and adversarial attributions, the estimated AR robustness constants (r), the semantic similarities (STS), relative perplexity increase (Δ_{PP}) and increase of number of grammatical errors (GE) in original and adversarial text inputs. We compare these values for our novel Context-AwareExplanationAttack (CEA - continuous lines) and the baseline TextExplanationFooler (TEF - dashed lines). We observe consistent improvement in robustness estimation with CEA compared to TEF, reflected in higher r-values in the second column. This is attributed to both lower PCC values, higher semantic similarities of perturbed sentences to the original ones and lower adversarial perplexity of CEA perturbations.

alter attributions more significantly than TEF. Finally, we compare the runtime of CEA to TEF and show that CEA achieves comparable runtimes, while still outperforming TEF in the previously mentioned aspects.

5.1 Setup

We evaluate the robustness constant r estimated by CEA on the AG's News (Zhang et al., 2015), MR Movie Reviews (Zhang et al., 2015), IMDB (Maas et al., 2011), Yelp (Asghar, 2016) and the Fake News datasets Lifferth (2018). We train a CNN, an LSTM, an LSTM with an attention layer (LSTMAtt), a finetuned BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019) and XLNet (Yang et al., 2019) classifier for each dataset. A description of these can be found in the appendix. We estimate the robustness of the Saliency (S), Integrated Gradients (IG) and Self-Attention (A) attribution methods. The CNN and LSTM architectures are used in combination with S and IG, the remaining LSTMAtt, BERT, RoBERTA and XLNet are used with all three attributions. Thus, we evaluate 16 combinations of models and attributions for each dataset.

We vary the ρ_{max} parameter of CEA between 0.01 and 0.4. A value of ρ_{max} does not necessarily lead to the actual perturbed ratio of tokens ρ to be $\rho = \rho_{max}$ due to the prediction constraint. We set the batch masking size $\rho_b = \min(\rho_{max}, 0.15)$, as the MLM was trained by masking ∼15% of the tokens (Sanh et al., 2019). We set $|\mathbb{C}| = 15$, as larger values do not result in better estimation in terms of r, but in significantly higher attack runtimes. This makes our experiments comparable to TEF (Ivankay et al., 2022).

Our attack and experiments are implemented in PyTorch (Paszke et al., 2019), utilizing the Hugging Face Transformer library (Wolf et al., 2020), Captum (Kokhlikyan et al., 2020) and SpaCy (Honnibal et al., 2020). We run each experiment on an NVIDIA A100 GPU with three different seeds and report the average results.

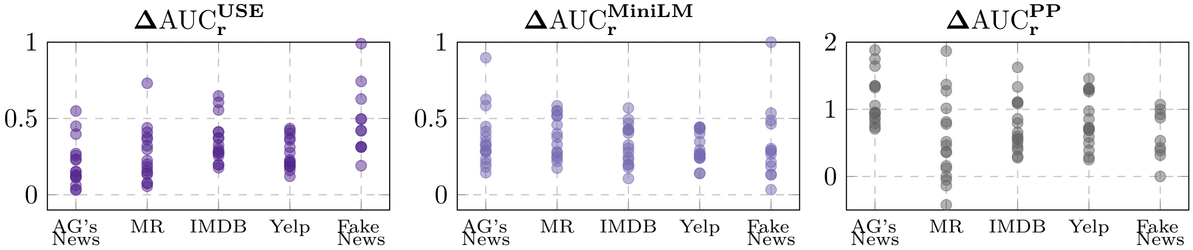


Figure 3: Relative increase Δ of AUC$_r$ when estimating the robustness constants r (Equation 4) with CEA compared to TEF. Each point corresponds to one of the 16 combinations of model and attribution method, on the indicated dataset. The r-values are estimated with the PCC as attribution similarity, varying the input distance measures d_s as described in Section 4.2. We observe a relative increase of $0.3 - 1.5$ for almost all models, attribution maps and datasets evaluated on. This shows that CEA consistently provides better perturbations that alter attributions more while being more fluent and semantically similar to the unperturbed input.

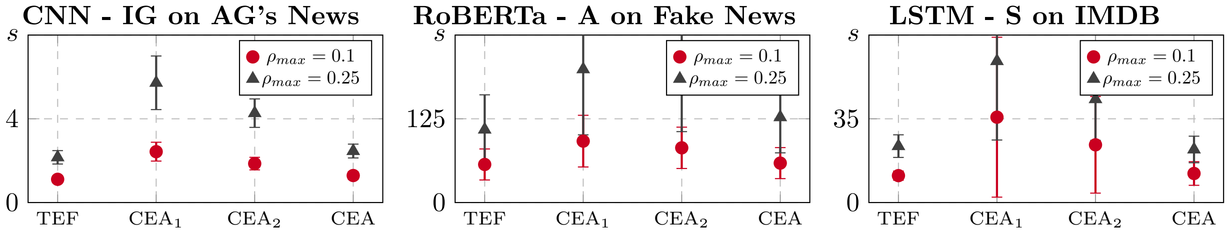


Figure 4: Per-sample runtime (s) of our AR estimator algorithm versions. CEA, with a distilled MLM and batch masking, achieves comparably fast estimation to TEF, while CEA with a non-distilled BERT MLM (CEA$_1$) is the slowest estimator, with a relative increase in runtime of approx. 1.5-2.5 compared to TEF. Distillation of the MLM (CEA$_2$) improves the runtime by around 25-35% compared to (CEA$_1$).

5.2 Results

We report the following metrics as functions of the true perturbed ratio ρ. The average PCC values of original and adversarial attribution maps indicate the amount of change in explanations. Lower values mean larger attribution changes. The input distance between text samples is captured by the semantic textual similarity values of the original and adversarial samples, measured by the cosine similarity between the USE (Cer et al., 2018) and MiniLM (Wang et al., 2020) sentence embeddings (STS_{USE} and STS_{MiniLM}), as well as the relative perplexity increase (Δ_{PP}). The average increase in number of grammatical errors (GE) after perturbation is also reported. Using these values, we report the estimated robustness constants r_{USE}, r_{MiniLM} and r_{PP}, according to Equation (4). We compare these metrics for our novel CEA algorithm and the direct competitor TEF (Ivankay et al., 2022). The results are reported in Figure 2. The continuous lines contain the metrics for our CEA attack, the dashed lines for the baseline TEF. The figures show that CEA perturbations alter explanations more (lower PCC values) while yielding adversarial samples semantically equally or more similar to the original inputs than TEF (higher average STS, lower average Δ_{PP} and GE values). Moreover, the perplexity increase is consistently lower for CEA perturbations, leading to more fluent adversarial samples. This is well-captured by resulting robustness constants r, which are higher for CEA than TEF, showing both that our AR definition of Equation (2) is a suitable indicator for AR in text classifiers, and that CEA estimates this robustness better than the state-of-the-art TEF attack. The rest of the results is reported in the appendix.

To quantify the overall performance of CEA over the whole operation interval of ρ, we compute the area under the estimated r curves (2$^{\text{nd}}$ column in Figure 2). These are calculated as the integral AUC$_r = \int_\rho r(A, F) d\rho$. High AUC$_r$ values correspond to high r-values, thus low overall attribution robustness. We then compare the resulting AUC$_r$ estimated with our CEA algorithm to the competitor method TEF. Figure 3 shows the relative increase of AUC when estimating with CEA rather than TEF, for each of the 16 combinations of models and attribution methods for a given dataset. For instance, a value of 0.5 indicates an increase of 50% in estimated AUC_r, i.e. if TEF results in $\text{AUC}_r = 1.0$, CEA yields $\text{AUC}_r = 1.5$. We plot the AUC_r increase estimated with the semantic textual similarities from USE ($\text{AUC}_r^{\textbf{USE}}$), MiniLM ($\text{AUC}_r^{\textbf{MiniLM}}$) and with the relative perplexity increase ($\text{AUC}_r^{\textbf{PP}}$). The attribution distance in the numerator of r is set to the PCC, described in Section 4. We observe an increase in AUC_r of $0.3 - 0.5$ with USE and MiniLM, and $0.5 - 1.5$ with PP for most models, attribution maps and datasets. This further shows that CEA consistently yields higher robustness constants r than TEF, providing better perturbations that alter attributions more while being less perceptible.

Querying transformer-based MLMs is computationally expensive. Substituting the synonym extraction from TEF with an MLM-based candidate extraction results in a significant increase in estimation time. Therefore, we use the methods described in Section 4 to lower the estimation time in CEA. Figure 4 contains the per-sample attack time for TEF, CEA with the non-distilled BERT MLM (CEA_1), CEA with DistilBERT MLM (CEA_2) and our CEA algorithm with DistilBERT MLM and batch masking, for $\rho_{max} \in \{0.1, 0.25\}$. We observe that CEA_1 results in a significant increase in mean estimation time by a factor of around 2 compared to TEF on both a smaller, medium and a large datasets. Using CEA_2 for estimating AR decreases the runtime by a large margin compared to CEA_1. Finally, when applying both a distilled MLM and batch masking - CEA, the per-sample attack time is comparable to the baseline TEF, while maintaining better AR estimation.

6 Conclusion

In this work, we introduced a novel definition of attribution robustness in text classifiers. Crucially, our definition incorporates perturbation size, which contributes significantly to the perceptibility of attacks. We introduce semantic textual similarity measures, the relative perplexity increase and the number of grammatical errors as ways to effectively quantify perturbation size in text. Next, we introduced ContextAwareExplanationAttack, a new state-of-the-art attack method that results in a tighter estimator for attribution robustness in text classification problems. It is a black-box estimator using a distilled MLM with batch masking to extract adversarial perturbations with small computational overhead. Finally, we showed that our new algorithm CEA outperforms current attacks by altering DNN attributions more with less perceptible perturbations. Our contributions allow for estimating the robustness of attributions more accurately and are a first step towards training robust, safely applicable DNNs in critical areas like medicine, law or finance.

References

N. Asghar. YELP Dataset Challenge: Review Rating Prediction. *arXiv preprint arXiv:1605.05362*, 2016.

D. Bahdanau, K. H. Cho, and Y. Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. In *International Conference on Learning Representations*, 2015.

P. E. Brown, V. J. Della Pietra, S. A. Della Pietra, and J. C. Lai. An Estimate of an Upper Bound for the Entropy of English. *Computational Linguistics*, 18(1).

D. Cer, M. Diab, E. Agirre, I. Lopez-Gazpio, and L. Specia. SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation. In *International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 1–14, 2017.

D. Cer, Y. Yang, S.-Y. Kong, N. Hua, N. Limtiaco, R. St John, N. Constant, M. Guajardo-Céspedes, S. Yuan, and C. Tar. Universal Sentence Encoder. *arXiv preprint arXiv:1803.11175*, 2018.

J. Chen, X. Wu, V. Rastogi, Y. Liang, and S. Jha. Robust Attribution Regularization. In *Advances in Neural Information Processing Systems*, pp. 14300–14310, 2019.

J. Devlin, M.-W. Chang, L. Kenton, and L. K. Toutanova. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*, pp. 4171–4186, 2019.

A.-K. Dombrowski, M Alber, C. Anders, M. Ackermann, K.-R. Müller, and P. Kessel. Explanations can be Manipulated and Geometry is to blame. In *Advances in Neural Information Processing Systems*, pp. 13589–13600, 2019.

J. Ebrahimi, A. Rao, D. Lowd, and D. Dou. HotFlip: White-Box Adversarial Examples for Text Classification. In *Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 31–36, 2018.

A. Ghorbani, A. Abid, and J. Zou. Interpretation of Neural Networks is Fragile. In *AAAI Conference on Artificial Intelligence*, volume 33, pp. 3681–3688, 2019.

I. Girardi, P. Ji, A.-P. Nguyen, N. Hollenstein, A. Ivankay, L. Kuhn, C. Marchiori, and C. Zhang. Patient Risk Assessment and Warning Symptom Detection Using Deep Attention-Based Neural Networks. In *International Workshop on Health Text Mining and Information Analysis*, pp. 139–148, 2018.

M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd. spaCy: Industrial-strength Natural Language Processing in Python, 2020. URL `https://doi.org/10.5281/zenodo.1212303`.

A. Ivankay, I. Girardi, C. Marchiori, and P. Frossard. FAR: A General Framework for Attributional Robustness. *The 32nd British Machine Vision Conference*, 2021.

A. Ivankay, I. Girardi, C. Marchiori, and P. Frossard. Fooling Explanations in Text Classifiers. In *International Conference on Learning Representations*, 2022.

A. Jacovi and Y. Goldberg. Towards Faithfully Interpretable NLP Systems: How Should We Define and Evaluate Faithfulness? In *Annual Meeting of the Association for Computational Linguistics*, pp. 4198–4205, 2020.

S. Jain and B. C. Wallace. Attention is not Explanation. In *Proceedings of NAACL-HLT*, pp. 3543–3556, 2019.

N. Kokhlikyan, V. Miglani, M. Martin, E. Wang, B. Alsallakh, J. Reynolds, A. Melnikov, N. Kliushkina, C. Araya, and S. Yan. Captum: A Unified and Generic Model Interpretability Library for PyTorch. *arXiv preprint arXiv:2009.07896*, 2020.

W. Lifferth. Fake News, 2018. URL `https://kaggle.com/competitions/fake-news`.

Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692*, 2019.

A. Maas, R. E Daly, P. T Pham, D. Huang, A. Y. Ng, and C. Potts. Learning Word Vectors for Sentiment Analysis. In *Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150, 2011.

S.-M. Moosavi-Dezfooli, A. Fawzi, J. Uesato, and P. Frossard. Robustness via Curvature Regularization, and vice versa. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9078–9086, 2019.

G. Navarro. A Guided Tour to Approximate String Matching. *ACM Computing Surveys (CSUR)*, 33(1): 31–88, 2001.

A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, and L. Antiga. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *International Conference on Neural Information Processing Systems*, pp. 8026–8037, 2019.

K. Pearson. Notes on Regression and Inheritance in the Case of Two Parents. *Proceedings of the Royal Society of London*, 58(347-352):240–242, 1895.

A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language Models are Unsupervised Multitask Learners. *OpenAI blog*, 1(8):9, 2019.

N. Reimers and I. Gurevych. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Conference on Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, 2019.

M. Rigotti, C. Miksovic, I. Giurgiu, T. Gschwind, and P. Scotton. Attention-Based Interpretability with Concept Transformers. In *International Conference on Learning Representations*, 2022.

V. Sanh, L. Debut, J. Chaumond, and T. Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

A. Shrikumar, P. Greenside, and A. Kundaje. Learning Important Features through Propagating Activation Differences. In *International Conference on Machine Learning*, pp. 3145–3153. PMLR, 2017.

K. Simonyan, A. Vedaldi, and A. Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *arXiv preprint arXiv:1312.6034*, 2013.

S. Sinha, H. Chen, A. Sekhon, Y. Ji, and Y. Qi. Perturbing Inputs for Fragile Interpretations in Deep Natural Language Processing. In *BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pp. 420–434, 2021.

L. Sun, K. Hashimoto, W. Yin, A. Asai, J. Li, P. Yu, and C. Xiong. Adv-BERT: BERT is not Robust on Misspellings! Generating Nature Adversarial Samples on BERT. *arXiv preprint arXiv:2003.04985*, 2020.

M. Sundararajan, A. Taly, and Q. Yan. Axiomatic Attribution for Deep Networks. In *International Conference on Machine Learning*, volume 70, pp. 3319–3328, 2017.

W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. Zhou. MiniLM: Deep Self-Attention Distillation for Task-Agnostic Compression of Pre-Trained Transformers. *Advances in Neural Information Processing Systems*, 33:5776–5788, 2020.

S. Wiegreffe and Y. Pinter. Attention is not not Explanation. In *Conference on Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing, EMNLP-IJCNLP*, pp. 11–20. Association for Computational Linguistics, 2020.

T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush. Transformers: State-of-the-Art Natural Language Processing. In *Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, October 2020. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/2020.emnlp-demos.6`.

Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In *International Conference on Neural Information Processing Systems*, pp. 5753–5763, 2019.

X. Zhang, J. Zhao, and Y. Lecun. Character-Level Convolutional Networks for Text Classification. *Advances in Neural Information Processing Systems*, 2015:649–657, 2015.

A Appendix

A.1 Datasets

We estimate the robustness of our attribution methods and models on five publicly available datasets. These are AG's News, MR movie review, IMDB movie review, Yelp and Fake News, all of which are in English. AG's News consists of 127552 news article samples, categorized into the classes World, News, Business and Sci/Tech. We use the concatenation of title and text of the samples to feed into our text classifiers, stripping any sample that is longer than 64 tokens. The MR Movie Review dataset contains 10592 short samples of positive or negative movie reviews. We only use the first 32 tokens in each sample as input to the classifiers. IMDB Movie Review is a dataset consisting of 49952 positive and negative movie reviews, with a maximum token length of 256. Yelp categorizes 700000 reviews of several topics into 5 classes, each representing a rating from 1 to 5. We strip the samples to a maximum length of 256. Fake News is a collection of 20080 news samples, each categorized into reliable or unreliable. These are rather long articles, thus we use a maximum sequence length of 512 for this dataset.

We apply basic preprocessing to all samples in each dataset, which includes converting them to lowercase, removing any special characters not in the English alphabet and emojis. We use 60% of the samples for training the classifier models, 20% for validation and 20% for testing and estimating the robustness of attribution methods.

Dataset	CNN	LSTM	LSTMAtt	BERT	RoBERTa	XLNet
AG's News	89.7%	90.8%	91.4%	94.2%	94.0%	93.8%
MR	73.0%	76.4%	78.0%	82.2%	87.7%	86.3%
IMDB	82.0%	87.2%	87.3%	89.4%	93.3%	93.7%
Yelp	49.0%	54.8%	60.0%	62.6%	67.6%	-
Fake News	98.9%	99.6%	99.6%	99.8%	100.0%	100.0%

Table 1: Accuracies of each classifier trained. Our models achieve comparable results to state-of-the-art performance for each dataset.

A.2 Models

		AG's News	**MR**	**IMDB**	**Yelp**	**Fake News**
CNN	Input shape	(64, 300)	(32,300)	(256, 300)	(256, 300)	(512, 300)
	Num. classes	4	2	2	5	2
	Filter sizes	[3, 5, 7]	[3, 5]	[3, 5, 7]	[3, 5, 7]	[3, 5, 7]
	Feature sizes	[8, 8, 8]	[8, 8]	[16, 16, 16]	[128, 128, 128]	[32, 32, 32]
	Pooling sizes	[2, 2, 2]	[2, 2]	[2, 2, 2]	[2, 2, 2]	[2, 2, 2]
	Lin. layer dim.	8	8	16	64	32
	Num. params	67748	27946	567458	16428293	4091714
LSTM	Input shape	(64, 300)	(32,300)	(256, 300)	(256, 300)	(512, 300)
	Num. classes	4	2	2	5	2
	Hidden dim.	8	8	16	256	16
	Num. layers	1	1	2	2	1
	Pooling sizes	2	2	1	2	2
	Lin. layer dim.	8	8	16	32	16
	Num. params	10988	10458	18162	2146693	85986
LSTMAtt	Input shape	(64, 300)	(32,300)	(256, 300)	(256, 300)	(512, 300)
	Num. classes	4	2	2	5	2
	Hidden dim.	8	8	16	256	16
	Num. layers	4	1	2	2	1
	Lin. layer dim.	8	8	16	32	16
	Num. params	25004	19994	47666	2752901	41826
BERT	Input shape	(64,)	(32,)	(256,)	(256,)	(512,)
	Num. classes	4	2	2	5	2
	Model ID	bert-base-uncased				
	Num. params	109485316	109483778	109483778	109486085	109483778
RoBERTa	Input shape	(64,)	(32,)	(256,)	(256,)	(512,)
	Num. classes	4	2	2	5	2
	Model ID	roberta-base				
	Num. params	124648708	124647170	124647170	124649477	124647170
XLNet	Input shape	(64,)	(32,)	(256,)	(256,)	(512,)
	Num. classes	4	2	2	5	2
	Model ID	xlnet-base-cased				
	Num. params	117312004	117310466	117310466	117312773	117310466

Table 2: Model specifications

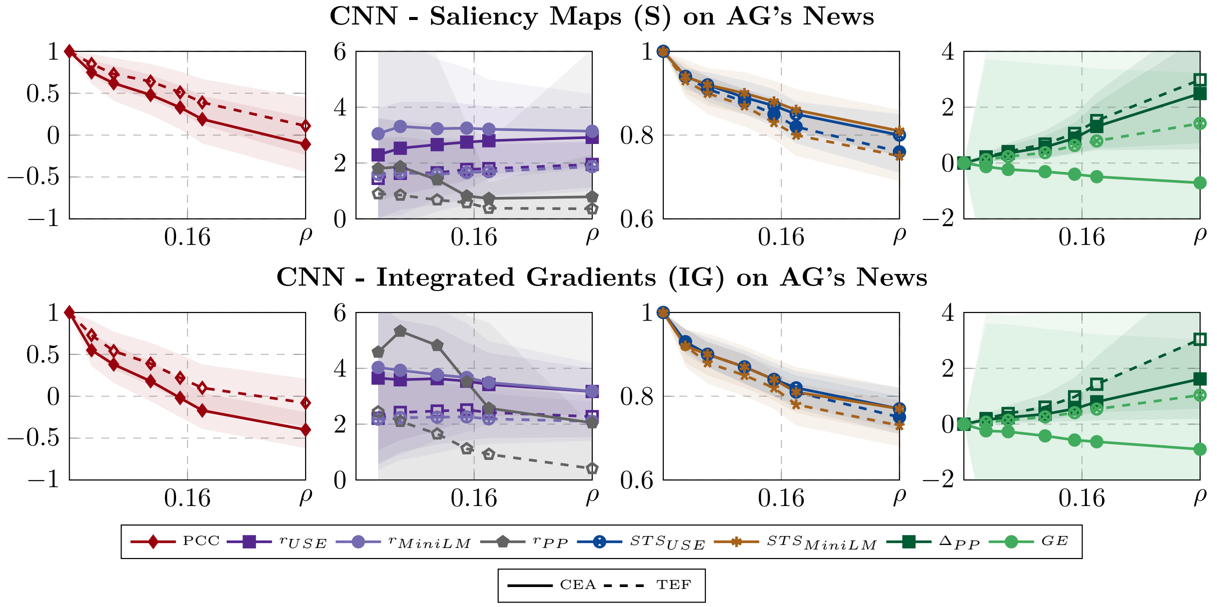
As described in the main paper, we train six classification architectures for each dataset, three DNN-based architectures, which are a CNN, an LSTM, an LSTM with an attention layer (LSTMAtt), as well as three transformer-based architectures, which are a finetuned BERT, RoBERTa and XLNet. The CNN, LSTM and LSTMAtt architectures use the 6B-300-dimensional Glove word embeddings, while the transformer-based architectures use the pretrained Hugging Face embeddings of the respective base-uncased versions. The

DNN-based classifiers each contain a linear layer on top of their feature extractors and use the built-in SpaCy English tokenizer, the transformers directly map the feature outputs to the output logits with a fully-connected layer and utilize the Hugging Face pretrained tokenizers for each architecture respectively. Table 2 contains the model specifications. We train each model with a standard learning rate of 0.001, using the Adam optimizer with the cross-entropy loss and early stopping. We utilize NVIDIA A100 GPUs to speed up training and AR estimation. The resulting accuracies of the models can be found in Table 1.

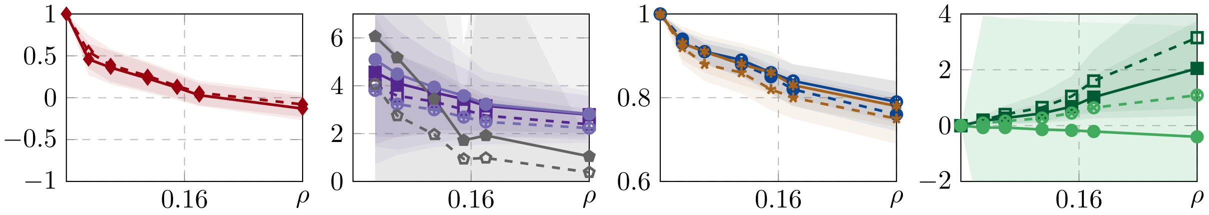
A.3 Additional AR results

As described in the main body of our paper, we plot the Pearson Correlation Coefficient between original and adversarial attribution values of the words (1^{st} column from left), the estimated robustness constants r (2^{nd} column from left) as well as the semantic similarities between unperturbed and perturbed input texts, the perplexity increase and the increase in number of grammatical errors (3^{rd} and 4^{th} column from left) after perturbation. We consider a high estimated robustness constant r as *successful* attack, thus low PCC values accompanied by high semantic similarities, low perplexity increase values and grammatical errors. Based on the graph below, we conclude that CEA consistently yields higher estimated robustness constants r than the reference method TEF, due to lower Pearson correlation between adversarial and original attribution maps, higher semantic similarities and smaller perplexity increases after applying the adversarial perturbations.

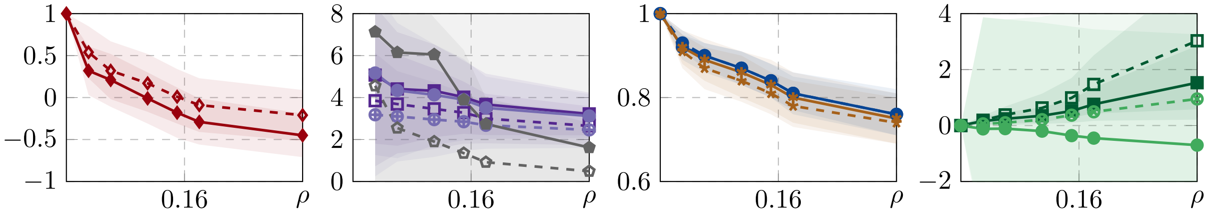
A.3.1 AG's News



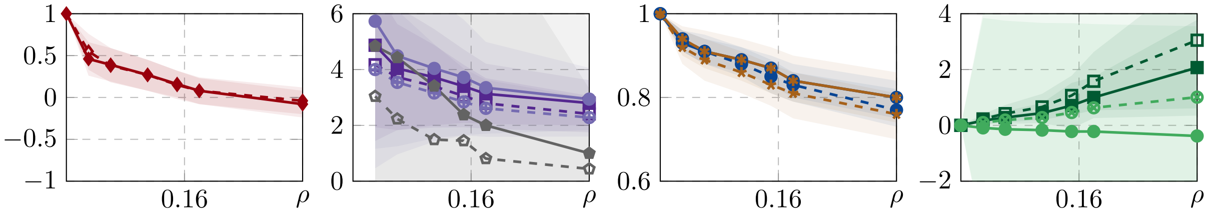
LSTM - Saliency Maps (S) on AG's News



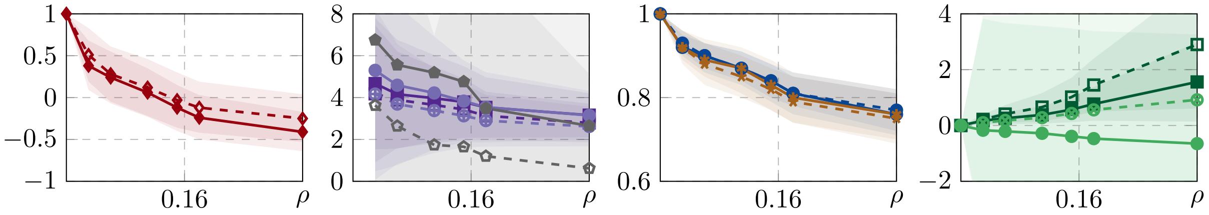
LSTM - Integrated Gradients (IG) on AG's News



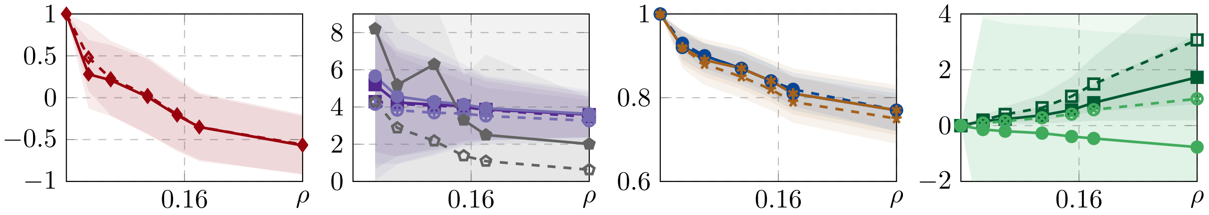
LSTMAtt - Saliency Maps (S) on AG's News



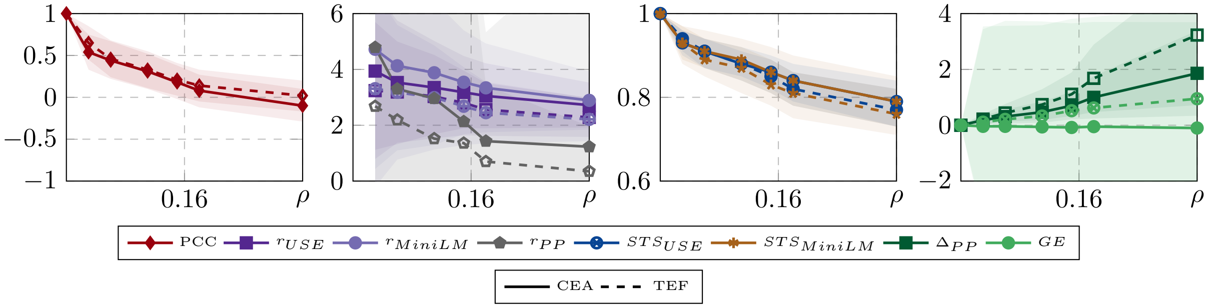
LSTMAtt - Integrated Gradients (IG) on AG's News

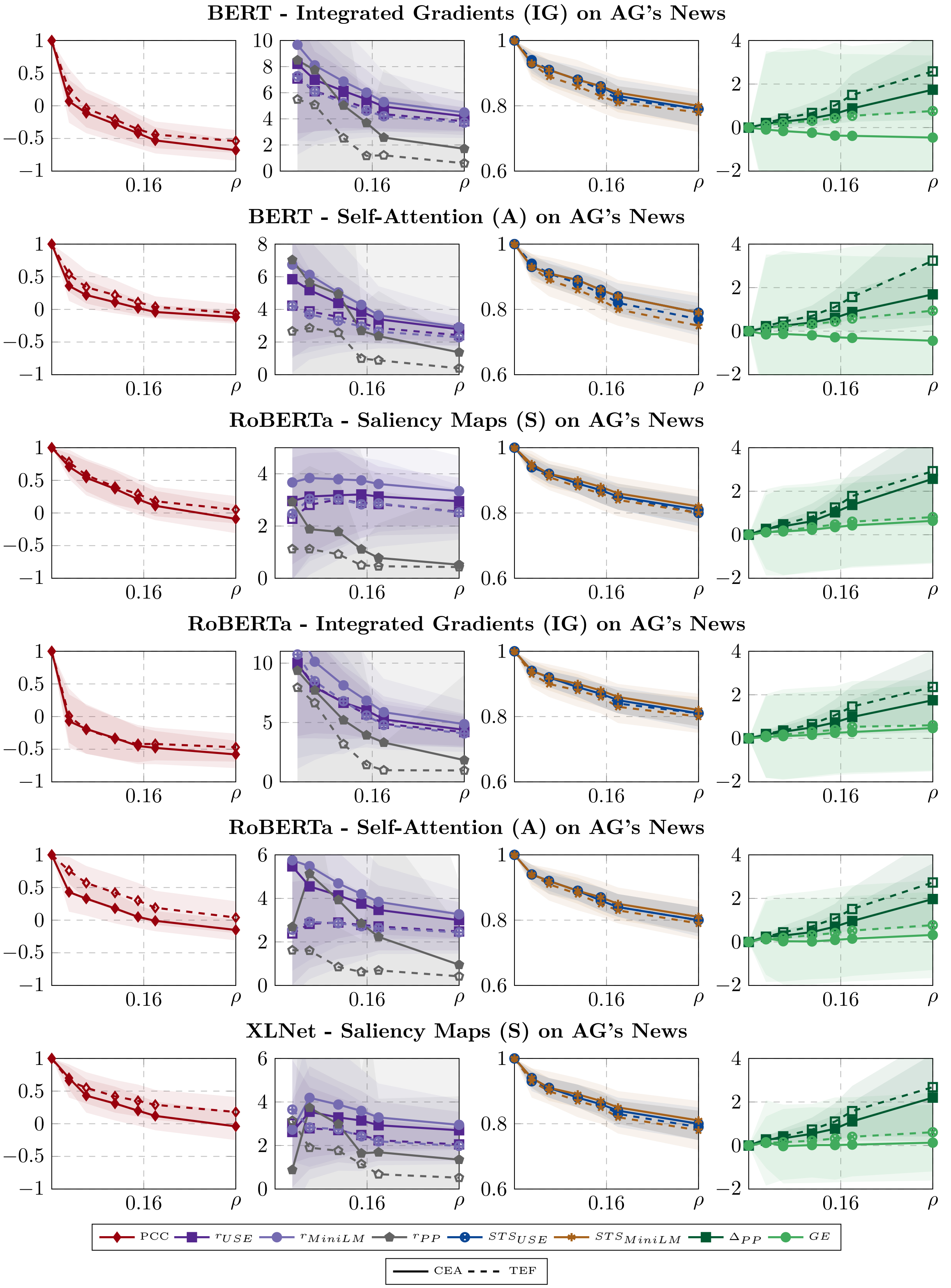


LSTMAtt - Self-Attention (A) on AG's News



BERT - Saliency Maps (S) on AG's News





BERT - Integrated Gradients (IG) on AG's News

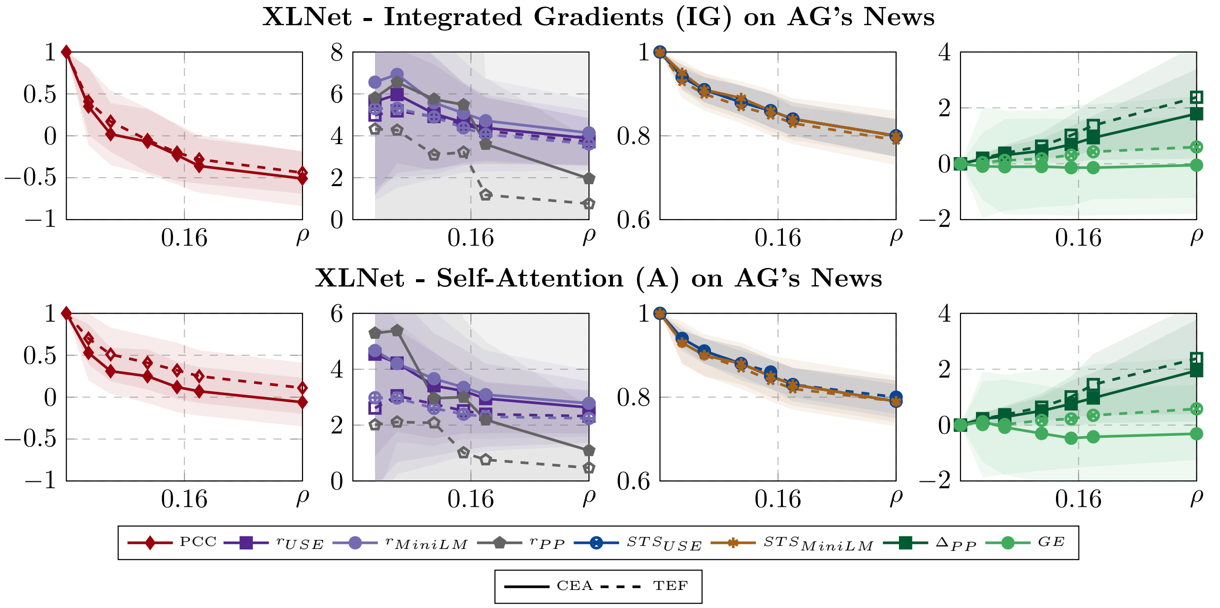
BERT - Self-Attention (A) on AG's News

RoBERTa - Saliency Maps (S) on AG's News

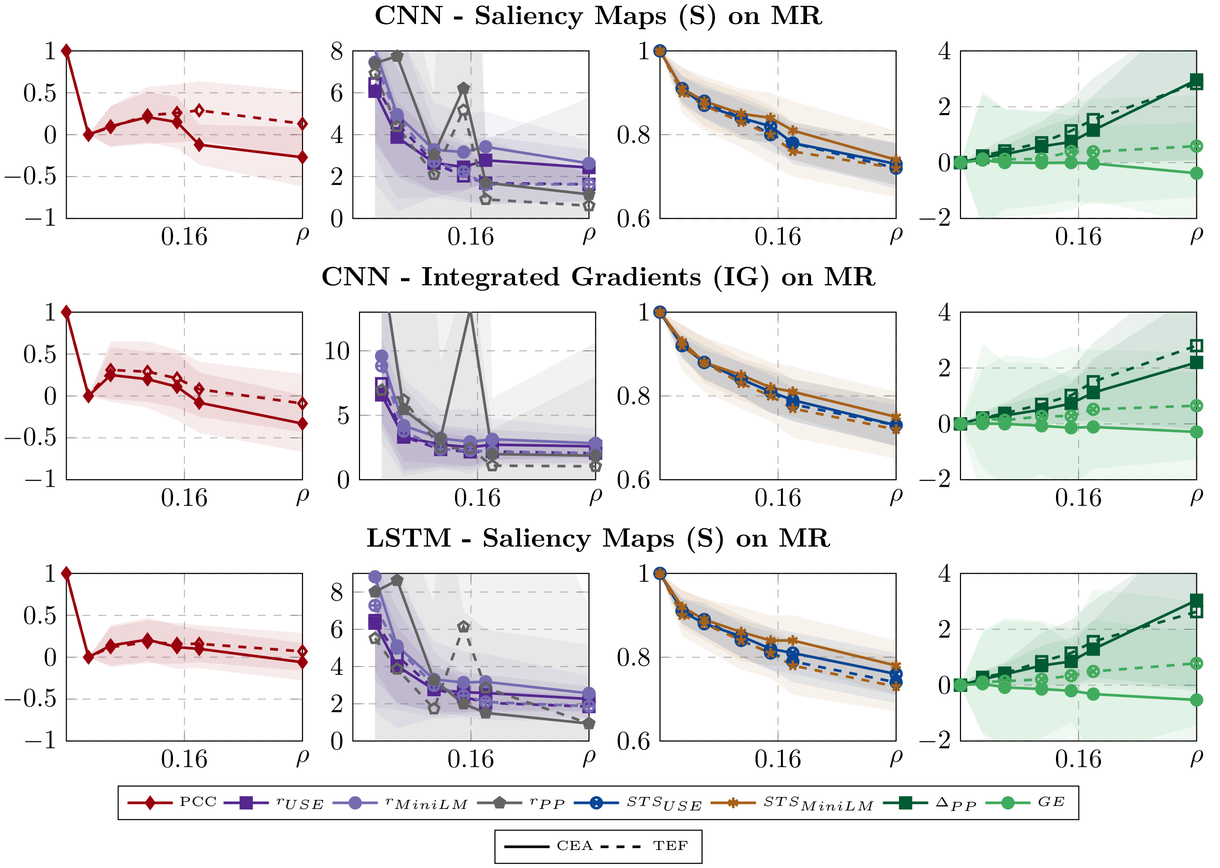
RoBERTa - Integrated Gradients (IG) on AG's News

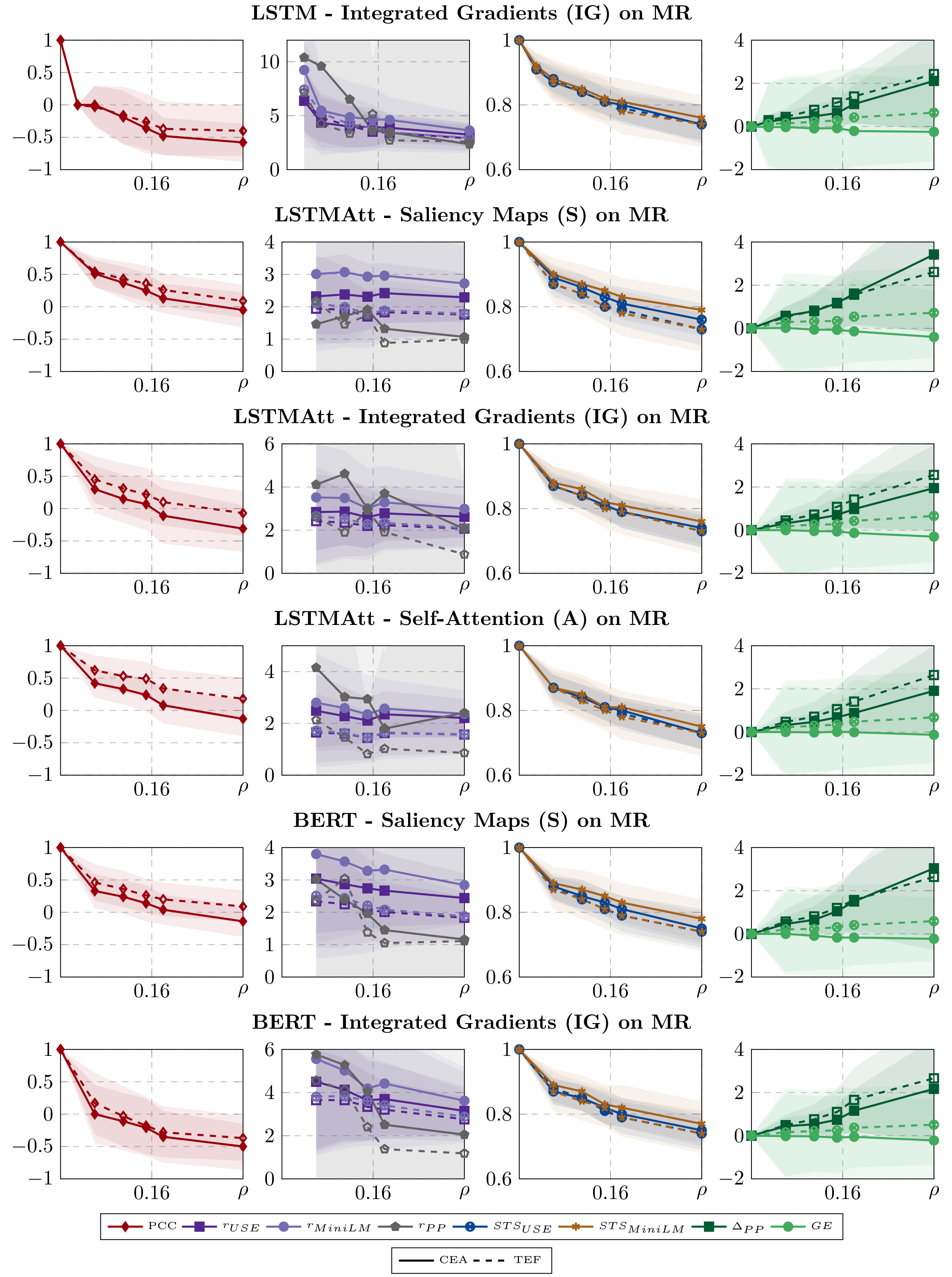
RoBERTa - Self-Attention (A) on AG's News

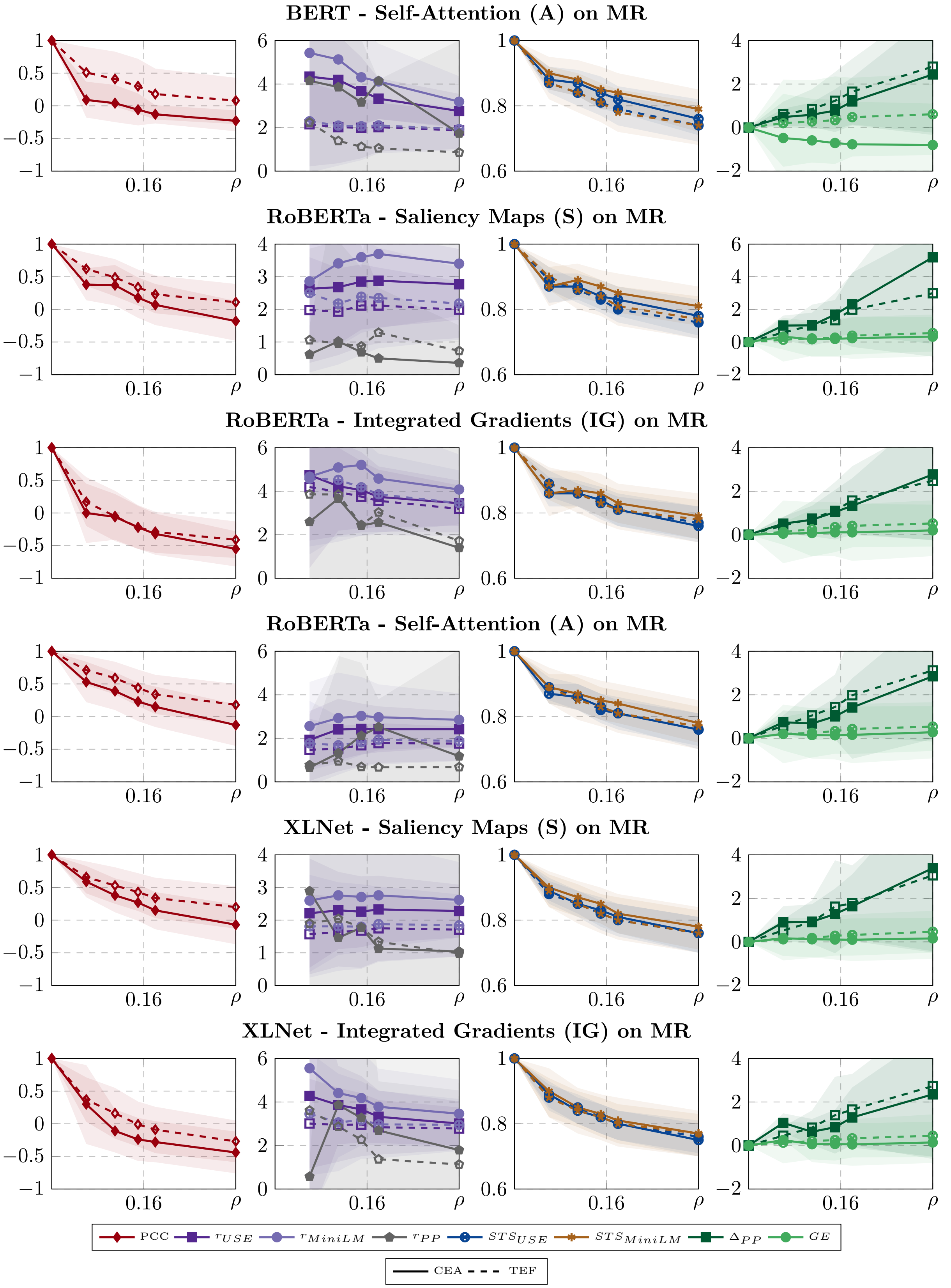
XLNet - Saliency Maps (S) on AG's News

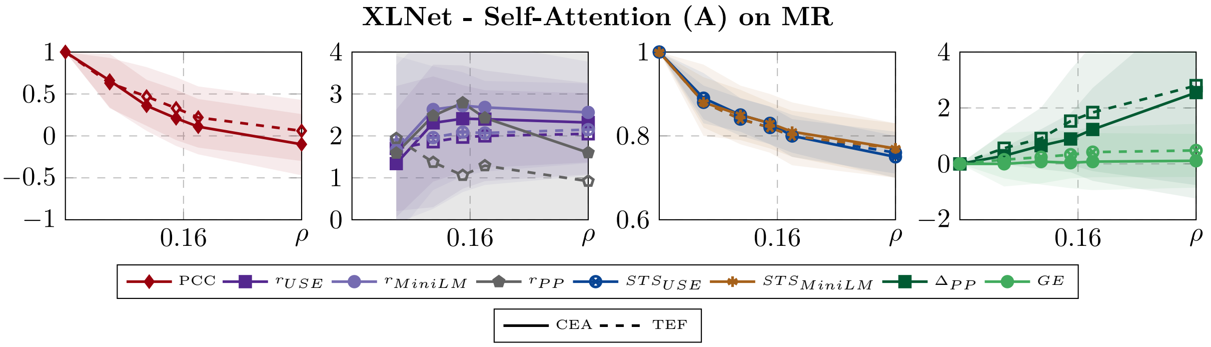

XLNet - Integrated Gradients (IG) on AG's News
XLNet - Self-Attention (A) on AG's News
1
0.5
0
−0.5
−1
8
6
4
2
0
0.8
0.6
−2
0.16
ρ
PCC
r_{USE}
r_{MiniLM}
r_{PP}
STS_{USE}
STS_{MiniLM}
Δ_{PP}
GE
CEA
TEF

A.3.2 MR


CNN - Saliency Maps (S) on MR
CNN - Integrated Gradients (IG) on MR
LSTM - Saliency Maps (S) on MR
1
0.5
0
−0.5
−1
10
8
6
5
4
2
0
0.8
0.6
−2
0.16
ρ
PCC
r_{USE}
r_{MiniLM}
r_{PP}
STS_{USE}
STS_{MiniLM}
Δ_{PP}
GE
CEA
TEF

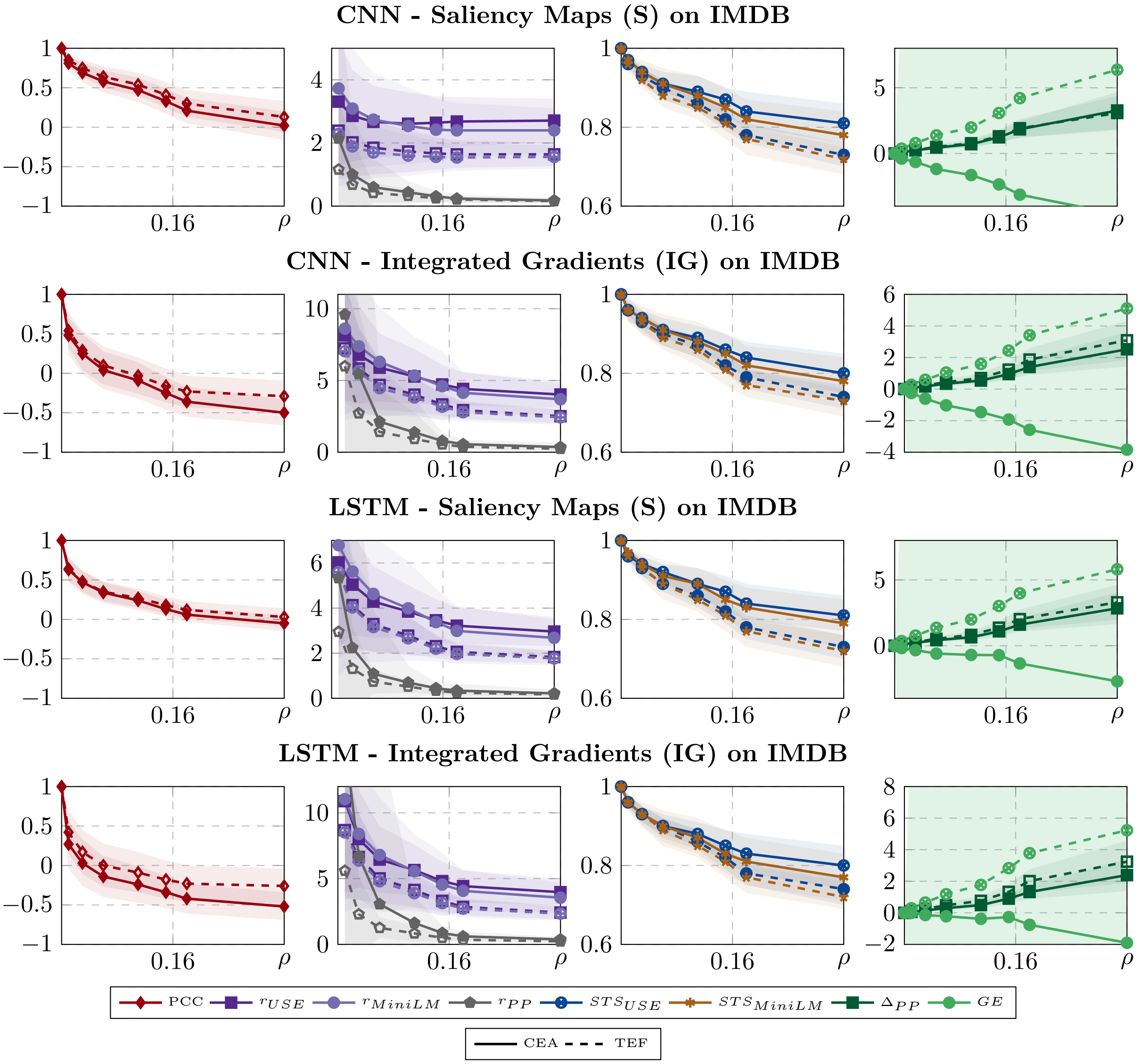

LSTM - Integrated Gradients (IG) on MR
LSTMAtt - Saliency Maps (S) on MR
LSTMAtt - Integrated Gradients (IG) on MR
LSTMAtt - Self-Attention (A) on MR
BERT - Saliency Maps (S) on MR
BERT - Integrated Gradients (IG) on MR
1
0.5
0
−0.5
−1
10
5
0
6
4
3
2
1
0.8
0.6
−2
0.16
ρ
PCC
r_{USE}
r_{MiniLM}
r_{PP}
STS_{USE}
STS_{MiniLM}
Δ_{PP}
GE
CEA
TEF


BERT - Self-Attention (A) on MR
RoBERTa - Saliency Maps (S) on MR
RoBERTa - Integrated Gradients (IG) on MR
RoBERTa - Self-Attention (A) on MR
XLNet - Saliency Maps (S) on MR
XLNet - Integrated Gradients (IG) on MR
1
0.5
0
−0.5
−1
6
4
2
0
0.8
0.6
−2
3
1
0.16
ρ
PCC
r_{USE}
r_{MiniLM}
r_{PP}
STS_{USE}
STS_{MiniLM}
Δ_{PP}
GE
CEA
TEF



XLNet - Self-Attention (A) on MR

A.3.3 IMDB

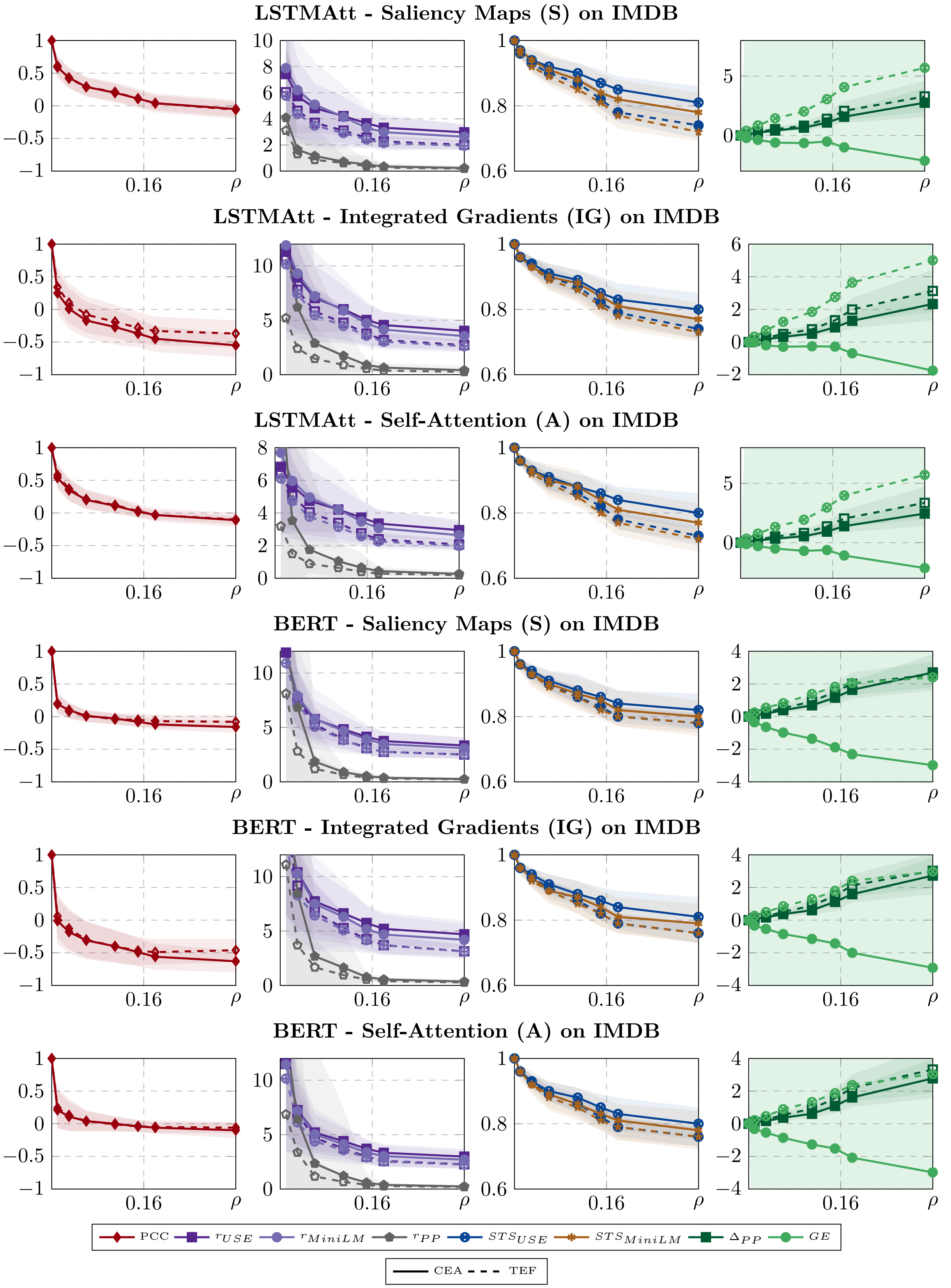


CNN - Saliency Maps (S) on IMDB

CNN - Integrated Gradients (IG) on IMDB

LSTM - Saliency Maps (S) on IMDB

LSTM - Integrated Gradients (IG) on IMDB



LSTMAtt - Saliency Maps (S) on IMDB

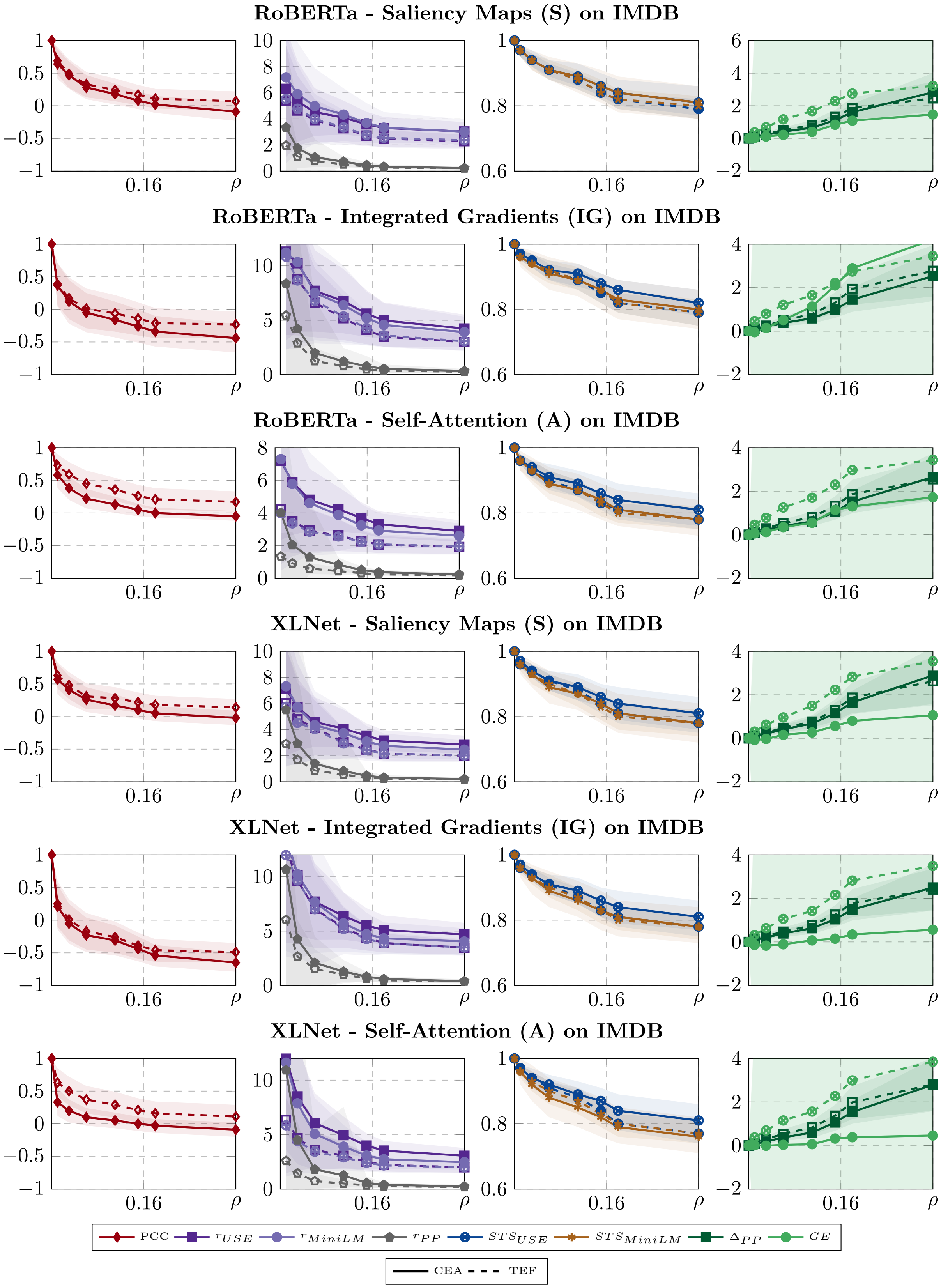
LSTMAtt - Integrated Gradients (IG) on IMDB

LSTMAtt - Self-Attention (A) on IMDB

BERT - Saliency Maps (S) on IMDB

BERT - Integrated Gradients (IG) on IMDB

BERT - Self-Attention (A) on IMDB



RoBERTa - Saliency Maps (S) on IMDB

RoBERTa - Integrated Gradients (IG) on IMDB

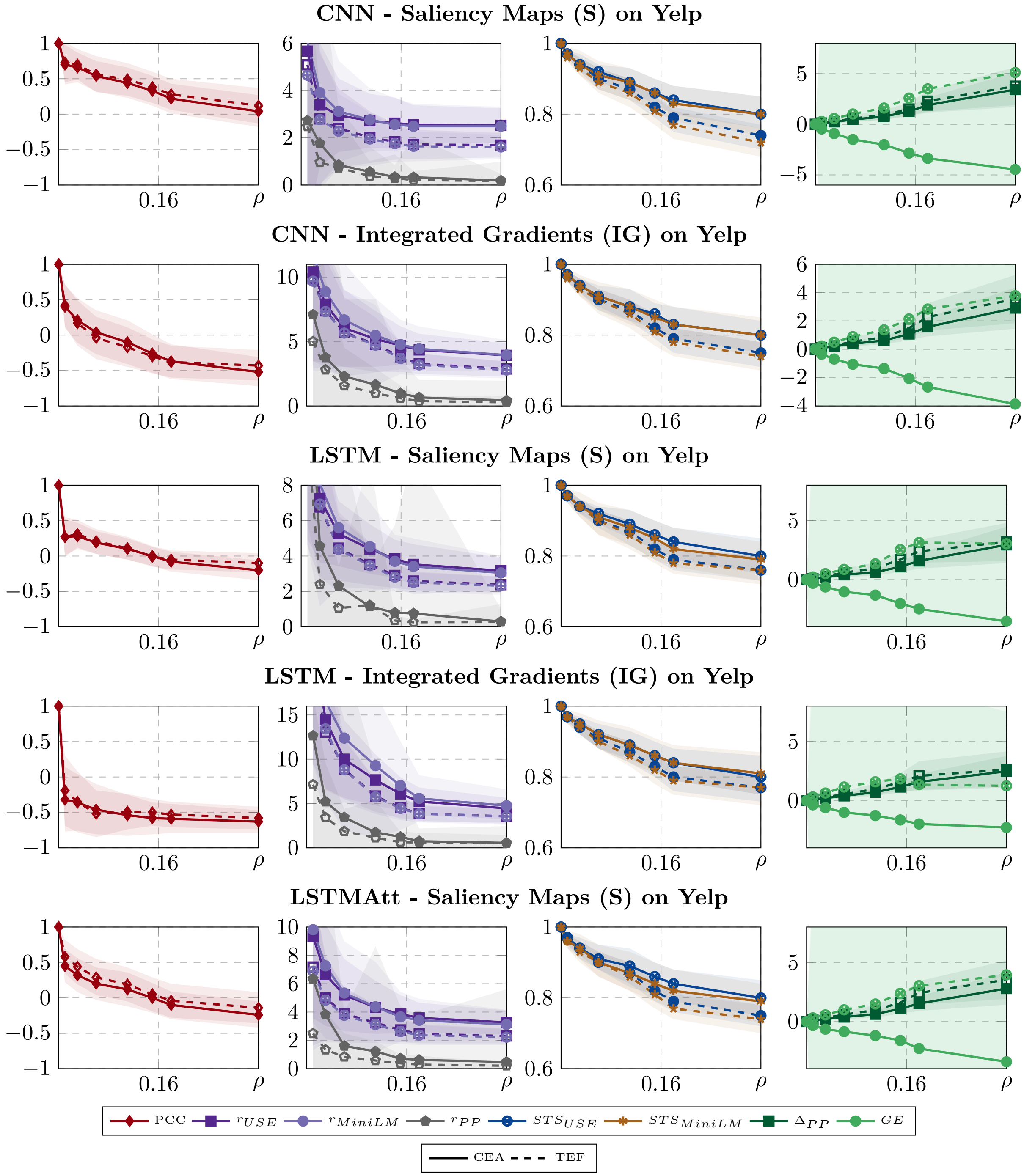
RoBERTa - Self-Attention (A) on IMDB

XLNet - Saliency Maps (S) on IMDB

XLNet - Integrated Gradients (IG) on IMDB

XLNet - Self-Attention (A) on IMDB

A.3.4 Yelp



CNN - Saliency Maps (S) on Yelp

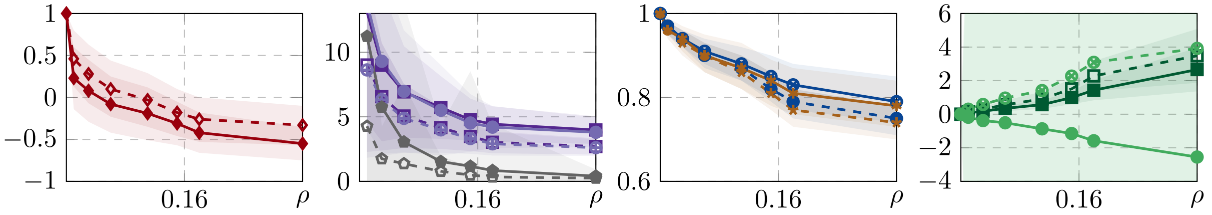
CNN - Integrated Gradients (IG) on Yelp

LSTM - Saliency Maps (S) on Yelp

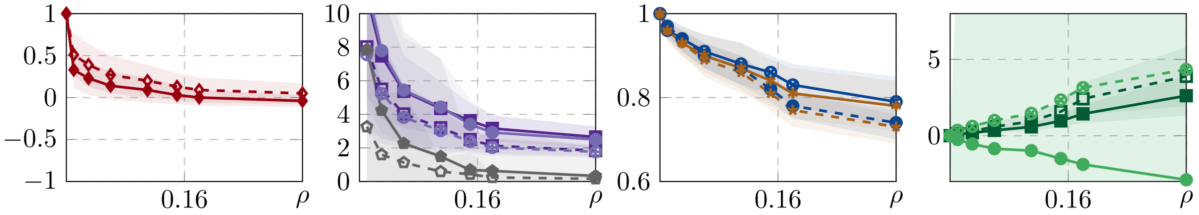
LSTM - Integrated Gradients (IG) on Yelp

LSTMAtt - Saliency Maps (S) on Yelp

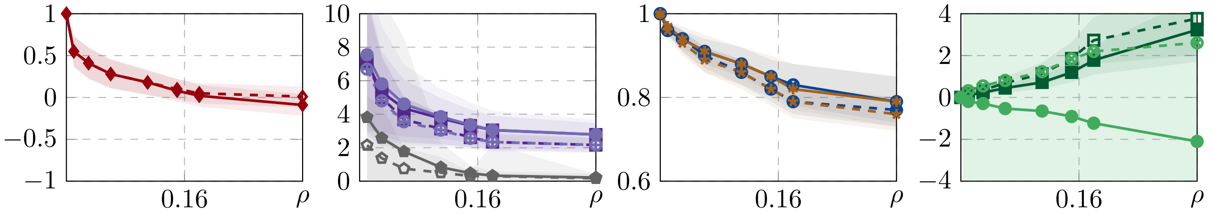
LSTMAtt - Integrated Gradients (IG) on Yelp



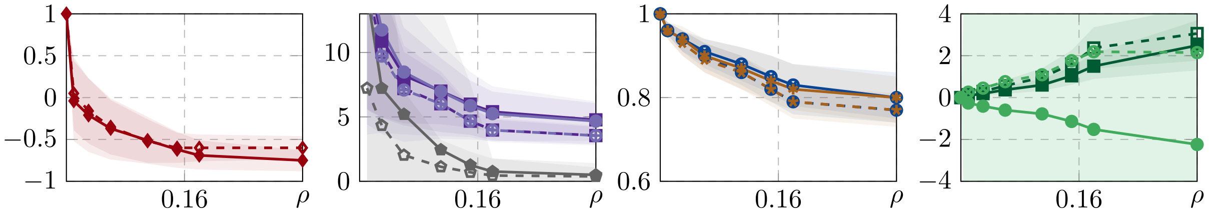
LSTMAtt - Self-Attention (A) on Yelp



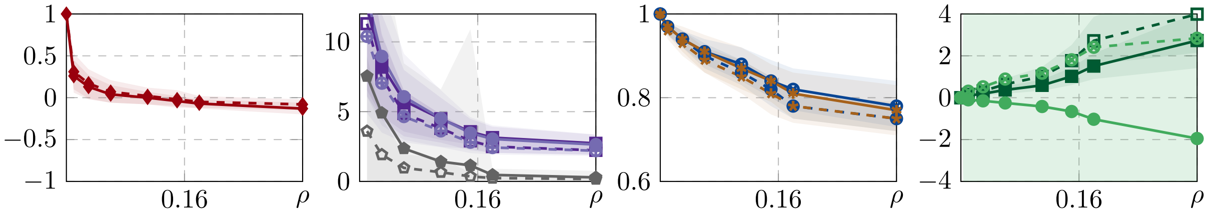
BERT - Saliency Maps (S) on Yelp



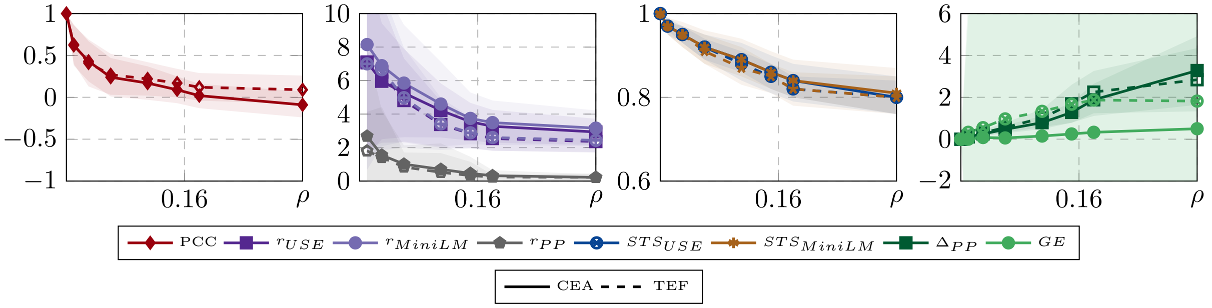
BERT - Integrated Gradients (IG) on Yelp

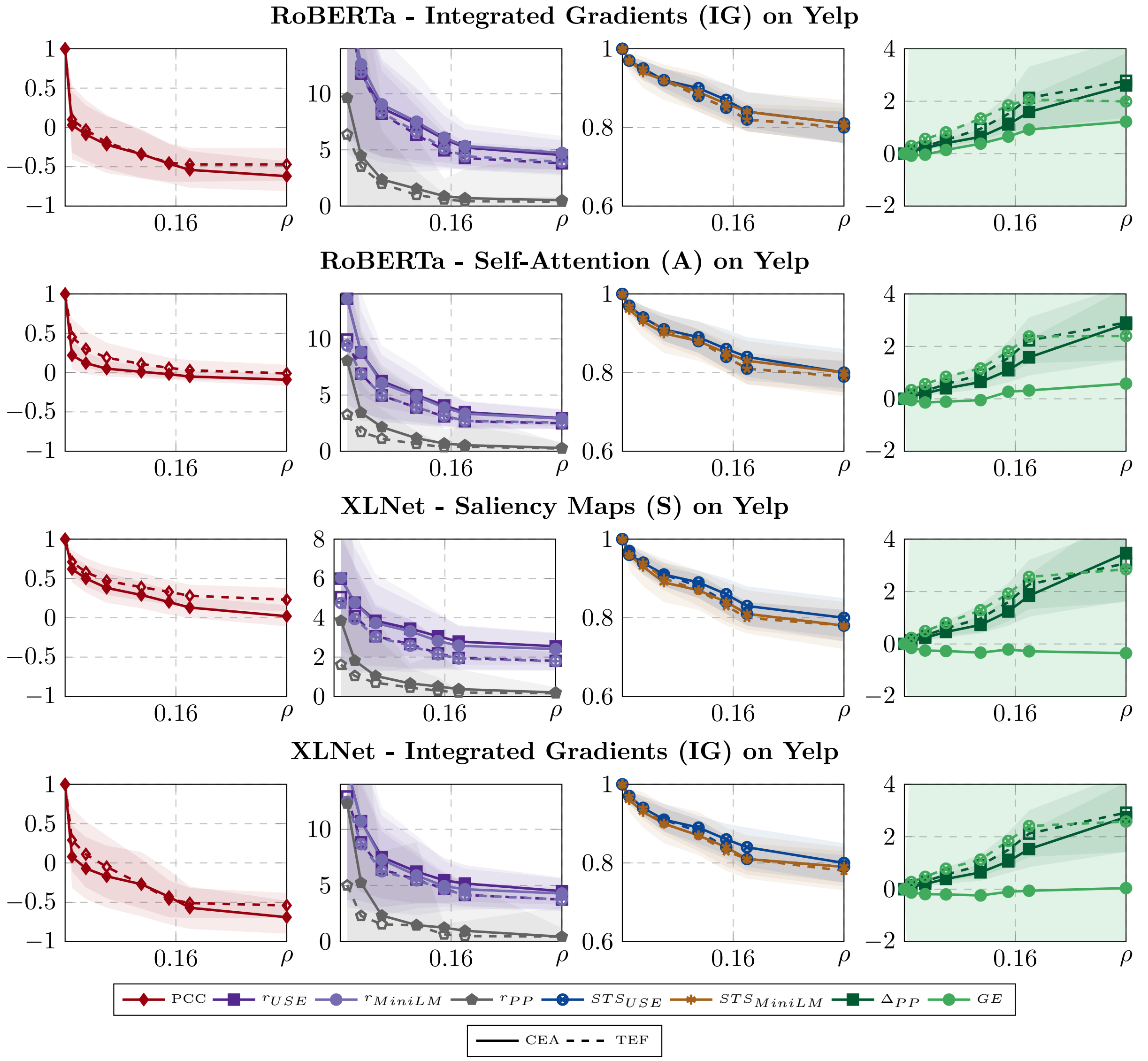


BERT - Self-Attention (A) on Yelp



RoBERTa - Saliency Maps (S) on Yelp



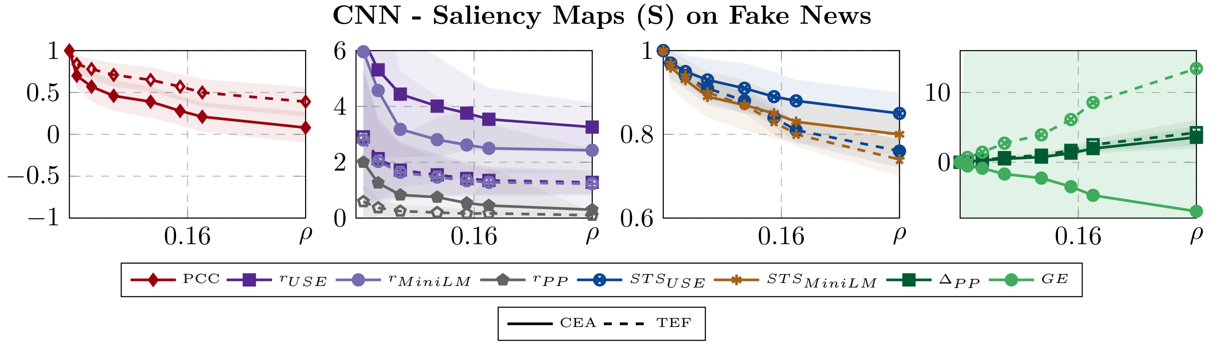

RoBERTa - Integrated Gradients (IG) on Yelp

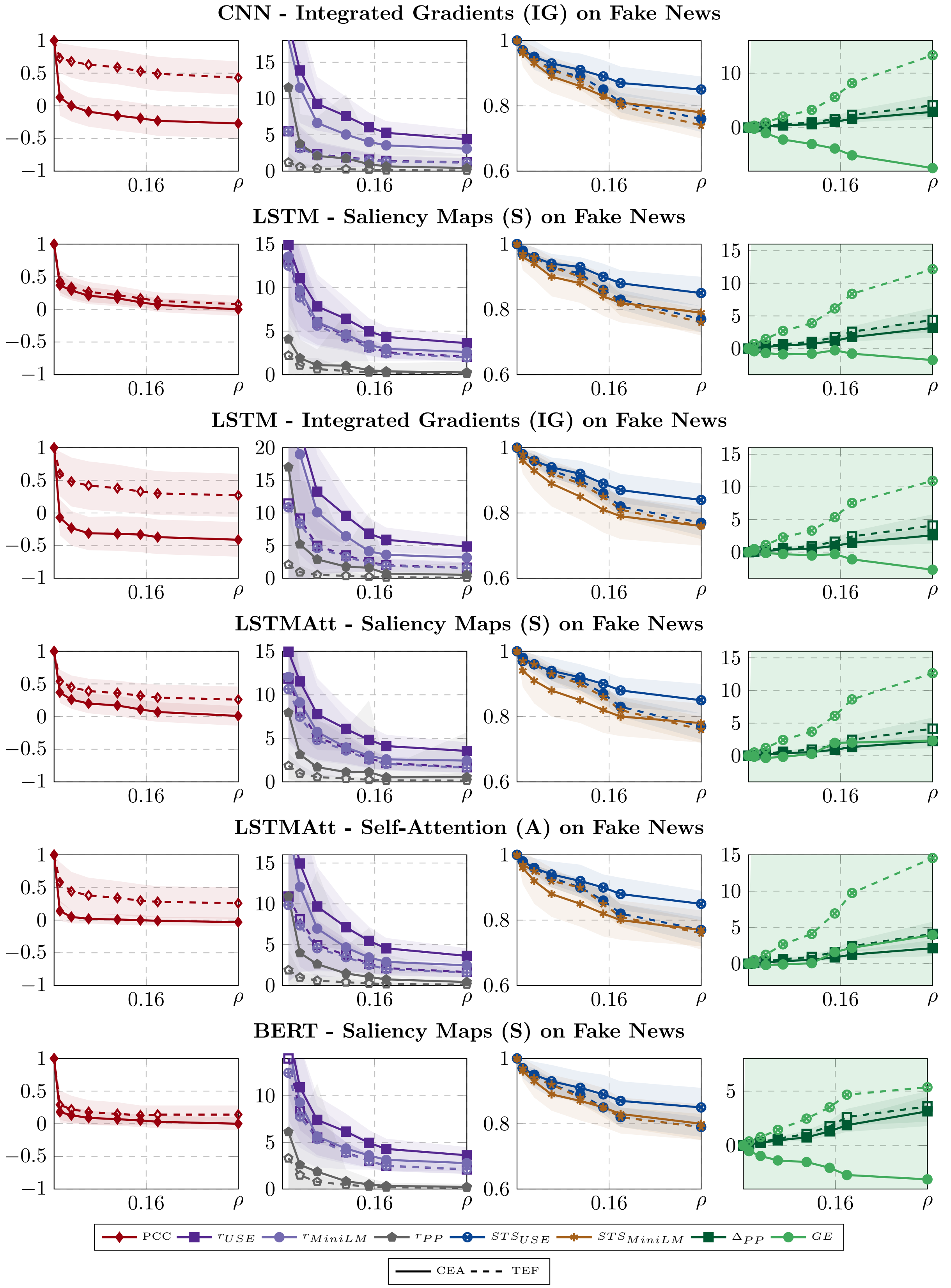
RoBERTa - Self-Attention (A) on Yelp

XLNet - Saliency Maps (S) on Yelp

XLNet - Integrated Gradients (IG) on Yelp

A.3.5 Fake News


CNN - Saliency Maps (S) on Fake News



CNN - Integrated Gradients (IG) on Fake News

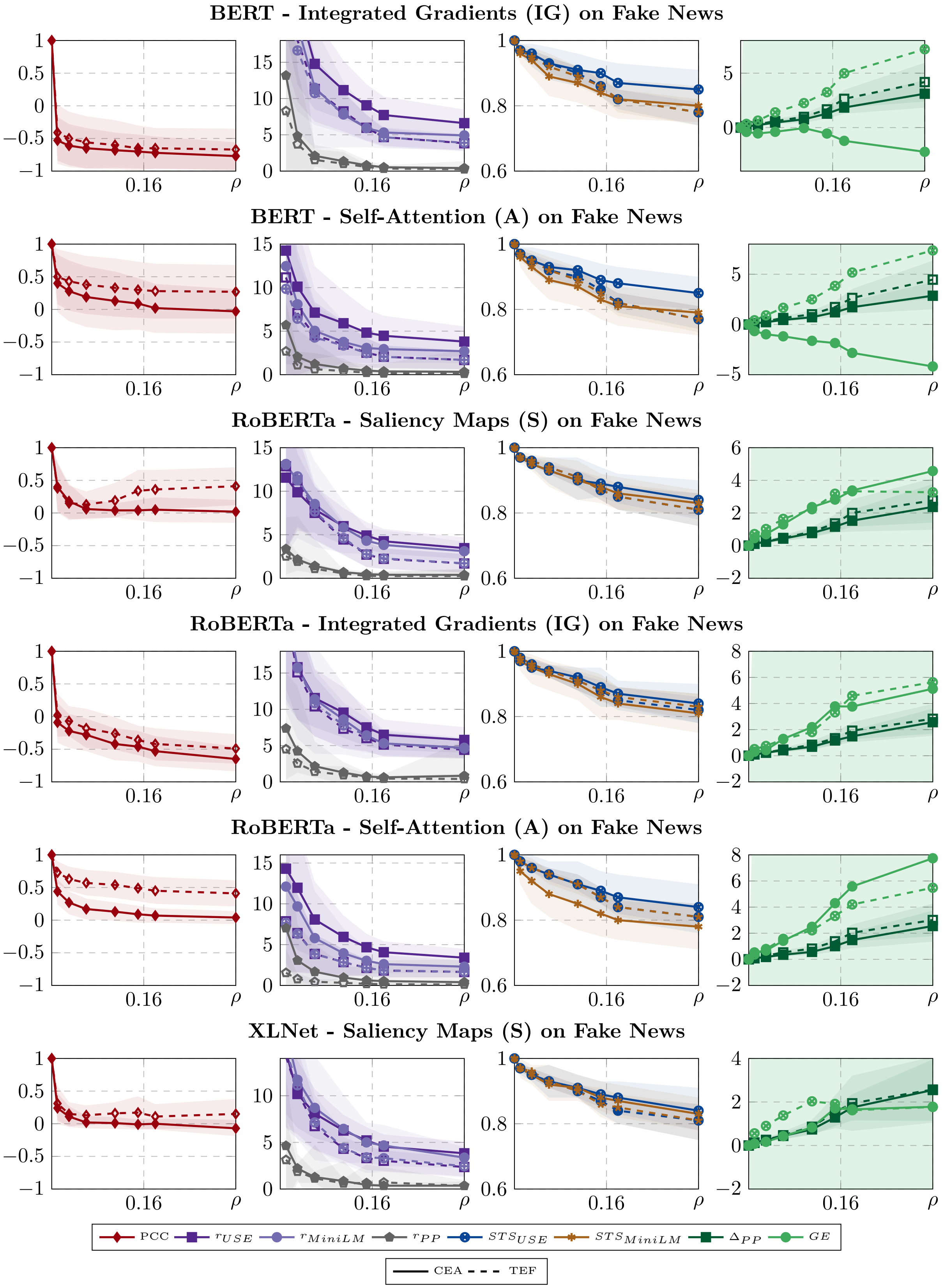
LSTM - Saliency Maps (S) on Fake News

LSTM - Integrated Gradients (IG) on Fake News

LSTMAtt - Saliency Maps (S) on Fake News

LSTMAtt - Self-Attention (A) on Fake News

BERT - Saliency Maps (S) on Fake News



BERT - Integrated Gradients (IG) on Fake News

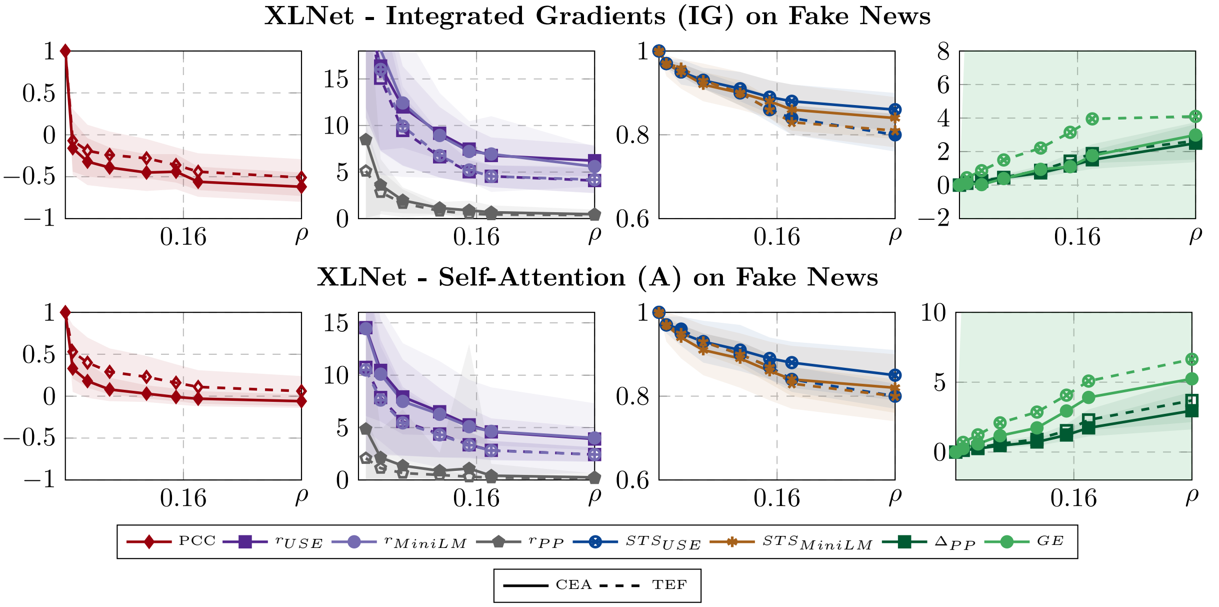
BERT - Self-Attention (A) on Fake News

RoBERTa - Saliency Maps (S) on Fake News

RoBERTa - Integrated Gradients (IG) on Fake News

RoBERTa - Self-Attention (A) on Fake News

XLNet - Saliency Maps (S) on Fake News



XLNet - Integrated Gradients (IG) on Fake News

XLNet - Self-Attention (A) on Fake News