In Situ Augmentation for Defending Against Adversarial Attacks on Text Classifiers

Lei Xu[1], Laure Berti-Equille[2], Alfredo Cuesta-Infante[3], Kalyan Veeramachaneni[1]
[1]MIT LIDS, [2]IRD, [3]Universidad Rey Juan Carlos

Abstract

In text classification, recent research shows that adversarial attack methods can generate sentences that dramatically decrease the classification accuracy of state-of-the-art neural text classifiers. However, very few defense methods have been proposed against these generated high-quality adversarial sentences. In this paper, we propose LMAg (Language-Model-based Augmentation using Gradient Guidance), an *in situ* data augmentation method as a defense mechanism effective in two representative attack setups. Specifically, LMAg uses the norm of the gradient to estimate the importance of a word to the classifier's prediction, then substitutes those words with alternatives proposed by a masked language model. LMAg is an additional protection layer on the classifier, thus does not require additional training. Experimental results show that LMAg can improve after-attack accuracy of BERT text classifier by 51.5% and 17.3% for two setups respectively.

Problem Formulation

Efficacy of Adversarial Attack on Text Classification

- Given a sentence $\mathbf{x} = \{x_1, \ldots, x_l\}$ and its label y, a text classifier $f(\cdot)$ is supposed to make a prediction $\hat{y} = f(\mathbf{x})$ where $\hat{y} = y$ with high probability. When $f(\mathbf{x}) = y$, an adversarial attack method $\mathcal{A}(\mathbf{x}, y, f)$ generates an adversarial sentence $\mathbf{u}$ where $\mathbf{u}$ is grammatically correct and has the same semantic meaning as $\mathbf{x}$, but $f(\mathbf{u}) \neq y$. The efficacy of adversarial attack is measured by after attack accuracy on the test set $\mathcal{D}$ such as:

$$\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}[f(\mathcal{A}(\mathbf{x}, y, f)) = y]. \quad (1)$$

Efficacy of Original Defense Against Adversarial Examples

- In this setup (Setup I), we generate adversarial examples by attacking the original classifier $f(\cdot)$, then we evaluate the robustness of the original classifier based on the absence of mistakes on these examples. In this setup, the after-attack accuracy on the test set $\mathcal{D}$ is defined as:

$$\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}[f'(\mathcal{A}(\mathbf{x}, y, f)) = y]. \quad (2)$$

Efficacy of Boosted Defense Against Adversarial Examples

- In this setup (Setup II), we generate adversarial examples by attacking the robustified classifier $f'(\cdot)$. In this setup, the after-attack accuracy is defined as:

$$\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}[f'(\mathcal{A}(\mathbf{x}, y, f')) = y]. \quad (3)$$

Method

LMAg consists of three steps:

- Estimate the importance of words using the gradient of the classifier.
- Generate multiple rephrases by stochastically masking important words in the input sentence and filling in with alternative words using a masked language model.
- Make a prediction based on the majority of predictions on the rephrases.

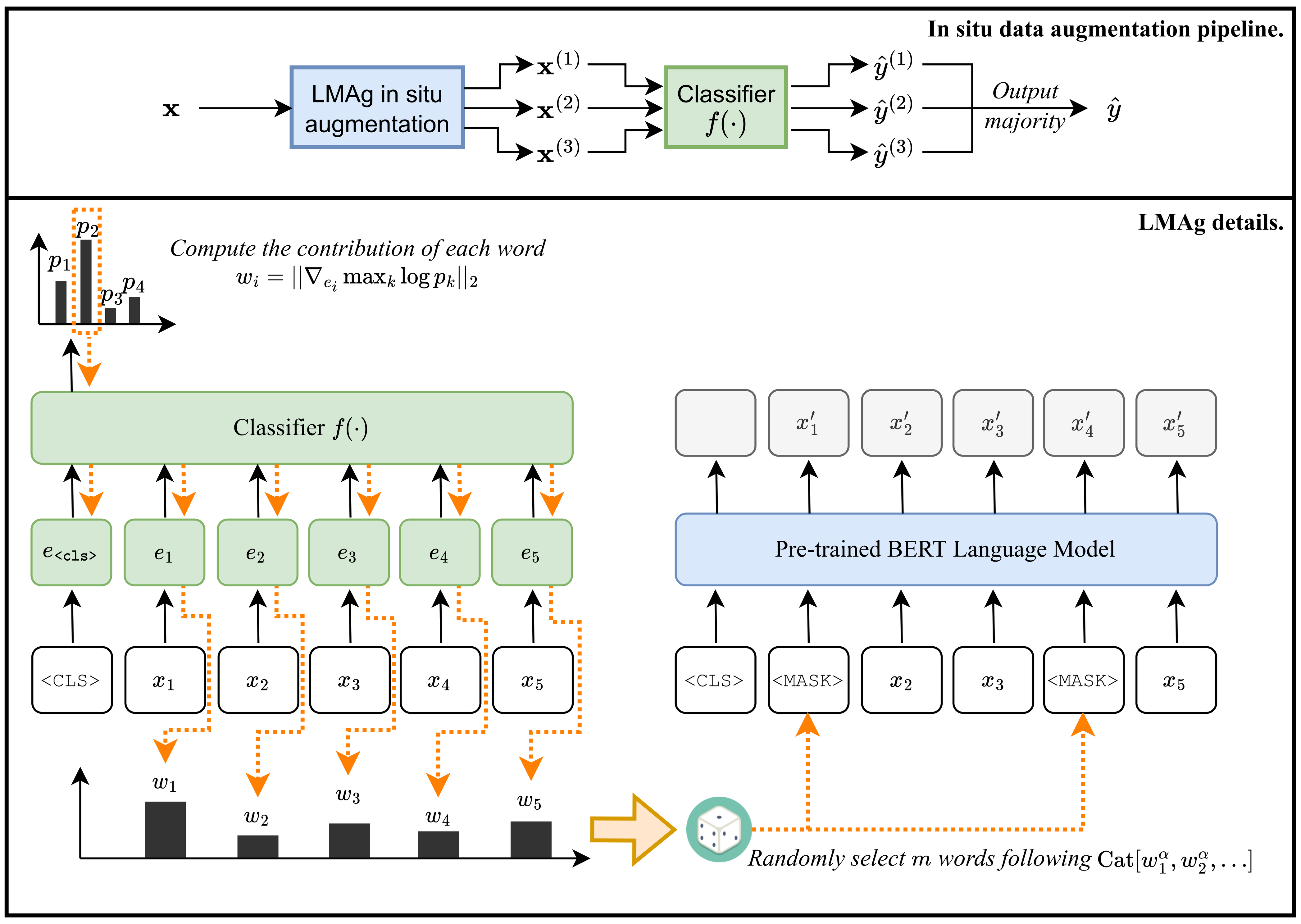

Figure 1: An overview of LMAg.

Experiment Settings

- **Datasets**. We use 5 text classification datasets: (1) **AG**'s News; (2) Movie Reviews (**MR**); (3) **Yelp** Reviews; (4) **IMDB** Movie Reviews; and (5) binary Sentiment Treebank (**SST2**).
- **Original Classifier**. For all datasets, we use the BERT-base classifier (#layers=12, hidden_size=768). We fine-tune the classifier on 20k batches (5k batches on MR and IMDB), with batch size 32. We use the AdamW optimizer and learning rate 0.00002.
- **Attack Methods:** We pick 5 recently proposed adversarial attack methods implemented in TextAttack: (1) **PWWS**, (2) TextFooler (**TF**), (3) BERT-ATTACK (**BA**), (4) **BAE**; and (5) SememePSO (**PSO**).
- **Baseline Defense Methods:** (1) Adversarial training (**AT**); and (2) Synonym encoding (**SEM**).

Experiment results

- In original defense, our LMAg improves the accuracy by 51.5% in average while AT performs slightly better with an improvement of 53.7%.
- In boosted defense, LMAg can improve the after-attack accuracy by 17.3% in average which is significantly better than the other two baselines.
- Effect of three hyperparameters in LMAg is shown on the right.

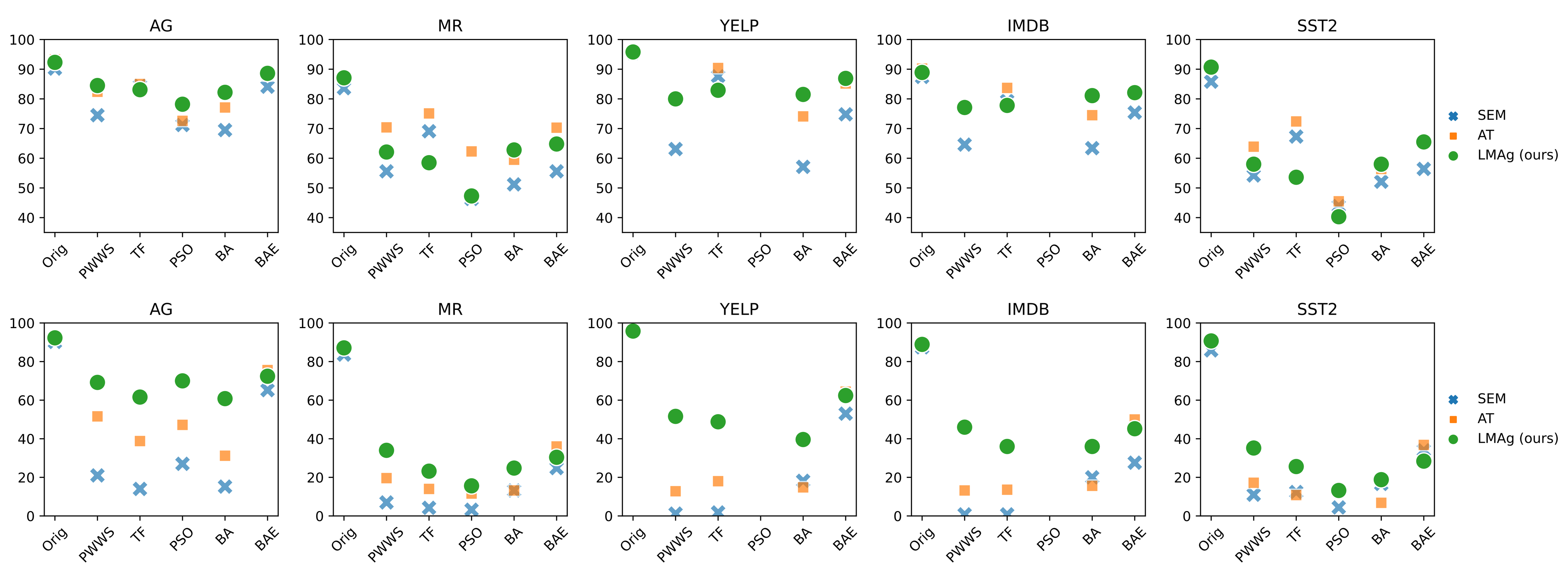

Figure 2: After-attack accuracy of the classifier (%) for each adversarial method (X-axis) on both setups: Setup I (top) – The adversarial examples are generated to attack the original classifier on the original test set; Setup II (bottom) – The adversarial examples are generated to attack the robustified classifier.

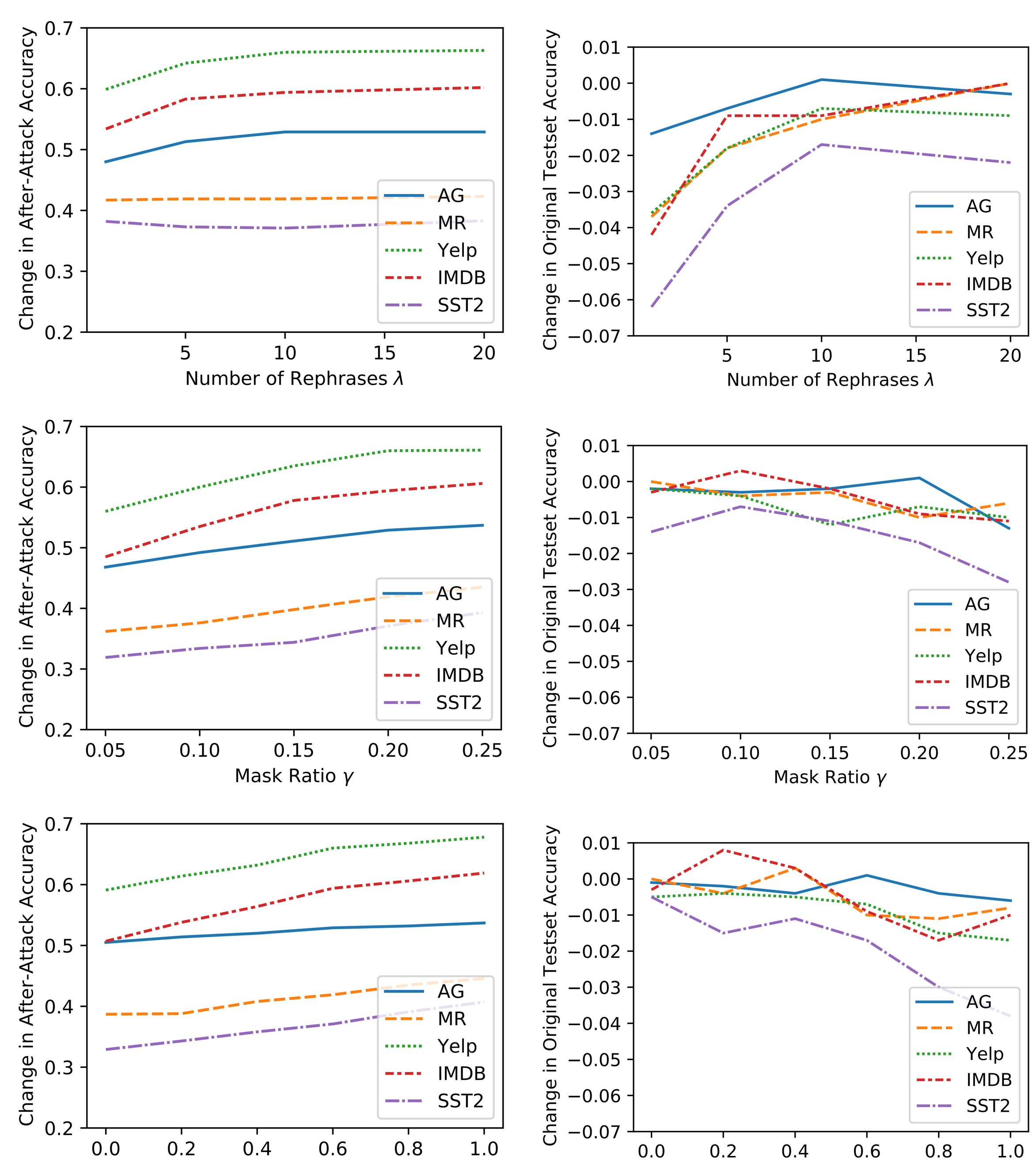

Figure 3: The effect of hyperparameters. The left column shows the change of after-attack accuracy on each data set, the right column shows the change of original test set accuracy.