From Adversarial Arms Race to Model-centric Evaluation Motivating a Unified Automatic Robustness Evaluation Framework

Yangyi Chen[1,2*], **Hongcheng Gao**[1,3*], **Ganqu Cui**[1*], **Lifan Yuan**[1,4]
Dehan Kong[5], **Hanlu Wu**[5], **Ning Shi**[5], **Bo Yuan**[5], **Longtao Huang**[5], **Hui Xue**[5]
Zhiyuan Liu[1,6†], **Maosong Sun**[1,6†], **Heng Ji**[2]
[1]NLP Group, DCST, IAI, BNRIST, Tsinghua University, Beijing
[2]UIUC [3]Chongqing University [4]HUST [5]Alibaba Group
[6] Jiangsu Collaborative Innovation Center for Language Ability, Jiangsu Normal University
yangyic3@illinois.edu, gaohongcheng2000@gmail.com

Abstract

Textual adversarial attacks can discover models' weaknesses by adding semantic-preserved but misleading perturbations to the inputs. The long-lasting adversarial attack-and-defense arms race in Natural Language Processing (NLP) is algorithm-centric, providing valuable techniques for automatic robustness evaluation. However, the existing practice of robustness evaluation may exhibit issues of incomprehensive evaluation, impractical evaluation protocol, and invalid adversarial samples. In this paper, we aim to set up a unified automatic robustness evaluation framework, shifting towards model-centric evaluation to further exploit the advantages of adversarial attacks. To address the above challenges, we first determine robustness evaluation dimensions based on model capabilities and specify the reasonable algorithm to generate adversarial samples for each dimension. Then we establish the evaluation protocol, including evaluation settings and metrics, under realistic demands. Finally, we use the perturbation degree of adversarial samples to control the sample validity. We implement a toolkit **RobTest** that realizes our automatic robustness evaluation framework. In our experiments, we conduct a robustness evaluation of RoBERTa models to demonstrate the effectiveness of our evaluation framework, and further show the rationality of each component in the framework. The code will be made public at https://github.com/thunlp/RobTest.

1 Introduction

Pre-trained language models (PLMs) are vulnerable to textual adversarial attacks that fool the models by adding semantic-preserved perturbations to the inputs (Zhang et al., 2020). Compared to the static evaluation benchmarks (Wang et al., 2018, 2019a), attack methods can continually generate diverse adversarial samples to reveal models' weaknesses, rendering a more comprehensive and rigorous model evaluation. Previous work explores adversarial NLP in both the attack (Gao et al., 2018a; Alzantot et al., 2018) and the defense (Mozes et al., 2021; Huang et al., 2019) sides, leading to a long-lasting adversarial arms race.

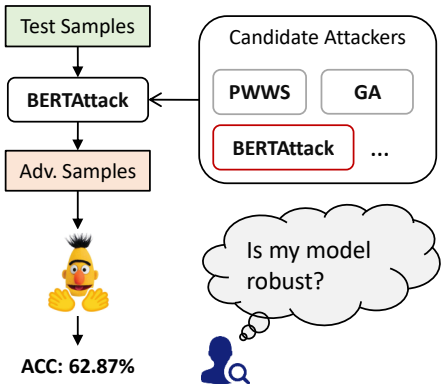


Figure 1: The original evaluation pipeline. The attacker is usually selected by intuition and practitioners get little information from scores.

The arms race is algorithm-centric. It continually motivates stronger attack and defense methods to explore and fix models' weaknesses, providing useful techniques for robustness evaluation. However, existing work on model robustness evaluation naturally follows the previous evaluation practice, and doesn't fully consider the real-world needs of robustness evaluation (Zeng et al., 2021; Wang et al., 2021b; Goel et al., 2021) (See Figure 1). We identify three weaknesses in previous robustness evaluation: (1) Relying on a single attack method (Zang et al., 2020) or static challenging datasets (Nie et al., 2019; Wang et al., 2021a), which can only measure a limited number of aspects of models' capabilities; (2) Directly inheriting the evaluation settings and metrics in the arms race era, which may result in impractical evaluation (Zeng et al., 2021; Morris et al., 2020b); (3) Designing

*Indicates equal contribution. Work done during internship at Tsinghua University.
†Corresponding Author.

Findings of the Association for Computational Linguistics: ACL 2023, pages 9607–9632
July 9-14, 2023

invalid adversarial sample[1] filtering rules based on certain thresholds (e.g., sentence similarity), which cannot generalize to all kinds of adversarial samples (Wang et al., 2021b; Zeng et al., 2021).

Thus, we propose to shift towards the model-centric evaluation, which should satisfy the following characteristics accordingly: (1) **Comprehensively** measuring NLP models' robustness; (2) Establishing a **reasonable** evaluation protocol considering practical scenarios; (3) Filtering out invalid adversarial samples for **reliable** robustness estimation. Given these challenges, a standard and acknowledged framework for employing adversarial attacks to automatically measure and compare NLP models' robustness is lacking (See Figure 7).

In this paper, we motivate a unified model-centric automatic robustness evaluation framework based on the foundation of the adversarial arms race. To achieve **comprehensive** evaluation, we define eight robustness dimensions from top to down, constituting a evaluation of multi-dimensional robustness towards sentence-level, word-level, and char-level transformations. For each robustness dimension, we specify the concrete algorithm to generate adversarial samples. Then we set up a **reasonable** evaluation protocol by specifying evaluation settings and metrics under realistic demands. Finally, we rely on the perturbation degree to control the validity of generated adversarial samples for more **reliable** robustness evaluation. Our intuition is that adversarial samples with smaller perturbation degrees are more likely to be valid, which is justified through human annotation experiments.

We implement a toolkit **RobTest** to realize our robustness evaluation framework (See Figure 6). We highlight four core features in RobTest, including basic adversarial attack methods, robustness report generation, general user instructions, and adversarial data augmentation. In experiments, we use RobTest to measure the robustness of RoBERTa models (Liu et al., 2019) to demonstrate the effectiveness of our evaluation framework in addressing the core challenges. Further, we show the rationality of each component in our robustness evaluation framework through detailed analysis.

2 Model-centric Robustness Evaluation

In this section, we motivate the first model-centric automatic robustness evaluation framework. We first define robustness evaluation dimensions and specify corresponding attack algorithms (Sec. 2.1). Then we discuss the evaluation protocol under realistic demands (Sec. 2.2). Finally, we provide solutions to filter out invalid adversarial samples for more reliable robustness evaluation (Sec. 2.3).

2.1 Robustness Evaluation Dimension

Motivation. Existing research designs adversarial attacks based on observations (Le et al., 2022) or intuitions (Li et al., 2020) and adopts the proposed method to test the robustness of evaluated models. In this procedure, the robustness evaluation is restricted to the specific attack method without considering samples from other potential distributions. We argue that considering only one single dimension cannot comprehensively describe the models' robustness (See Sec. 4.3 for verification).

Selection Criteria. We build our model-centric robustness evaluation framework based on the foundation of adversarial NLP but aim to cover a more comprehensive set of robustness dimensions. We integrate previous adversarial attack methods in a systematic way. We focus on task-agnostic robustness dimensions[2], and define them from top to down (See Table 1). The selection criteria of robustness evaluation dimensions and attack methods are: (1) **Important and practical**: Methods that can reasonably simulate common inputs from real-world users or attackers; (2) **Representative**: Methods that have been studied for a long time in the adversarial arms race stage and have many homogeneous counterparts; (3) **Diversified**: Methods that explore various aspects of model capabilities.

Note that we don't consider the "imperceptible perturbations" requirement in the selection of robustness dimensions, although previous work repeatably emphasizes this requirement (Goodfellow et al., 2014; Ren et al., 2019; Zang et al., 2020). We give our justification in Appendix B.

Dimensions. We start from a high-level categorization, considering char-level, word-level, and sentence-level transformations, differing in the perturbation granularity (See Table 1). **Char-level** transformations add perturbations to characters in the word units. We include the following dimensions in our framework: (1) **Typo** (Li et al., 2018; Eger and Benz, 2020) considers five basic operations to add typos in the inputs, including randomly

[1]Detailed explanation for validity is in Appendix A.

[2]Task-specific robustness dimensions are also essential, and we leave it for future work.

Granularity	Dimension	General?	Malicious?	Case
Char-level	Typo	Yes	Yes	I watch a smart, swet adn playful romantic comedy.
	Glyph	Yes	Yes	I watch a ŝmârt, sweet and playful romaňtîĉ comedy.
	Phonetic	Yes	Yes	I wotch a smart, sweet and playful romentic comedy.
Word-level	Synonym	Yes	No	I watch a smart, sweet and naughty romantic comedy.
	Contextual	Yes	No	We watch a smart, sweet and playful romantic teleplay.
	Inflection	Yes	No	I watched a smart, sweet and playful romantic comedies.
Sentence-level	Syntax	Yes	No	In my eyes will be a witty, sweet romantic comedy.
	Distraction	No	Yes	I watch a smart, sweet and playful romantic comedy. True is not False.

Table 1: The robustness dimensions included in our framework. We also attach general and malicious robustness tags to each dimension. The original sentence is "I watch a smart, sweet and playful romantic comedy."

delete, insert, replace, swap, or repeat one character; (2) **Glyph** (Li et al., 2018; Eger et al., 2019) replaces characters with visually-similar ones; (3) **Phonetic** (Le et al., 2022) replaces characters but makes the whole word sound similar to the origin. **Word-level** transformations modify word units as a whole. We include the following dimensions in our framework: (1) **Synonym** (Ren et al., 2019; Zang et al., 2020) replaces words with their synonymous substitutes according to external knowledge sources. We consider WordNet (Miller, 1995) and HowNet (Dong and Dong, 2003) in our implementation; (2) **Contextual** (Li et al., 2020; Garg and Ramakrishnan, 2020) replaces words with their context-similar substitutes, which are generated by masked language models; (3) **Inflection** (Tan et al., 2020) perturbs the inflectional morphology of words. **Sentence-level** transformations generate adversarial samples directly from the entire original sentences. We include the following dimensions in our framework: (1) **Syntax** (Iyyer et al., 2018; Huang and Chang, 2021; Sun et al., 2021) transforms the syntactic patterns of original samples; (2) **Distraction** (Naik et al., 2018; Ribeiro et al., 2020; Chen et al., 2022a) appends some irrelevant contents to the end of sentences.

Malicious & General Tags. For each robustness dimension, we also attach the general or malicious tag to characterize the intended simulated agents. The general (malicious) tag indicates that the generated samples mainly come from benign users (malicious attackers). For example, Synonym and Distraction are representative types of general and malicious dimensions respectively. Note that we attach both tags to three char-level transformations since both benign users and malicious attackers can produce these kinds of samples.

2.2 Evaluation Protocol

Motivation. Previous work in adversarial NLP naturally follows the early attempts (Szegedy et al., 2013; Goodfellow et al., 2014; Liang et al., 2017; Gao et al., 2018a) to establish the evaluation protocol. However, Chen et al. (2022b) categorize and summarize four different roles of textual adversarial samples, and argue for a different evaluating protocol for each role. In our framework, we reconsider the robustness evaluation protocol when employing adversarial attack methods for model evaluation. We first describe the evaluation setting, and then the evaluation metrics in our framework.

Evaluation Setting (available information from evaluated models). Most existing attack methods assume the accessibility to confidence scores only (Alzantot et al., 2018; Ren et al., 2019; Zang et al., 2020; Li et al., 2020; Chen et al., 2021). We acknowledge the rationality of this assumption since the size of models may become too large nowadays (Radford et al., 2019; Brown et al., 2020), resulting in inefficient evaluation if also requiring the gradients information for adversarial samples generation (Goodfellow et al., 2014). However, in practice, we as practitioners mostly have all access to the evaluated models, including the parameters and gradient information, for better robustness evaluation.

Thus, we implement three evaluation settings in our framework, assuming different available information from evaluated models. The settings include rule-based, score-based, and gradient-based attacks. Rule-based attacks don't assume any information from the evaluated models and generate adversarial samples based on pre-defined rules. Score-based and gradient-based attacks assume access to the confidence scores and gradients information respectively from evaluated models for more rigorous evaluation. They first compute the saliency maps that give the importance scores to each word for samples and then perform selective perturbations based on the scores. Specifically, for score-based attacks, we employ the difference in confidence scores when iteratively masking each word as the

Original	I love the way that it took chances and really asks you to take these great leaps of faith and pays off.
BERT-Attack (Li et al., 2020)	I hate the way that it took chances and jesus asking you to take these grand leaps of faith and pays off.
GA (Alzantot et al., 2018)	I screw the way that it read chances and really asks you to remove these great leaps of faith and pays off.
Textbugger (Li et al., 2018)	I lve the way that it took cances and really a sks you to take these grwat lezps of fith and pay5 off.

Table 2: Cases of invalid adversarial samples crafted by three popular attack methods. The original label is positive.

important score for that word. For gradient-based attacks, we employ integrated gradient (IG) (Sundararajan et al., 2017) to compute the saliency map. IG computes the average gradient along the linear path of varying the input from a baseline value to itself. Besides, we use greedy search since it can achieve satisfying performance within a reasonable time (Yoo et al., 2020).

Evaluation Metrics. Most previous work considers the "is robust" problem (Li et al., 2020, 2021; Chen et al., 2021). They generate adversarial samples for each original sample and test if at least one of them can successfully attack the evaluated models. Then the final score is computed as the percentage of samples that are not attacked successfully. This is the **worst performance estimation**, requiring models to be robust to all potential adversarial samples in order to score. In our framework, we introduce the **average performance estimation** for a more comprehensive robustness evaluation. Specifically, for each original sample, we compute the percentage of cases that models can correctly classify among all potential adversarial samples. Then we average over all original samples to get the average performance estimation score.

2.3 Reliable Robustness Evaluation

Motivation. Previous work chases for higher attack success rate, while the validity of adversarial samples may be sacrificed[3]. The consequence of this practice is unreliable and inaccurate robustness evaluation. We showcase adversarial samples crafted by three popular methods on SST-2 (Socher et al., 2013) in Table 2. While all samples successfully flip the predictive label, they are not good choices for robustness evaluation because the ground truth label is changed (e.g., BERT-Attack) or the meaning of the original sentence is changed (e.g., GA, Textbugger). Morris et al. (2020a); Wang et al. (2021a); Hauser et al. (2021) show that there are many such invalid cases in adversarial samples that successfully mislead models' predictions. We further conduct a human evaluation to support this conclusion. We hire annotators to evaluate adversarial samples validity of three representative attack methods, namely contextual-based (Li et al., 2020), synonym-based (Zang et al., 2020), and typo-based attacks (Karpukhin et al., 2019). The results show that on average only **25.5%**, **20.0%**, and **31.5%** generated samples are valid. Thus, if directly employing original adversarial samples for robustness evaluation, the results are unreliable and don't convey too much useful information to practitioners.

Potential Solutions. For reliable robustness evaluation, we need to consider how to ensure the validity of constructed adversarial samples. We can approach this problem in two different ways: (1) Verify generated adversarial samples; (2) Incorporating the validity criterion in robustness evaluation. All existing work focuses on verification. For example, in the implementation of OpenAttack (Zeng et al., 2021) and TextFlint (Wang et al., 2021b), an embedding similarity threshold is set for filtering adversarial samples. However, we argue that **a unified sample selection standard without considering the specific trait of the attack method can not perform effective filtering.** For example, consider the adversarial sample crafted by adding typos: "I love the way that it took chancs and really asks you to takke these great leaps of faith and pays off." This sample may be filtered out by the similarity or perplexity threshold due to its unnatural expression. However, it well simulates the input from real-world users and retains the original meaning, thus should be considered in the evaluation.

Our Method. In our framework, we consider incorporating the validity criterion into robustness evaluation. We hold a basic intuition that there is an inverse correlation between the perturbation degree and the validity of adversarial samples. Thus, we rely on the perturbation degree to measure the adversarial sample validity. Note that the perturbation degree is defined according to the concrete transformation level[4]. We justify our intuition and demonstrate the superiority of this filtering strategy compared to previous heuristic rules (e.g., grammar

[3]We give a detailed explanation for adversarial samples validity in Appendix A.

[4]The computational details are described in Appendix C.

error, sentence similarity, perplexity) in Sec. 4.3.

We propose to measure models' robustness under the specific attack method in various perturbation degrees and compute a robustness score for each degree. The robustness score is the model's worst performance estimation or average performance estimation. We put more emphasis on the robustness scores computed at lower perturbation degrees[5] and employ the exponentially weighted moving average (Hunter, 1986) to compute the final score for each robustness dimension. Formally, we use $\theta_1, \theta_2, ..., \theta_n$ to denote robustness scores computed at n perturbation degrees from high to low. Set $\mathcal{V}_1 = \theta_1$. To compute the **final robustness score** $\mathcal{V}_n$:

$$\mathcal{V}_t = \beta * \mathcal{V}_{t-1} + (1-\beta) * \theta_t, \quad t = 2, ..., n, \quad (1)$$

where β controls the weights on scores computed at different degrees. Empirically, it should be chosen depending on the risk level of the considered task, and smaller β will more emphasize the importance of evaluation on high-perturbed samples, which is essential for high-stake applications. In our framework, we set β=0.5 for demonstration.

3 RobTest

We implement an automatic robustness evaluation toolkit named **RobTest** to realize our proposed framework. We highlight four features of RobTest.

Basic Adversarial Attack Methods. We implement eight attack methods, corresponding to eight robustness evaluation dimensions in our framework. We also include three attack types that assume different information available from evaluated models, namely rule-based, score-based, and gradient-based attacks. RobTest allows practitioners to customize evaluated models and datasets and design new attack methods to test specified robustness dimensions. Also, it supports the multi-process running of adversarial attacks for efficiency.

Robustness Report. RobTest provides comprehensive robustness reports for evaluated models. See Figure 2 and Appendix G for examples of single-model robustness reports. See Figure 3 and Appendix H for examples of the robustness comparison of the two models. We further discuss the details of robustness reports in Sec. 4.

[5]Note that the perturbation degree computation methods are different for different dimensions (See Appendix C).

General Instructions. Existing toolkits that implement various attack methods don't provide detailed guidance on how to conduct robustness evaluation (Morris et al., 2020b; Zeng et al., 2021; Wang et al., 2021b). In RobTest, we provide general instructions for practitioners. Two kinds of instructions are included: (1) How to select appropriate robustness dimensions to evaluate, and which accessibility (e.g., score-based) should be considered. We introduce detailed descriptions of all robustness dimensions in RobTest, including the real-world distributions they consider; (2) How to understand the robustness report. We give detailed explanations for the figures and tables in the report.

Data Augmentation. Practitioners may identify several weak robustness dimensions of evaluated models. RobTest supports generating adversarial samples under the specified perturbation degree for data augmentation to improve the robustness.

4 Experiment

We conduct experiments to demonstrate the effectiveness of our automatic robustness evaluation framework using RobTest. We aim to show how our framework fulfills the characteristics of model-centric robustness evaluation[6].

4.1 Experimental Setting

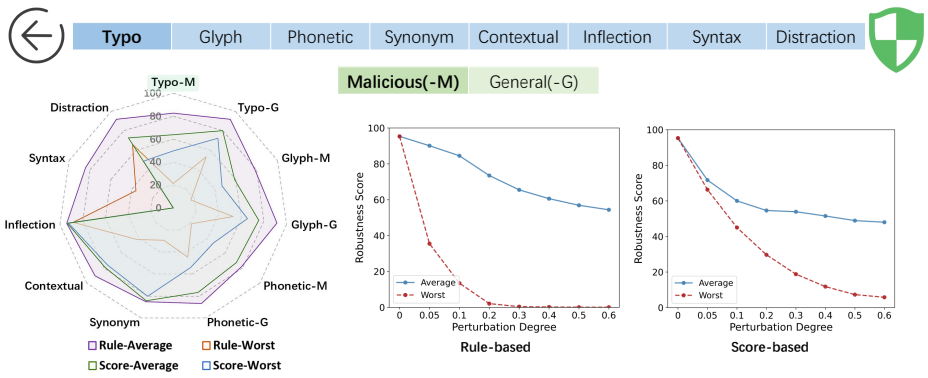
Dataset and Evaluated Model. In our experiments, we choose the general, common, and application-driven tasks that our task-agnostic robustness dimensions can be applied to[7]. We consider sentiment analysis, news classification, and hate-speech detection tasks. We choose SST-2 (Socher et al., 2013), AG's News (Zhang et al., 2015), and Jigsaw[8] as evaluation datasets. We choose RoBERTa-base and RoBERTa-large (Liu et al., 2019) as evaluated models.

Evaluation Setting. For each dataset, we sample 1,000 samples from the test set for experiments and generate at least 100 testing cases for each sample under each perturbation degree. In pilot experiments, we found no advantage of employing gradient information to generate saliency maps, and

[6]We leave the detailed evaluation and analysis of various model architectures and robustness-enhanced algorithms for future work.

[7]Task-specific robustness dimensions can be designed for certain tasks, e.g., name entity robustness for reading comprehension (Yan et al., 2021). We leave it for future work.

[8]https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.3	90.1	84.5	73.5	65.5	60.6	56.9	54.4	*82.7*
Rule-Worst	95.3	35.5	13.5	2.1	0.4	0.2	0.1	0.1	*21.4*
Score-Average	95.3	71.7	60.0	54.6	53.9	51.5	48.9	48.0	*63.8*
Score-Worst	95.3	66.3	45.0	29.7	18.8	11.8	7.3	5.8	*49.8*

Figure 2: Example of one single page of the robustness report of RoBERTa-base on SST-2, regarding the Typo (Malicious) dimension. The full report is shown in Figure 10. We use Rule- and Score- to denote two evaluation settings, and use -Average and -Worst to denote two metrics.

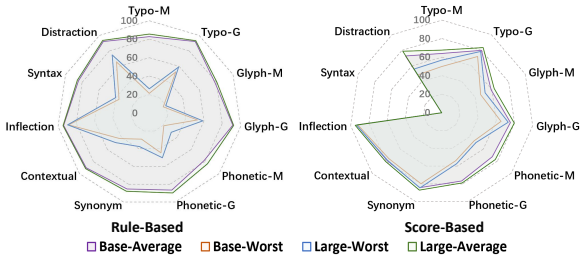


Figure 3: Radar map to compare the robustness of RoBERTa-base and -large considering all dimensions on SST-2. We use Base- and Large- to denote two models, and other denotations are the same as Figure 2.

thus we only consider rule-based and score-based accessibility in experiments. Further research is needed for more effective utilization of gradients.

4.2 Robustness Evaluation

We consider two kinds of robustness evaluation: (1) Robustness evaluation of a given model; (2) Robustness comparison of two models. This can be easily extended to three or more models included.

Single-model Robustness Evaluation. We generate robustness evaluation reports for given evaluated models. Figure 2 shows an example of one single page of the robustness report of RoBERTa-base on SST-2, considering the Typo (Malicious) dimension. Full reports for all datasets and models are in Appendix G. For each dimension, we show the robustness score computed at each robustness level considering two evaluation settings and two metrics, in both figures and the table. We can observe that on average, the model can tolerate inputs with very small perturbation degrees (e.g., 0.05), but its performance degrades significantly in the worst performance estimation. This indicates that the model will be misled if malicious attackers try a little longer, even in small perturbation degrees. The final robustness scores for this dimension are derived by averaging over all robustness scores using Eq. 1, which will serve as overall estimations of the model's robustness in this dimension considering the validity criterion. Also, we adopt the radar map to record the final robustness scores for all robustness dimensions, from which we can easily observe which dimension models fail. For example, we can observe from the radar map in Figure 2 that RoBERTa-base fails frequently when users use various syntactic structures in their expressions or char-level transformations have been adopted for malicious attacks. The implications are: (1) Practitioners should improve the model's capacity to capture syntax patterns or have extra mechanisms to deal with inputs with complex syntactic structures; (2) Practitioners should avoid deploying the model on security-related applications (e.g., hate-speech detection) to prevent hidden dangers.

Robustness Comparison. We can also generate reports to compare the two models' robustness. Figure 3 shows the core part of the report that compares the robustness of RoBERTa-base and RoBERTa-large considering all dimensions on SST-2. We also employ radar maps to clearly show the robustness gap between the two models. The full report is in Appendix H for demonstra-

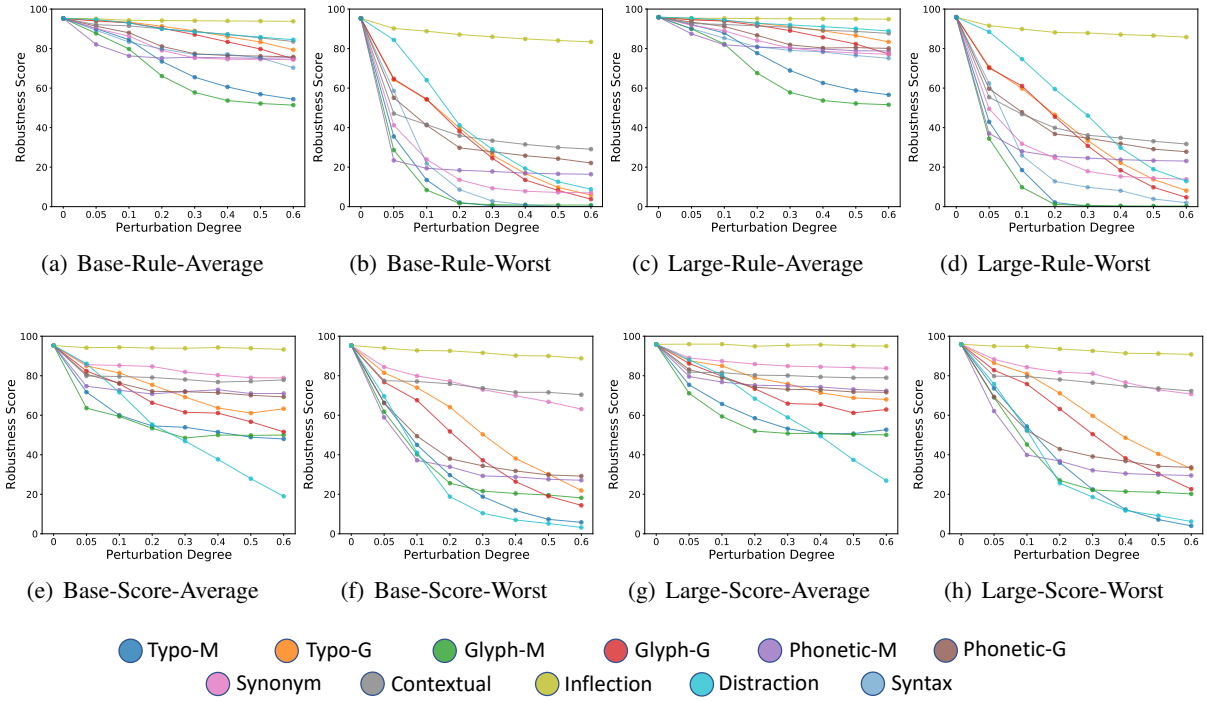


Figure 4: Comprehensive results of RoBERTa-base (Base) and RoBERTa-large (Large) on SST-2. We consider rule-based (Rule) and score-based (Score) attacks, and worst (Worst) and average (Average) performance estimation.

tion. We observe that RoBERTa-large consistently shows better robustness in all dimensions compared to RoBERTa-base. This can be attributed to two potential factors: a) Larger models can generalize better beyond simple patterns (e.g., spurious correlations) in the in-distribution training dataset, thus more robust to distribution shifts (Tu et al., 2020); b) Given the strong correlation between in-distribution and out-of-distribution performance (Miller et al., 2021), the robustness of larger models can be partially explained by better performance on in-distribution data. The quantification of these two factors is left for future work since the experiments in this paper are mainly for demonstration purposes.

4.3 Analysis of Framework Components

In this section, we analyze and prove the rationality of each component in our framework, including eight robustness dimensions, evaluation protocol, and our method to tackle the validity of adversarial samples. For better demonstrations, we aggregate the results of eight dimensions considering two model sizes, two evaluation settings, and two metrics. The results on SST-2 are in Figure 4. The results on AG's News and Jigsaw are in Appendix E.

Robustness Dimensions. We observe that models exhibit different capacities across all robustness dimensions, evidenced by substantially different robustness scores. This indicates the insufficiency in previous practice that adopts one single attack method to evaluate models' robustness. For example, only showing models' robustness to morphology inflection doesn't guarantee the same robustness transfer to inputs containing typos. Thus, a multi-dimensional robustness evaluation in our framework is needed to reveal models' vulnerability in various circumstances, ensuring a more comprehensive evaluation of model capacities.

Evaluation Protocol. Our evaluation protocol includes two evaluation metrics (average and worst performance estimation) and two evaluation settings (rule-based and score-based). We show that the average performance estimation is in complementary to the worst performance estimation, showing the models' average success rates on the corresponding robustness dimension. Thus, it can better reflect models' capacities since most attack methods can reduce models' worst performance estimation to near zero in high perturbation degrees, making it hard to compare different models.

Also, score-based and rule-based attacks consider different evaluation settings. The score-based attacks are more effective than rule-based attacks considering average performance estimation. But the opposite is true considering worst performance

estimation, probably because score-based attacks only perturb certain important words, limiting the search space. Thus, incorporating these two evaluation settings is essential in robustness evaluation.

Invalid Adversarial Samples Filtering. We observe that robustness scores drop along with the increase in the perturbation degrees across different models, datasets, and attack methods. However, as we argue, the robustness scores in higher perturbation degrees underestimate models' robustness since many successful but invalid adversarial samples exist. Thus, directly looking into the robustness curves without considering the influence of perturbation degrees on validity is unreliable.

We justify our solution of incorporating the validity criterion into the robustness estimation process. The basic intuition is that adversarial samples with higher perturbation degrees are more likely to become invalid. We conduct human annotation to verify it (See Table 3). The annotation details are in Appendix D. We can observe that (1) attack methods have a large impact on sample validity, and (2) our intuition is justifiable since mostly a larger perturbation degree substantially harms the validity.

Also, we compare with previous heuristic filtering rules based on grammar errors (Grammar) (Zang et al., 2020; Chen et al., 2021), sentence similarity (USE) (Li et al., 2020; Morris et al., 2020a; Wang et al., 2021b; Zeng et al., 2021), and perplexity (Perplexity) (Qi et al., 2021). We compute predictive validity scores for each adversarial sample based on the filtering rules (e.g., the perplexity rule will assign low validity scores to samples with high perplexity). For each filtering rule, we divide generated adversarial samples into five validity levels based on their validity scores and compute the average human annotated validity score of samples in five levels respectively (See Figure 5). Our method based on the perturbation degree better aligns with the ideal trend, while previous filtering methods show inconsistent trends and cannot effectively distinguish invalid cases.

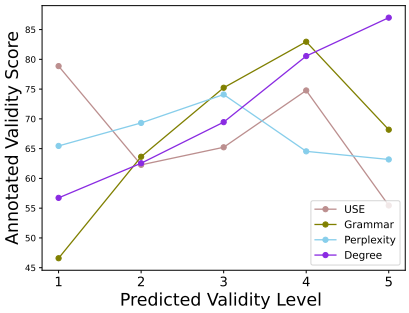


Figure 5: Results of the validity prediction. An ideal prediction should ensure the annotation validity score is proportional to the predicted validity level.

5 Related Work

Standard evaluation benchmarks (Wang et al., 2018, 2019a) follow the Independently Identical Distribution hypothesis that assumes the training and testing data come from the same distribution. However, there is no such guarantee in practice, motivating the requirement to evaluate models' robustness beyond the standard accuracy. Various approaches have been proposed to simulate distribution shifts to construct static robustness evaluation benchmarks, including stress test (Naik et al., 2018), identifying and utilizing spurious correlations (McCoy et al., 2019; Zhang et al., 2019), and domain shifts construction (Hendrycks et al., 2020; Yang et al., 2022). Also, adversarial samples have been involved in robustness benchmarks, including machine-generated (Wang et al., 2021a) or human-in-the-loop generated (Wallace et al., 2019, 2021; Kiela et al., 2021) samples.

Compared to static benchmarks, we motivate to employ automatic attack methods to evaluate models' robustness dynamically, which is more comprehensive and rigorous. Our work is built upon the long-lasting attack-and-defense arms race in adversarial NLP (Wang et al., 2019b; Zhang et al., 2020), mainly absorbing various attack methods. The attack methods can be roughly categorized into char-level, word-level, and sentence-level attacks, corresponding to the hierarchy in our framework. Char-level attacks perturb the texts in the finest granularity, including deleting, inserting, replacing, swapping, and repeating characters (Karpukhin et al., 2019; Gao et al., 2018b). Word-level attacks search for an optimal solution for word substitutions, using external knowledge bases (Ren et al., 2019; Zang et al., 2020) or contextual information (Li et al., 2020; Garg and Ramakrishnan, 2020; Yuan et al., 2021). Sentence-level attacks transform the text considering syntactic patterns (Iyyer et al., 2018), text styles (Qi et al., 2021), and domains (Wang et al., 2020).

6 Conclusion

We present a unified framework, providing solutions to three core challenges in automatic robustness evaluation. We give a further discussion about robustness evaluation in Appendix F. In the future,

we will selectively include more robustness dimensions in our framework.

Limitation

Although we explore diverse robustness dimensions, there are more possible dimensions to cover, and we highly encourage future researchers to complete our paradigm for more comprehensive robustness evaluations. Moreover, our sample selection strategy is base on the perturbation degree. While being effective, this strategy is an approximate suboptimal solution to the problem. We leave finding better selection strategies as future work.

Ethical Consideration

In this section, we discuss the intended use and energy saving considered in our paper.

Intended Use. In this paper, we consider beyond the textual attack-and-defense arms race and highlight the role of adversarial attacks in robustness evaluation. We design a systematic robustness evaluation paradigm to employ adversarial attacks for robustness evaluation. We first summarize deficiencies in current works that limit the further use of adversarial attacks in practical scenarios. Then we propose a standardized paradigm to evaluate the robustness of models using adversarial attacks. We also develop an extensible toolkit to instantiate our paradigm.

Energy Saving. We describe our experimental details to prevent other researchers from unnecessary hyper-parameter adjustments and to help them quickly reproduce our results. We will also release all models we use in our experiments.

Acknowledgements

This work is supported by the National Key R&D Program of China (No. 2020AAA0106502), Major Project of the National SocialScience Foundation of China (No. 22&ZD298), Institute Guo Qiang at Tsinghua University.

Yangyi Chen and Ganqu Cui made the original research proposal and wrote the paper. Hongcheng Gao conducted experiments and helped to organize the paper. Lifan Yuan initiated the codebase and contributed to the proposal. Everyone else participated in the discussion, experiments, and paper writing of this study.

References

Roee Aharoni and Yoav Goldberg. 2020. Unsupervised domain clusters in pretrained language models. In *Proceedings of ACL*.

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. Generating natural language adversarial examples. In *Proceedings of EMNLP*.

Omer Antverg, Eyal Ben-David, and Yonatan Belinkov. 2022. Idani: Inference-time domain adaptation via neuron-level interventions. *arXiv preprint arXiv:2206.00259*.

Max Bartolo, Tristan Thrush, Robin Jia, Sebastian Riedel, Pontus Stenetorp, and Douwe Kiela. 2021. Improving question answering model robustness with synthetic adversarial data generation. In *Proceedings of EMNLP*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Proceedings of NeurIPS*.

Howard Chen, Jacqueline He, Karthik Narasimhan, and Danqi Chen. 2022a. Can rationalization improve robustness? In *Proceedings of NAACL*.

Yangyi Chen, Hongcheng Gao, Ganqu Cui, Fanchao Qi, Longtao Huang, Zhiyuan Liu, and Maosong Sun. 2022b. Why should adversarial perturbations be imperceptible? rethink the research paradigm in adversarial NLP. In *Proceedings of EMNLP*.

Yangyi Chen, Jin Su, and Wei Wei. 2021. Multi-granularity textual adversarial attack with behavior cloning. *arXiv preprint arXiv:2109.04367*.

Alexandra Chronopoulou, Matthew E Peters, and Jesse Dodge. 2021. Efficient hierarchical domain adaptation for pretrained language models. *arXiv preprint arXiv:2112.08786*.

Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2019. Don't take the easy way out: ensemble based methods for avoiding known dataset biases. *arXiv preprint arXiv:1909.03683*.

Ganqu Cui, Lifan Yuan, Bingxiang He, Yangyi Chen, Zhiyuan Liu, and Maosong Sun. 2022. A unified evaluation of textual backdoor learning: Frameworks and benchmarks. In *Proceedings of NeurIPS*.

Chuyun Deng, Mingxuan Liu, Yue Qin, Jia Zhang, Haixin Duan, and Donghong Sun. 2022. Valcat: Variable-length contextualized adversarial transformations using encoder-decoder language model. *ACL rolling review preprint*.

Zhendong Dong and Qiang Dong. 2003. Hownet - a hybrid language and knowledge resource. In *International Conference on Natural Language Processing and Knowledge Engineering, 2003. Proceedings. 2003*.

Steffen Eger and Yannik Benz. 2020. From hero to zéroe: A benchmark of low-level adversarial attacks. In *Proceedings of AACL*.

Steffen Eger, Gözde Gül Şahin, Andreas Rücklé, Ji-Ung Lee, Claudia Schulz, Mohsen Mesgar, Krishnkant Swarnkar, Edwin Simpson, and Iryna Gurevych. 2019. Text processing like humans do: Visually attacking and shielding nlp systems. In *Proceedings of NAACL*.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018a. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *2018 IEEE Security and Privacy Workshops (SPW)*.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018b. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *2018 IEEE Security and Privacy Workshops (SPW)*.

Siddhant Garg and Goutham Ramakrishnan. 2020. BAE: bert-based adversarial examples for text classification. In *Proceedings of EMNLP*.

Karan Goel, Nazneen Rajani, Jesse Vig, Samson Tan, Jason Wu, Stephan Zheng, Caiming Xiong, Mohit Bansal, and Christopher Ré. 2021. Robustness gym: Unifying the nlp evaluation landscape. *arXiv preprint arXiv:2101.04840*.

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

Junliang Guo, Zhirui Zhang, Linlin Zhang, Linli Xu, Boxing Chen, Enhong Chen, and Weihua Luo. 2021. Towards variable-length textual adversarial attacks. *arXiv preprint arXiv:2104.08139*.

Jens Hauser, Zhao Meng, Damián Pascual, and Roger Wattenhofer. 2021. Bert is robust! a case against synonym-based adversarial examples in text classification. *arXiv preprint arXiv:2109.07403*.

Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, and Dawn Song. 2020. Pretrained transformers improve out-of-distribution robustness. *arXiv preprint arXiv:2004.06100*.

Weihua Hu, Gang Niu, Issei Sato, and Masashi Sugiyama. 2018. Does distributionally robust supervised learning give robust classifiers? In *Proceedings of ICML*.

Kuan-Hao Huang and Kai-Wei Chang. 2021. Generating syntactically controlled paraphrases without using annotated parallel pairs. In *Proceedings of EACL*.

Po-Sen Huang, Robert Stanforth, Johannes Welbl, Chris Dyer, Dani Yogatama, Sven Gowal, Krishnamurthy Dvijotham, and Pushmeet Kohli. 2019. Achieving verified robustness to symbol substitutions via interval bound propagation. In *Proceedings of EMNLP*.

J Stuart Hunter. 1986. The exponentially weighted moving average. *Journal of quality technology*.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. 2019. Adversarial examples are not bugs, they are features. *Proceedings of NeurIPS*.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of NAACL*.

Vladimir Karpukhin, Omer Levy, Jacob Eisenstein, and Marjan Ghazvininejad. 2019. Training on synthetic noise improves robustness to natural noise in machine translation. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, et al. 2021. Dynabench: Rethinking benchmarking in nlp. *arXiv preprint arXiv:2104.14337*.

Thai Le, Jooyoung Lee, Kevin Yen, Yifan Hu, and Dongwon Lee. 2022. Perturbations in the wild: Leveraging human-written text perturbations for realistic adversarial attack and defense. *arXiv preprint arXiv:2203.10346*.

Dianqi Li, Yizhe Zhang, Hao Peng, Liqun Chen, Chris Brockett, Ming-Ting Sun, and Bill Dolan. 2021. Contextualized perturbation for textual adversarial attack. In *Proceedings of NAACL*.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. 2018. Textbugger: Generating adversarial text against real-world applications. *arXiv preprint arXiv:1812.05271*.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. 2020. BERT-ATTACK: Adversarial attack against BERT using BERT. In *Proceedings of EMNLP*.

Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. 2017. Deep text classification can be fooled. *arXiv preprint arXiv:1704.08006*.

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*.

R Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *arXiv preprint arXiv:1902.01007*.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*.

John P Miller, Rohan Taori, Aditi Raghunathan, Shiori Sagawa, Pang Wei Koh, Vaishaal Shankar, Percy Liang, Yair Carmon, and Ludwig Schmidt. 2021. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In *International Conference on Machine Learning*. PMLR.

John Morris, Eli Lifland, Jack Lanchantin, Yangfeng Ji, and Yanjun Qi. 2020a. Reevaluating adversarial examples in natural language. In *Findings of EMNLP*.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020b. TextAttack: A framework for adversarial attacks, data augmentation, and adversarial training in NLP. In *Proceedings of EMNLP*.

Maximilian Mozes, Pontus Stenetorp, Bennett Kleinberg, and Lewis Griffin. 2021. Frequency-guided word substitutions for detecting textual adversarial examples. In *Proceedings of EACL*.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. Stress test evaluation for natural language inference. In *Proceedings of COLING*.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2019. Adversarial nli: A new benchmark for natural language understanding. *arXiv preprint arXiv:1910.14599*.

Yonatan Oren, Shiori Sagawa, Tatsunori B Hashimoto, and Percy Liang. 2019. Distributionally robust language modeling. *arXiv preprint arXiv:1909.02060*.

Danish Pruthi, Bhuwan Dhingra, and Zachary C. Lipton. 2019. Combating adversarial misspellings with robust word recognition. In *Proceedings of ACL*.

Fanchao Qi, Yangyi Chen, Xurui Zhang, Mukai Li, Zhiyuan Liu, and Maosong Sun. 2021. Mind the style of text! adversarial and backdoor attacks based on text style transfer. In *Proceedings of EMNLP*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of EMNLP*.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of ACL*.

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

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

Jiao Sun, Xuezhe Ma, and Nanyun Peng. 2021. Aesop: Paraphrase generation with adaptive syntactic control. In *Proceedings of EMNLP*.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *International conference on machine learning*.

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*.

Samson Tan, Shafiq Joty, Min-Yen Kan, and Richard Socher. 2020. It's morphin'time! combating linguistic discrimination with inflectional perturbations. *arXiv preprint arXiv:2005.04364*.

Lifu Tu, Garima Lalwani, Spandana Gella, and He He. 2020. An empirical study on robustness to spurious correlations using pre-trained language models. *TACL*.

Eric Wallace, Pedro Rodriguez, Shi Feng, Ikuya Yamada, and Jordan Boyd-Graber. 2019. Trick me if you can: Human-in-the-loop generation of adversarial examples for question answering. *Transactions of the Association for Computational Linguistics*.

Eric Wallace, Adina Williams, Robin Jia, and Douwe Kiela. 2021. Analyzing dynamic adversarial training data in the limit. *arXiv preprint arXiv:2110.08514*.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019a. Superglue: A stickier benchmark for general-purpose language understanding systems. *Proceedings of NeurIPS*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Boxin Wang, Chejian Xu, Shuohang Wang, Zhe Gan, Yu Cheng, Jianfeng Gao, Ahmed Hassan Awadallah, and Bo Li. 2021a. Adversarial glue: A multi-task benchmark for robustness evaluation of language models. *arXiv preprint arXiv:2111.02840*.

Tianlu Wang, Xuezhi Wang, Yao Qin, Ben Packer, Kang Li, Jilin Chen, Alex Beutel, and Ed Chi. 2020. CAT-gen: Improving robustness in NLP models via controlled adversarial text generation. In *Proceedings of EMNLP*.

Wenqi Wang, Run Wang, Lina Wang, Zhibo Wang, and Aoshuang Ye. 2019b. Towards a robust deep neural network in texts: A survey. *arXiv preprint arXiv:1902.07285*.

Xiao Wang, Qin Liu, Tao Gui, Qi Zhang, Yicheng Zou, Xin Zhou, Jiacheng Ye, Yongxin Zhang, Rui Zheng, Zexiong Pang, Qinzhuo Wu, Zhengyan Li, Chong Zhang, Ruotian Ma, Zichu Fei, Ruijian Cai, Jun Zhao, Xingwu Hu, Zhiheng Yan, Yiding Tan, Yuan Hu, Qiyuan Bian, Zhihua Liu, Shan Qin, Bolin Zhu, Xiaoyu Xing, Jinlan Fu, Yue Zhang, Minlong Peng, Xiaoqing Zheng, Yaqian Zhou, Zhongyu Wei, Xipeng Qiu, and Xuanjing Huang. 2021b. TextFlint: Unified multilingual robustness evaluation toolkit for natural language processing. In *Proceedings of ACL*.

Xiaosen Wang, Jin Hao, Yichen Yang, and Kun He. 2021c. Natural language adversarial defense through synonym encoding. In *Uncertainty in Artificial Intelligence*.

Jun Yan, Yang Xiao, Sagnik Mukherjee, Bill Yuchen Lin, Robin Jia, and Xiang Ren. 2021. On the robustness of reading comprehension models to entity renaming. *arXiv preprint arXiv:2110.08555*.

Linyi Yang, Shuibai Zhang, Libo Qin, Yafu Li, Yidong Wang, Hanmeng Liu, Jindong Wang, Xing Xie, and Yue Zhang. 2022. Glue-x: Evaluating natural language understanding models from an out-of-distribution generalization perspective. *arXiv preprint arXiv:2211.08073*.

Jin Yong Yoo, John X Morris, Eli Lifland, and Yanjun Qi. 2020. Searching for a search method: Benchmarking search algorithms for generating nlp adversarial examples. *arXiv preprint arXiv:2009.06368*.

Lifan Yuan, Yichi Zhang, Yangyi Chen, and Wei Wei. 2021. Bridge the gap between cv and nlp! a gradient-based textual adversarial attack framework. *arXiv preprint arXiv:2110.15317*.

Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. 2020. Word-level textual adversarial attacking as combinatorial optimization. In *Proceedings of ACL*.

Guoyang Zeng, Fanchao Qi, Qianrui Zhou, Tingji Zhang, Zixian Ma, Bairu Hou, Yuan Zang, Zhiyuan Liu, and Maosong Sun. 2021. OpenAttack: An open-source textual adversarial attack toolkit. In *Proceedings of ACL*.

Wei Emma Zhang, Quan Z Sheng, Ahoud Alhazmi, and Chenliang Li. 2020. Adversarial attacks on deep-learning models in natural language processing: A survey. *ACM Transactions on Intelligent Systems and Technology (TIST)*.

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

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. Paws: Paraphrase adversaries from word scrambling. *arXiv preprint arXiv:1904.01130*.

Ruiqi Zhong, Charlie Snell, Dan Klein, and Jacob Steinhardt. 2022. Summarizing differences between text distributions with natural language. *arXiv preprint arXiv:2201.12323*.

Chunting Zhou, Xuezhe Ma, Paul Michel, and Graham Neubig. 2021. Examining and combating spurious features under distribution shift. In *Proceedings of ICML*.

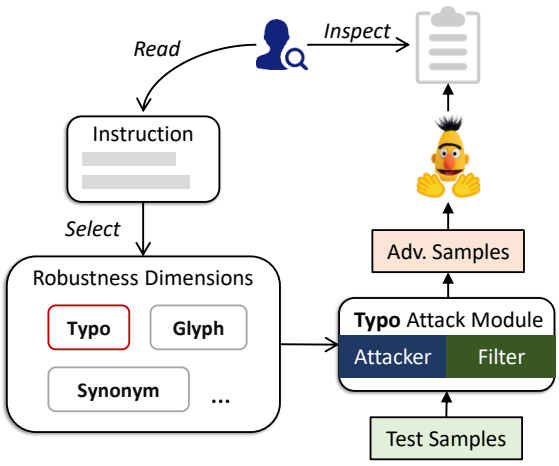


Figure 6: Our robustness evaluation framework. The "Instruction" refers to the written guidance for robustness evaluation.

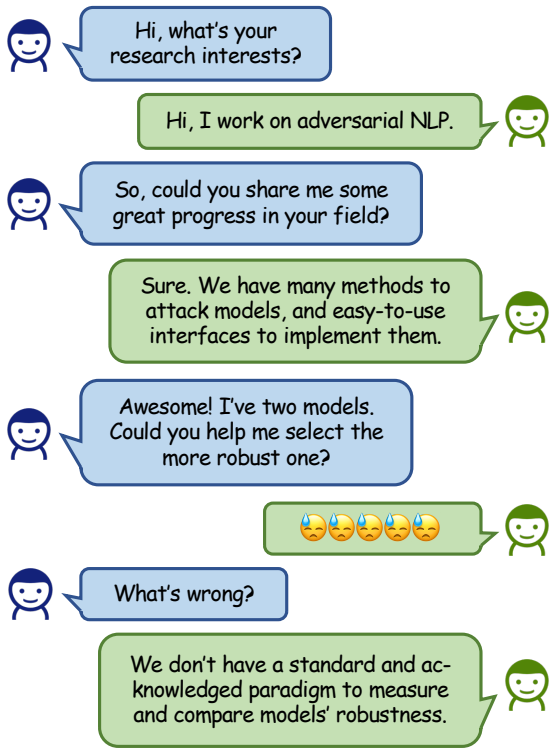


Figure 7: The current dilemma in adversarial NLP.

A Validity of Adversarial Samples

The original definition of adversarial samples in computer vision requires the perturbation to be imperceptible to human (Goodfellow et al., 2014). However in NLP, texts are made of discrete tokens, where the changes are more apparent and difficult to measure. Therefore, the common practice in adversarial NLP recommend to evaluate the *validity* of adversarial samples, which measures whether the transformed samples preserve the same meanings with the original samples, considering only the rationale part (a.k.a., the contents that determine the golden label). More precisely, valid adversarial samples preserve (1) the original labels and (2) the semantics of the rational part.

B Justification of Perceptible Perturbations

Consider the sample crafted by adding typos: "I love the way that it took chancs and really asks you to takke these great leaps of faith and pays off." The common belief in adversarial NLP is to make the perturbations as small as possible. So this sample with obvious perturbations highlighted in red will be dismissed in previous work. But in our robustness evaluation framework, the requirement is to employ attack methods to simulate real-world inputs, which may contain some so-called perceptible perturbations like the above example. Thus, we include various kinds of samples with perceptible perturbations in our framework provided that they can simulate real-world inputs well.

C Computation of Perturbation Degree

For three transformation levels, we employ different computational methods to measure the perturbation degree. For char-level transformations with the malicious tag, we adopt the relative Levenshtein Distance. For char-level transformations with the general tag, we restrict the algorithms to perturb less than two characters for each word to better simulate inputs from benign users and adopt the word modification rate to measrue the perturbation degree. For word-level transformations, we employ the word modification rate. For sentence-level transformations, we employ embedding similarity. Next, we introduce how to compute these measurements.

Relative Edit Distance. We use relative edit distance to measure the perturbation degree of char-level attacks with the malicious tag. Assume that the original text has N_c characters in total. We modify n_c characters in original text X and get a new text X'. Then the Edit Distance between X and X' is n_c, and the perturbation degree is:

$$D_c = \frac{n_c}{N_c}.$$

Word Modification Rate. We use word modification rate to measure the perturbation degree of

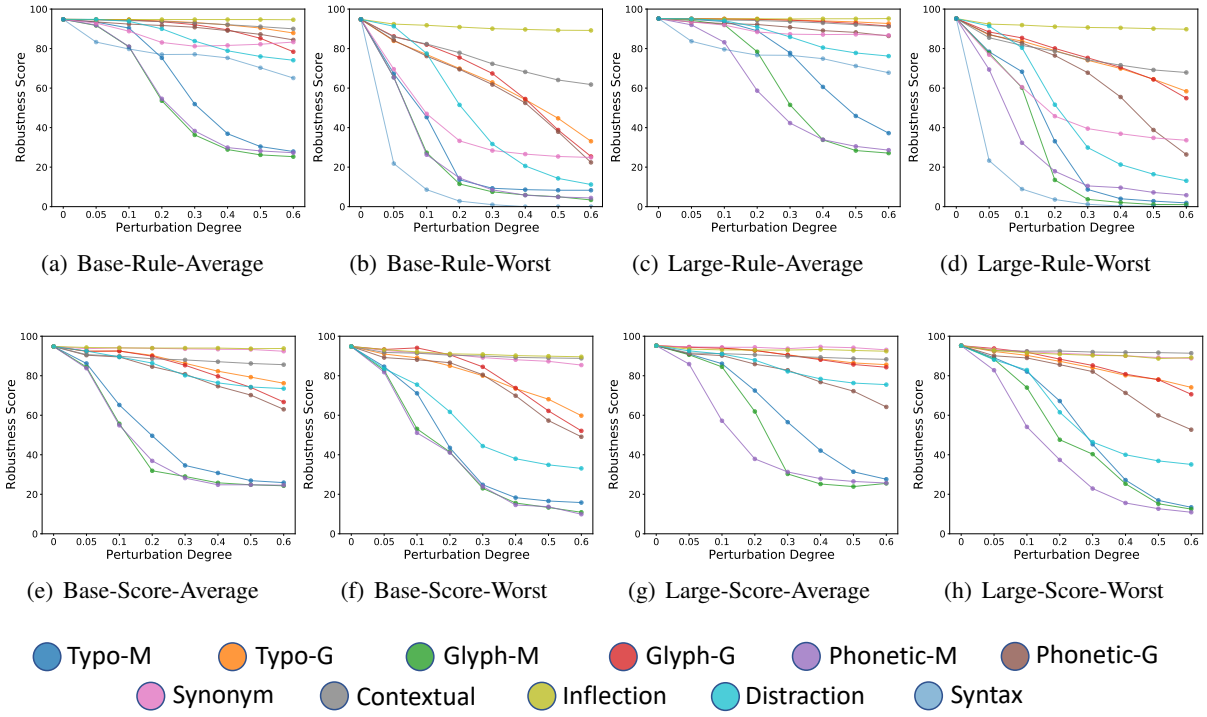


Figure 8: Comprehensive results of RoBERTa-base (Base) and RoBERTa-large (Large) on AG's News. We consider rule-based (Rule) and score-based (Score) attacks, and worst (Worst) and average (Average) performance estimation.

char-level attacks with the general tag and word-level attacks. Assume that the original text has N_w words in total, and we perturb n_w words. Then the perturbation degree is:

$$D_w = \frac{n_w}{N_w}.$$

Specifically for char-level attack, we only conduct one char-level modification for each perturbed word.

Embedding Similarity. We adopt embedding similarity to measure the perturbation degree of sentence-level attack. We get the sentence embeddings with Sentence-Transformers (Reimers and Gurevych, 2019). Denote the sentence embedding of original sentence $\boldsymbol{x}$, the transformed sentence embedding as $\boldsymbol{x'}$, and the embedding similarity between $\boldsymbol{x}$ and $\boldsymbol{x'}$ is calculated by cosine function $\cos(\boldsymbol{x}, \boldsymbol{x'})$. We compute the cosine similarity between two embeddings. Then the degree is:

$$D_s = 1 - \cos\left(\boldsymbol{x}, \boldsymbol{x'}\right).$$

D Human Annotation

D.1 Annotation Details

We conduct human annotation to evaluate the validity of adversarial samples generated by different methods at different perturbation degrees. We employ 3 human annotators, and use the voting strategy to produce the annotation results. For each method and perturbation degree, we sample 50 successful adversarial samples. The final score is averaged over all 50 adversarial samples. Specifically for the annotation, we show annotators the original sample, the perturbed sample, and the original label, and ask annotators to give a binary score. 1 represents (1) the original label is the same in the perturbed sample, and (2) the semantic preservation of the rationale part is good. 0 indicates that either rule is not satisfied, or the perturbed sample is hard to comprehend. Note that we don't let the annotators to predict the labels of the perturbed samples and check the label consistency since validity is a higher-standard task that requires semantics invariance.

In the annotation process, we first write an annotation document containing some cases and instructions for annotators. Then we compose some cases to test the annotators. Only qualified annotators are

Degree	Typo-M	Glyph-M	Phonetic-M	Typo-G	Glyph-G	Phonetic-G	Synonym	Contextual	Inflection	Syntax	Distraction
0.05	0.96	1	1	1	1	1	0.44	0.46	1	-	0.98
0.1	0.94	0.98	1	1	1	1	0.32	0.44	1	0.28	0.94
0.3	0.26	0.94	1	1	1	1	0.20	0.32	1	0.06	0.94
0.5	0.06	0.86	1	0.82	1	1	0.14	0.20	0.98	0.02	0.82
0.8	0.02	0.70	0.98	0.64	1	0.98	0.14	0.06	0.98	0	0.64

Table 3: Human annotation of samples validity considering five perturbation degrees and all attack methods.

involved in the final annotation task.

D.2 Annotation Results

The human annotation results to verify the intuition that adversarial samples with higher perturbation degrees are more likely to become invalid are listed in Table 3. Additionally, it is pertinent to mention that our evaluation methodology for assessing validity can also be applied to textual backdoor learning, which faces the same evaluation challenge (Cui et al., 2022).

E Additional Result

We list results on AG's News in Figure 8 and results on Jigsaw in Figure 9.

F Discussion

Chen et al. (2022b) categorizes four different roles of textual adversarial samples. In this paper, we consider how to employ adversarial attacks for automatic robustness evaluation, corresponding to the defined evaluation role. In this section, we give a further discussion about potential future directions on adversarial NLP for robustness evaluation, considering both the attack and the defense sides.

F.1 Adversarial Attack

Complemented robustness dimension We consider general and representative robustness dimensions in our framework. We hope that future work can identify more important dimensions spanning three transformation levels to complement the framework. Specifically, task-specific dimensions can be explored for more specific and comprehensive evaluation.

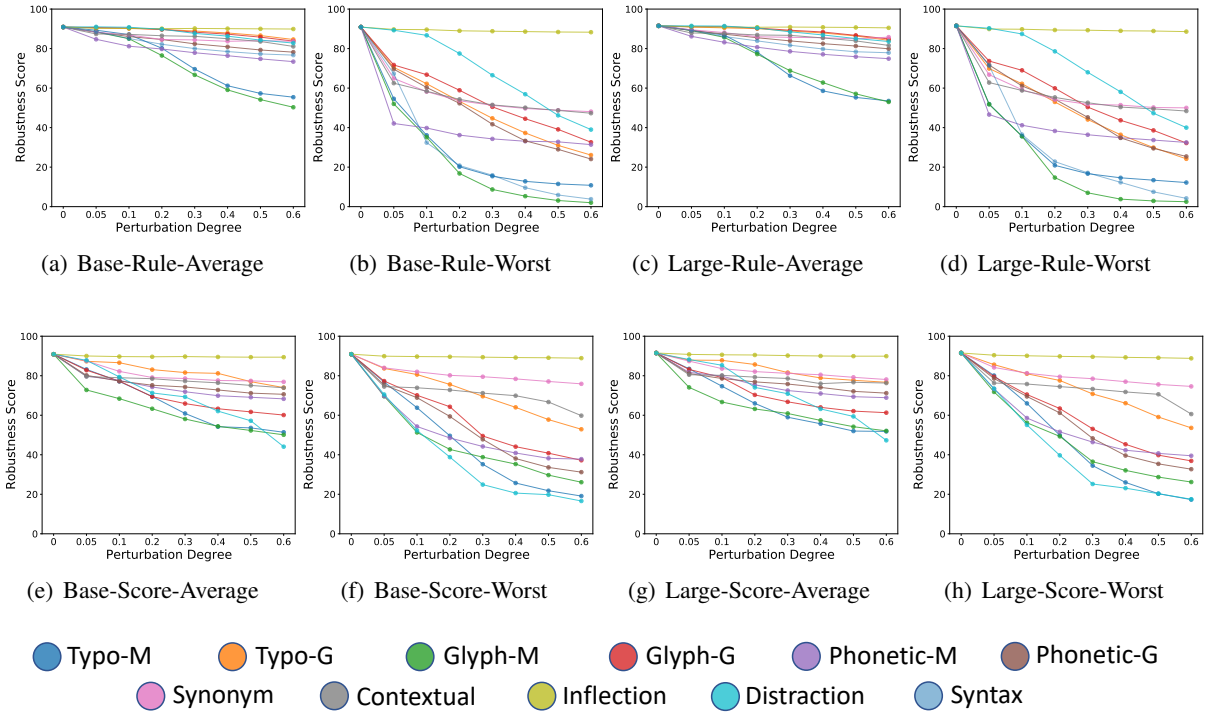
Reliable evaluation For invalid adversarial sample filtering, we employ a heuristic weighted average in our framework. Further improvement is needed for a more reliable robustness estimation. The potential directions are: (1) Identify specific metrics that are justifiable for expected valid adversarial samples; (2) Thoroughly investigate the problem of validity-aware robustness evaluation. For example, one can improve our method by using the human annotation results to better characterize the difference between various attack methods since there exist methods that can craft valid adversarial samples even in high perturbation degrees. Thus, the human annotation scores can serve as weights to average robustness scores computed at different perturbation degrees.

Develop methods based on the model-centric evaluation. The motivation of this paper is to bring out the more practical significance of attack methods. The core part is to shift towards model-centric robustness evaluation and consider how attack methods can actually contribute to the practitioners. Thus, we recommend future research make a mild shift in method development to better fit the model-centric robustness evaluation scene. For example, the central problem in the adversarial arms race era is how to make the attack methods stronger to achieve a higher attack success rate and beat the defense methods. Now the model-centric evaluation requires that the attack methods can better reveal practical, important, and diversified vulnerabilities in models.

Additional work We note that there are some adversarial methods that don't fit into our paradigm because we cannot clearly describe the concrete distribution shift, including challenging samples generated by the human-in-the-loop process (Wallace et al., 2019, 2021; Kiela et al., 2021), non-dimension-specified attack methods (Bartolo et al., 2021; Guo et al., 2021; Deng et al., 2022). Future works can explore characterizing the distribution shift through natural language (Zhong et al., 2022) or model estimation (Aharoni and Goldberg, 2020; Chronopoulou et al., 2021) to include more dimensions in the evaluation framework.

F.2 Adversarial Defense.

In our evaluation framework, we don't approach the defense side. We leave it for future work. Here we discuss how we consider adversarial defense methods and how we can benefit from them.



(a) Base-Rule-Average (b) Base-Rule-Worst (c) Large-Rule-Average (d) Large-Rule-Worst

(e) Base-Score-Average (f) Base-Score-Worst (g) Large-Score-Average (h) Large-Score-Worst

Figure 9: Comprehensive results of RoBERTa-base (Base) and RoBERTa-large (Large) on Jigsaw. We consider rule-based (Rule) and score-based (Score) attacks, and worst (Worst) and average (Average) performance estimation.

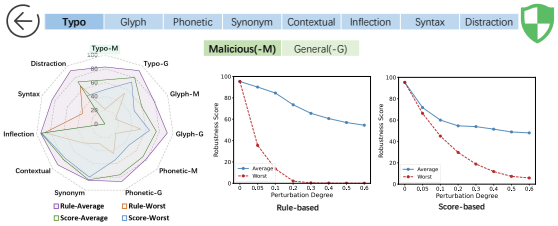
Current practices often situate their defense methods in the scenario of malicious attacks. We present an alternative perspective that accompanies our framework. As adversarial attack methods can be employed to generate samples from different distributions, defense methods can also be employed to deal with out-of-distribution samples, which can address the challenge of diverse inputs from different users or attackers. However, the deficiency in current defense methods is that they mostly can only tackle a specific kind of distribution shift. For example, Pruthi et al. (2019) consider samples containing typos. Wang et al. (2021c) consider rich vocabulary of real-world users. Currently, a generalized and widely applicable defense method is lacking. The promising directions include: (1) Inference-time adaptation (Antverg et al., 2022); (2) Learning robust features from in-distribution data (Ilyas et al., 2019; Clark et al., 2019; Zhou et al., 2021); (3) Distributionally robust optimization (Hu et al., 2018; Oren et al., 2019).

G Single-model Robustness Report

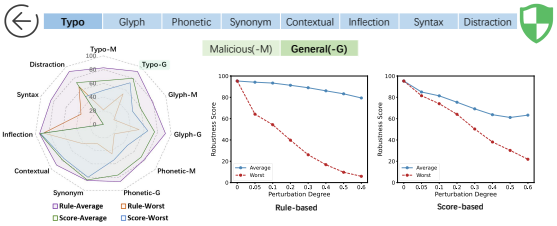
We show robustness reports of two models and three datasets. The robustness reports for RoBERTa-base are shown in Figure 10 (SST-2), Figure 12 (AG's News), and Figure 14 (Jigsaw). The robustness reports for RoBERTa-large are shown in Figure 11 (SST-2), Figure 13 (AG's News), and Figure 15 (Jigsaw).

H Robustness Comparison Report

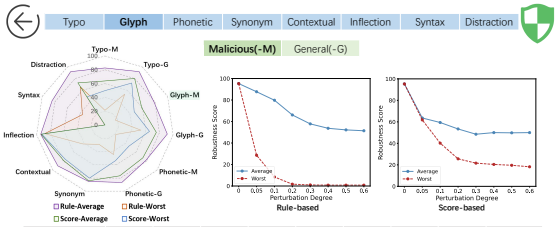
We show the robustness report that compares the two models' robustness in Figure 16 (rule-based evaluation) and Figure 17 (score-based evaluation).



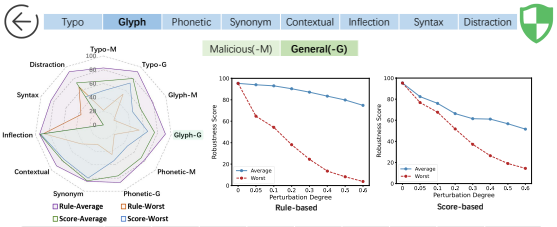
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.3	90.1	84.5	73.5	65.5	60.6	56.9	54.4	*82.7*
Rule-Worst	95.3	35.5	13.5	2.1	0.4	0.2	0.1	0.1	*21.4*
Score-Average	95.3	71.7	60.0	54.6	53.9	51.5	48.9	48.0	*63.8*
Score-Worst	95.3	66.3	45.0	29.7	18.8	11.8	7.3	5.8	*49.8*



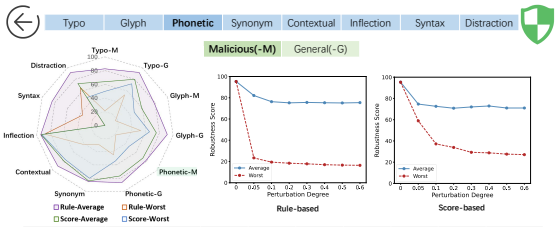
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.3	94.2	93.4	91.3	89.0	86.1	83.3	79.4	*92.0*
Rule-Worst	95.3	64.2	54.3	39.8	26.1	16.8	9.7	5.9	*53.0*
Score-Average	95.3	85.0	81.4	75.4	69.2	63.7	61.1	63.3	*80.0*
Score-Worst	95.3	81.5	74.0	64.1	50.3	38.1	30.2	21.9	*72.2*



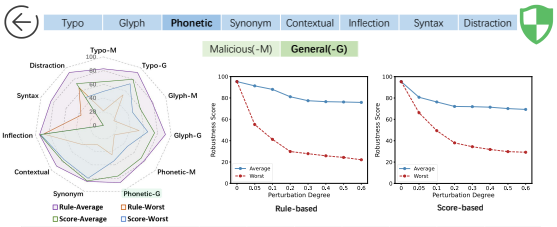
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.3	87.7	79.8	66.1	57.8	53.7	52.2	51.4	*78.6*
Rule-Worst	95.3	28.6	8.4	1.7	1.0	0.8	0.8	0.8	*16.7*
Score-Average	95.3	63.6	59.4	53.4	48.5	50.0	49.8	50.0	*59.1*
Score-Worst	95.3	61.8	40.2	25.6	21.6	20.4	19.6	18.2	*46.6*



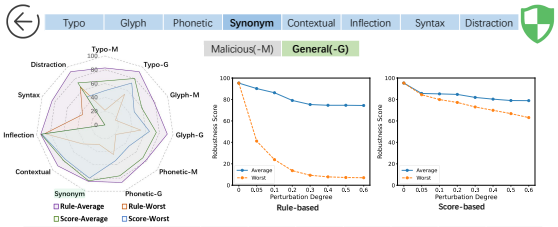
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.3	94.0	93.0	90.3	87.1	83.4	79.8	74.8	*91.4*
Rule-Worst	95.3	64.7	54.3	38.1	24.5	13.5	8.2	3.8	*52.8*
Score-Average	95.3	82.4	76.0	66.3	61.5	61.1	56.7	51.6	*75.5*
Score-Worst	95.3	76.8	67.6	51.8	37.2	26.4	19.0	14.4	*65.3*



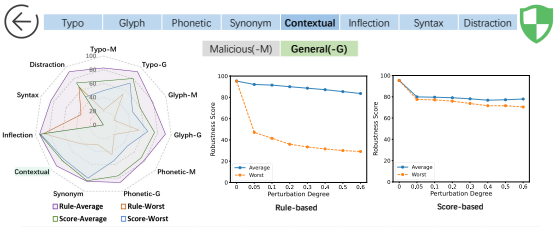
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.3	82.1	76.3	75.2	75.6	75.3	75.1	75.4	*78.4*
Rule-Worst	95.3	23.4	19.4	18.4	17.8	17.0	16.6	16.4	*20.9*
Score-Average	95.3	74.7	72.6	70.8	72.0	72.9	71.0	71.0	*72.8*
Score-Worst	95.3	58.9	37.2	33.9	29.3	28.8	27.6	27.1	*46.4*



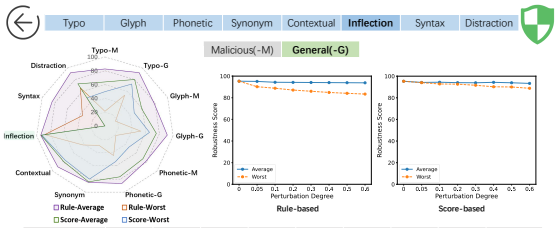
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.3	91.3	88.0	81.1	77.4	76.5	76.2	75.8	*86.8*
Rule-Worst	95.3	55.1	41.2	29.8	27.8	25.8	24.3	22.1	*44.7*
Score-Average	95.3	80.7	76.3	72.1	71.9	71.4	70.1	69.3	*76.8*
Score-Worst	95.3	66.2	49.4	38.0	34.4	31.8	29.8	29.2	*54.0*



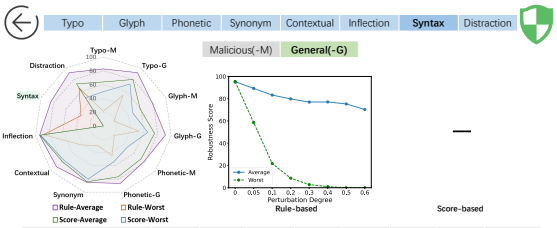
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.3	90.2	86.3	79.1	75.3	74.6	74.6	74.4	*85.3*
Rule-Worst	95.3	41.2	23.9	13.6	9.3	7.8	7.2	7.0	*29.3*
Score-Average	95.3	85.6	85.2	84.7	81.9	80.3	79.0	78.9	*84.2*
Score-Worst	95.3	84.4	79.9	77.2	73.0	69.9	66.8	63.1	*80.1*



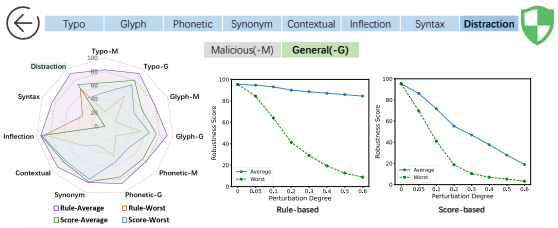
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.3	92.1	91.5	90.0	88.6	87.2	85.4	83.6	*90.4*
Rule-Worst	95.3	47.1	41.5	35.9	33.4	31.5	30.0	29.1	*42.2*
Score-Average	95.3	79.9	79.6	79.1	78.1	76.8	77.2	77.9	*78.8*
Score-Worst	95.3	77.5	77.1	75.9	73.7	71.6	71.6	70.4	*76.0*



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.3	95.1	94.3	94.2	94.1	94.0	93.9	93.8	*93.9*
Rule-Worst	95.3	90.2	88.8	87.1	86.0	84.9	84.1	83.4	*88.2*
Score-Average	95.3	94.2	94.4	94.0	93.9	94.3	93.9	93.3	*93.5*
Score-Worst	95.3	94.0	92.8	92.6	91.6	90.2	90.0	88.8	*92.4*



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.3	89.3	83.3	79.8	77.0	77.1	75.3	70.3	*84.4*
Rule-Worst	95.3	58.6	21.8	8.6	2.8	1.0	0.0	0.0	*36.0*
Score-Average	-	-	-	-	-	-	-	-	-
Score-Worst	-	-	-	-	-	-	-	-	-

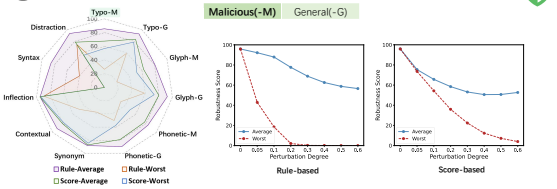


Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.3	94.7	93.0	90.0	88.5	87.1	85.8	84.5	*92.1*
Rule-Worst	95.3	84.4	64.0	41.2	29.0	19.3	12.6	8.8	*66.0*
Score-Average	95.3	86.1	71.6	55.3	46.9	37.7	27.9	19.0	*72.6*
Score-Worst	95.3	69.6	41.0	18.8	10.4	7.0	5.2	3.2	*48.4*

Figure 10: Robustness report for RoBERTa-base on SST-2.

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

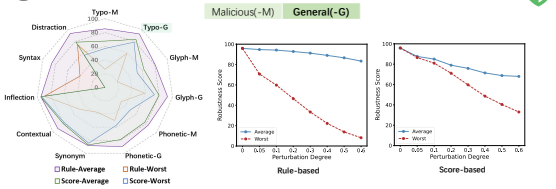
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.9	92.2	87.9	77.7	68.9	62.6	58.8	56.6	*85.4*
Rule-Worst	95.9	42.9	18.5	2.1	0.2	0.1	0.0	0.0	*26.4*
Score-Average	95.9	75.4	65.7	58.5	53.2	50.6	50.7	52.7	*67.5*
Score-Worst	95.9	73.6	54.3	36.0	22.4	12.3	7.2	4.0	*56.8*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

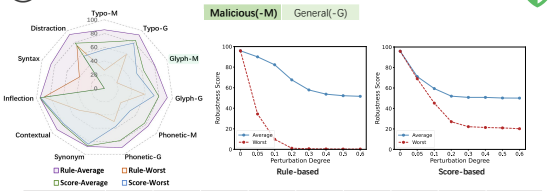
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.9	94.7	94.2	92.8	91.2	89.0	86.6	83.4	*93.0*
Rule-Worst	95.9	70.7	59.8	46.5	33.4	22.2	13.8	8.1	*59.2*
Score-Average	95.9	87.6	85.0	79.0	75.9	71.4	68.8	68.0	*83.5*
Score-Worst	95.9	86.5	81.0	71.1	59.7	48.6	40.4	33.0	*78.5*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

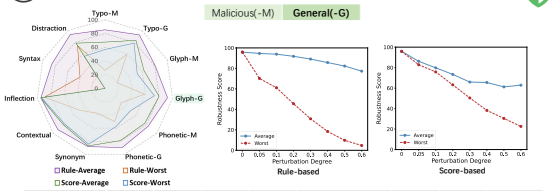
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.9	90.0	82.3	67.6	57.8	53.7	52.2	51.6	*80.5*
Rule-Worst	95.9	34.4	9.8	1.1	0.6	0.4	0.3	0.3	*19.8*
Score-Average	95.9	71.1	59.4	52.0	50.8	50.8	50.2	50.1	*62.8*
Score-Worst	95.9	69.0	45.2	27.0	22.2	21.4	21.0	20.2	*51.7*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

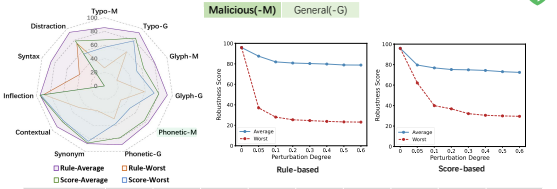
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.9	94.6	93.9	91.8	89.1	85.7	82.3	77.3	*92.4*
Rule-Worst	95.9	70.2	61.1	45.5	30.8	18.4	9.8	4.7	*58.8*
Score-Average	95.9	86.2	79.8	73.4	65.9	65.5	61.2	62.9	*79.8*
Score-Worst	95.9	82.8	75.8	63.2	50.4	38.2	30.4	22.6	*73.2*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

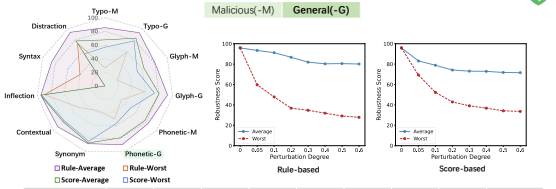
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.9	87.5	81.9	80.8	80.3	79.8	78.9	78.8	*83.7*
Rule-Worst	95.9	37.0	28.0	25.4	24.6	23.8	23.3	23.1	*31.5*
Score-Average	95.9	79.6	76.8	75.2	74.9	74.3	73.1	72.4	*77.1*
Score-Worst	95.9	62.1	39.9	36.8	32.1	30.5	29.9	29.5	*49.3*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

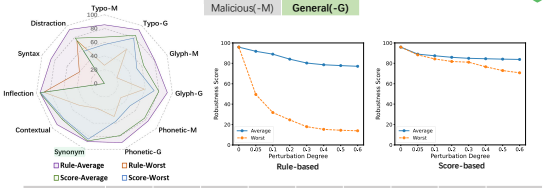
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.9	93.4	91.2	86.7	81.9	80.3	80.5	80.1	*89.8*
Rule-Worst	95.9	59.7	47.8	36.8	34.7	31.9	29.1	27.8	*50.2*
Score-Average	95.9	83.1	78.9	74.2	73.1	72.9	71.8	71.6	*79.1*
Score-Worst	95.9	69.3	52.3	42.9	39.1	36.8	34.2	33.6	*57.5*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

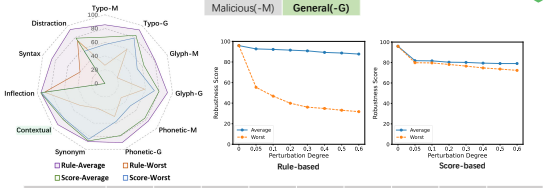
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.9	91.8	89.0	84.1	80.3	78.6	77.8	77.1	*88.0*
Rule-Worst	95.9	49.5	31.8	24.6	17.9	15.3	14.4	13.9	*37.7*
Score-Average	95.9	89.0	87.4	85.9	84.9	84.5	84.1	83.8	*87.0*
Score-Worst	95.9	88.3	84.3	81.8	81.1	76.6	72.9	70.7	*84.6*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

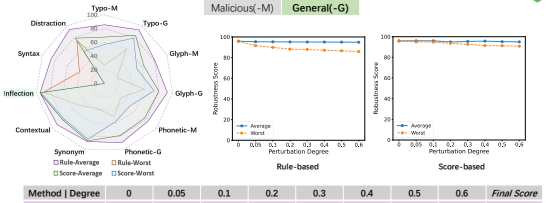
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.9	92.7	92.2	91.5	90.8	89.3	88.7	87.7	*91.4*
Rule-Worst	95.9	55.4	46.8	39.9	36.1	34.8	33.1	31.7	*48.5*
Score-Average	95.9	81.8	81.6	80.3	80.1	79.4	79.0	79.0	*80.7*
Score-Worst	95.9	79.8	79.6	78.1	76.5	74.7	73.6	72.3	*78.4*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

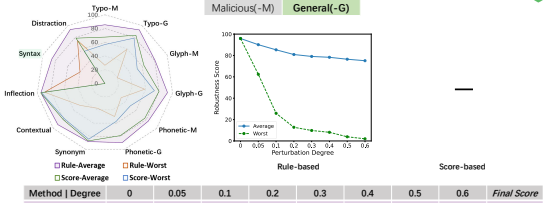
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.9	95.4	95.3	95.2	95.1	95.1	95.0	94.9	*94.6*
Rule-Worst	95.9	91.6	89.9	88.2	87.9	87.1	86.6	85.8	*89.5*
Score-Average	95.9	96.0	96.0	94.9	95.4	95.7	95.2	95.0	*95.0*
Score-Worst	95.9	95.0	94.8	93.6	92.6	91.4	91.2	90.8	*93.7*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

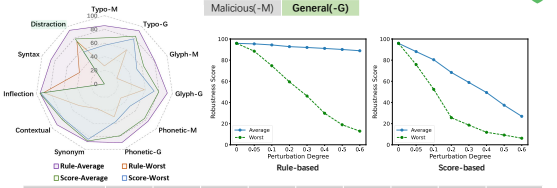
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.9	90.1	85.3	80.9	79.1	78.3	76.5	75.1	*85.7*
Rule-Worst	95.9	62.4	25.9	12.8	9.8	8.1	3.9	2.0	*40.2*
Score-Average	-	-	-	-	-	-	-	-	-
Score-Worst	-	-	-	-	-	-	-	-	-

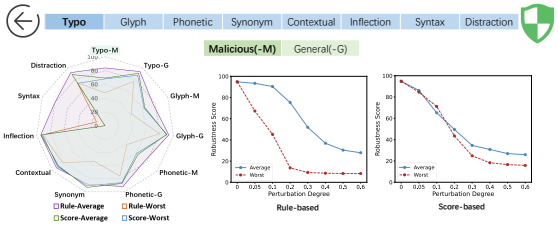
Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

Malicious(-M) | General(-G)

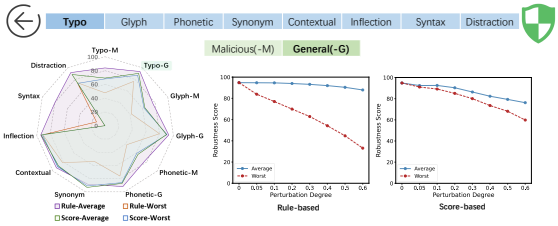


Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.9	95.4	94.4	92.8	92.0	91.1	90.1	88.9	*93.6*
Rule-Worst	95.9	88.5	74.7	59.5	46.1	29.8	18.9	12.9	*74.6*
Score-Average	95.9	88.1	80.4	68.4	58.9	49.4	37.4	26.9	*78.7*
Score-Worst	95.9	75.8	52.4	25.6	18.6	11.8	9.2	6.2	*55.9*

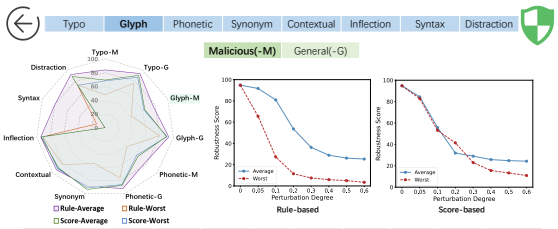
Figure 11: Robustness report for RoBERTa-large on SST-2.



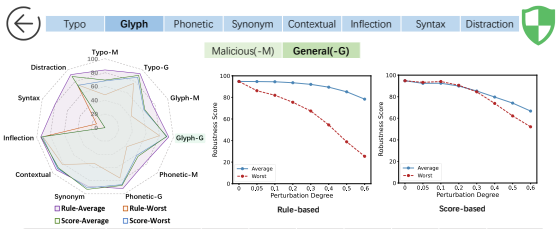
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	94.8	93.4	90.4	75.3	51.9	36.9	30.4	27.9	*83.8*
Rule-Worst	94.8	67.3	45.3	13.6	9.3	8.6	8.3	8.3	*47.7*
Score-Average	94.8	86.2	65.2	49.6	34.6	30.8	26.9	25.9	*69.3*
Score-Worst	94.8	84.6	71.1	43.5	24.8	18.3	16.6	15.8	*68.0*



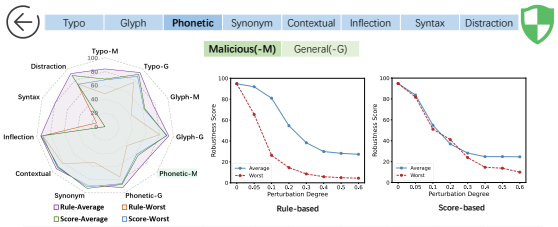
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	94.8	94.7	94.6	94.0	93.2	92.0	90.4	87.9	*93.5*
Rule-Worst	94.8	83.9	77.0	69.9	62.9	54.2	44.7	33.1	*76.5*
Score-Average	94.8	92.4	92.5	90.3	86.3	82.3	79.3	76.2	*90.4*
Score-Worst	94.8	91.0	89.1	85.0	80.0	73.5	68.1	59.8	*87.2*



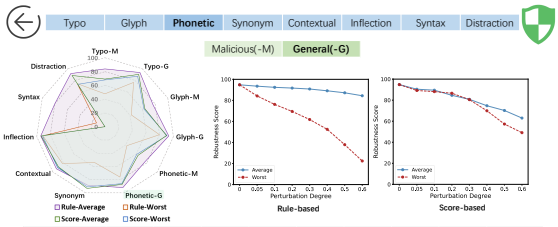
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	94.8	91.7	80.9	53.6	36.3	28.9	26.2	25.3	*76.6*
Rule-Worst	94.8	65.5	27.3	11.5	7.5	5.9	5.0	3.4	*41.8*
Score-Average	94.8	84.4	55.7	31.9	29.0	25.8	24.8	24.3	*63.3*
Score-Worst	94.8	82.8	53.1	41.4	23.0	15.6	13.3	10.9	*62.1*



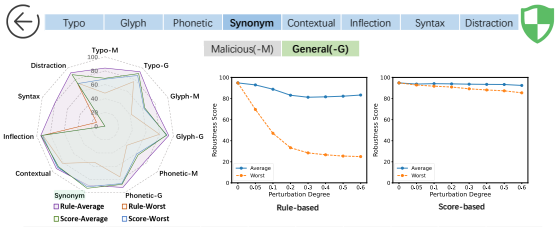
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	94.8	94.7	94.5	93.6	92.1	89.5	85.2	78.4	*93.2*
Rule-Worst	94.8	86.2	82.0	75.5	67.4	54.5	38.8	25.4	*79.8*
Score-Average	94.8	92.4	92.5	89.9	85.3	79.7	74.1	66.7	*90.1*
Score-Worst	94.8	93.3	94.1	90.6	84.5	73.8	62.2	52.1	*90.5*



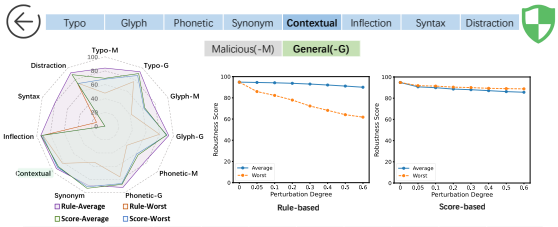
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	94.8	91.9	81.0	54.6	38.3	29.9	28.2	27.3	*77.0*
Rule-Worst	94.8	65.4	26.3	14.5	8.5	5.8	4.9	4.4	*41.9*
Score-Average	94.8	83.8	54.9	36.9	28.2	24.8	24.8	24.6	*63.4*
Score-Worst	94.8	81.8	51.1	41.1	23.9	14.6	13.7	9.9	*61.1*



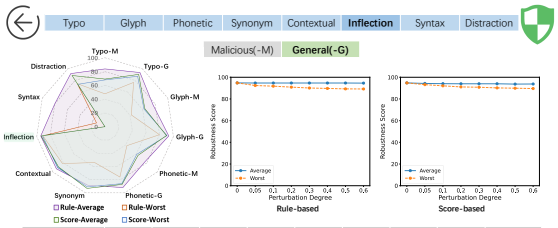
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	94.8	93.5	92.4	91.7	90.8	89.1	87.2	84.4	*91.8*
Rule-Worst	94.8	84.2	76.2	69.5	61.8	52.5	37.9	22.4	*76.1*
Score-Average	94.8	90.4	89.5	84.6	80.8	74.7	70.2	63.0	*87.1*
Score-Worst	94.8	89.2	88.1	86.6	80.5	69.9	57.3	49.1	*85.9*



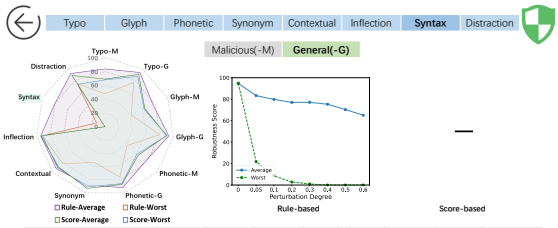
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	94.8	92.9	88.8	83.1	81.2	81.6	82.2	83.3	*88.6*
Rule-Worst	94.8	69.6	47.0	33.3	28.4	26.6	25.4	24.9	*53.9*
Score-Average	94.8	93.7	94.1	93.9	93.6	93.4	93.3	92.4	*93.1*
Score-Worst	94.8	92.8	91.7	90.9	89.2	88.1	87.3	85.4	*91.0*



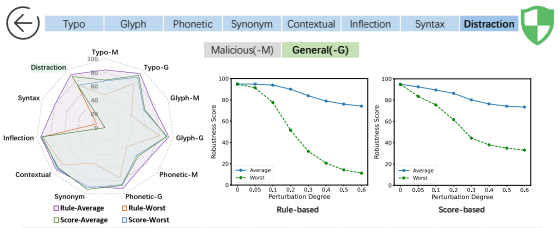
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	94.8	94.5	94.2	93.7	93.0	92.1	91.1	90.0	*93.3*
Rule-Worst	94.8	85.9	82.3	77.9	72.3	68.2	64.1	61.8	*81.4*
Score-Average	94.8	90.7	89.9	88.6	88.0	87.1	86.2	85.6	*89.1*
Score-Worst	94.8	91.8	91.4	90.4	90.0	89.3	89.0	88.8	*90.6*



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	94.8	94.7	94.7	94.7	94.7	94.7	94.7	94.6	*94.0*
Rule-Worst	94.8	92.4	91.8	90.9	90.1	89.7	89.3	89.2	*91.0*
Score-Average	94.8	94.3	94.1	94.0	94.0	94.0	93.7	93.8	*93.4*
Score-Worst	94.8	93.2	92.1	91.1	90.8	90.2	89.9	89.6	*91.6*



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	94.8	83.3	79.8	77.0	77.1	75.3	70.3	65.1	*80.0*
Rule-Worst	94.8	21.8	8.6	2.8	1.0	0.0	0.0	0.0	*13.5*
Score-Average	-	-	-	-	-	-	-	-	-
Score-Worst	-	-	-	-	-	-	-	-	-

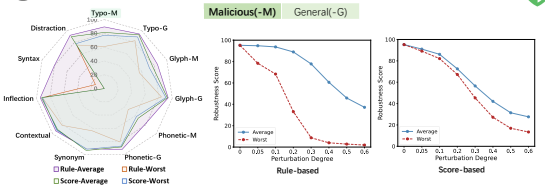


Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	94.8	94.8	93.8	90.0	83.8	78.9	76.0	74.1	*91.6*
Rule-Worst	94.8	91.3	77.5	51.5	31.7	20.6	14.3	11.2	*74.4*
Score-Average	94.8	92.5	89.5	86.4	80.2	76.4	74.3	73.5	*88.6*
Score-Worst	94.8	83.6	75.5	61.7	44.4	38.0	34.9	33.1	*73.2*

Figure 12: Robustness report for RoBERTa-base on AG's News.

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

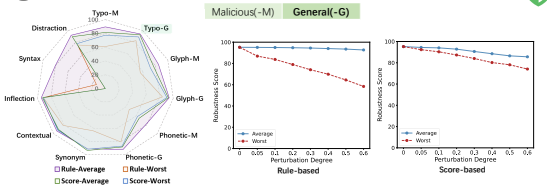
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.2	94.8	93.7	89.0	77.8	60.6	45.9	37.2	*89.7*
Rule-Worst	95.2	78.4	68.3	33.1	8.7	4.0	2.8	1.9	*61.1*
Score-Average	95.2	91.0	86.1	72.5	56.5	42.1	31.4	27.6	*81.6*
Score-Worst	95.2	89.1	82.1	67.2	45.3	27.2	16.9	13.4	*77.5*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

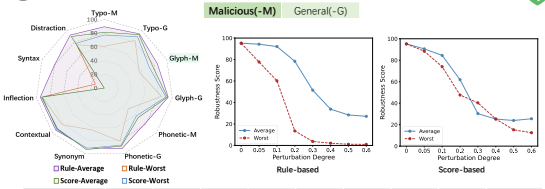
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.2	95.1	95.0	94.8	94.5	94.0	93.5	92.7	*94.2*
Rule-Worst	95.2	86.9	83.8	79.0	74.1	69.9	64.5	58.4	*82.6*
Score-Average	95.2	94.4	94.0	92.8	90.6	88.5	86.5	85.6	*92.7*
Score-Worst	95.2	92.3	90.3	87.3	84.0	80.2	78.1	74.1	*89.2*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

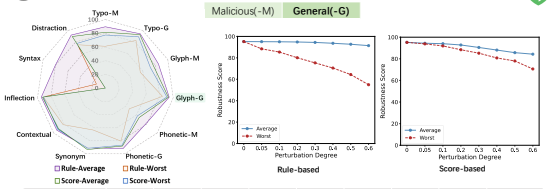
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.2	94.3	92.1	78.4	51.5	33.8	28.4	27.1	*84.9*
Rule-Worst	95.2	77.7	60.2	13.5	3.7	2.1	1.1	1.0	*55.9*
Score-Average	95.2	90.6	84.5	61.9	30.3	25.2	23.9	25.5	*77.4*
Score-Worst	95.2	88.4	74.0	47.6	40.3	25.3	15.2	12.5	*72.3*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

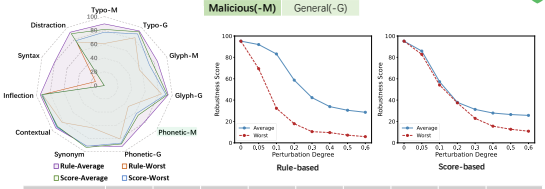
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.2	95.1	95.0	94.8	94.4	93.6	92.7	91.4	*94.1*
Rule-Worst	95.2	88.4	85.4	80.1	75.3	70.5	64.4	54.9	*83.9*
Score-Average	95.2	94.4	94.0	92.8	90.5	88.1	85.7	84.3	*92.7*
Score-Worst	95.2	93.8	91.9	88.4	85.2	80.8	78.0	70.6	*90.5*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

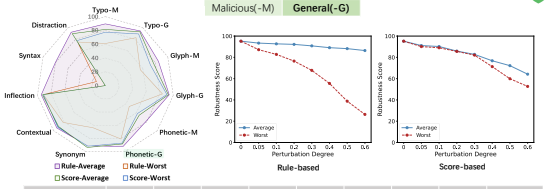
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.2	92.0	83.2	58.7	42.3	33.9	30.5	28.6	*78.5*
Rule-Worst	95.2	69.4	32.3	17.9	10.5	9.6	7.2	5.8	*46.1*
Score-Average	95.2	85.9	57.2	37.9	31.3	27.9	26.5	25.7	*65.4*
Score-Worst	95.2	82.8	54.1	37.4	22.9	15.6	12.7	10.9	*61.8*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

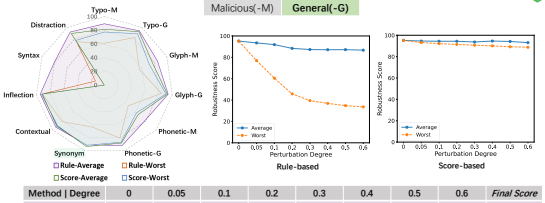
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.2	93.4	92.7	92.2	90.8	89.1	88.2	86.4	*91.9*
Rule-Worst	95.2	87.2	82.8	76.5	67.8	55.5	38.8	26.4	*80.6*
Score-Average	95.2	91.1	90.2	85.9	82.8	76.8	72.2	64.2	*88.0*
Score-Worst	95.2	90.1	89.0	85.6	82.1	71.3	59.9	52.7	*86.7*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

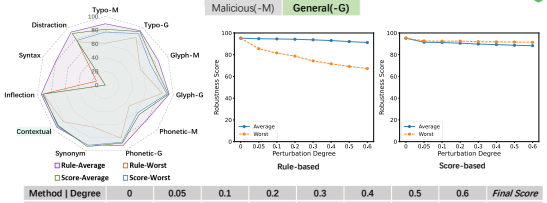
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.2	93.5	91.9	88.4	87.3	87.1	87.2	86.7	*91.0*
Rule-Worst	95.2	76.9	60.5	45.8	39.5	36.9	34.8	33.6	*63.7*
Score-Average	95.2	94.6	94.4	94.4	93.7	94.6	94.2	93.1	*93.7*
Score-Worst	95.2	93.2	92.2	91.4	90.7	90.1	89.2	88.8	*91.6*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

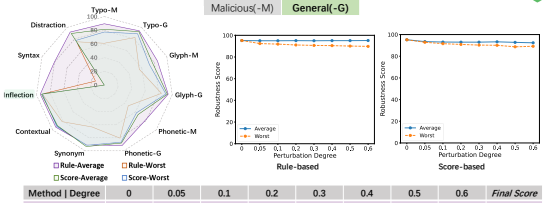
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.2	94.7	94.5	94.2	93.7	93.0	92.1	91.2	*93.7*
Rule-Worst	95.2	85.5	81.6	78.7	74.3	71.6	69.2	67.9	*81.5*
Score-Average	95.2	91.5	91.2	90.6	89.8	89.3	88.7	88.3	*90.4*
Score-Worst	95.2	92.7	92.4	92.5	92.0	91.8	91.7	91.4	*91.8*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

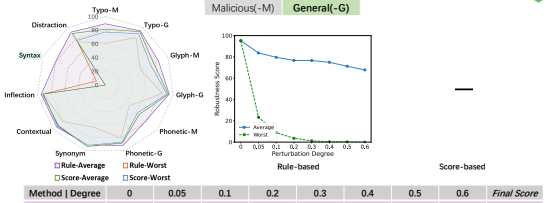
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.2	95.0	95.0	95.1	95.0	95.1	95.1	95.2	*94.3*
Rule-Worst	95.2	92.4	91.9	91.1	90.7	90.5	90.1	89.8	*91.2*
Score-Average	95.2	93.5	93.1	93.0	93.0	93.3	92.8	92.4	*92.6*
Score-Worst	95.2	92.9	91.6	90.9	90.3	90.1	88.7	89.2	*91.3*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

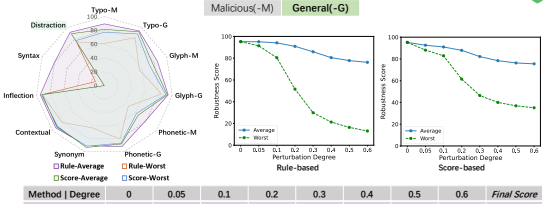
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.2	83.7	79.6	76.7	76.6	74.9	71.2	67.8	*80.1*
Rule-Worst	95.2	23.3	8.9	3.6	1.2	0.3	0.2	0.0	*14.4*
Score-Average	-	-	-	-	-	-	-	-	-
Score-Worst	-	-	-	-	-	-	-	-	-

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

Malicious(-M) | General(-G)

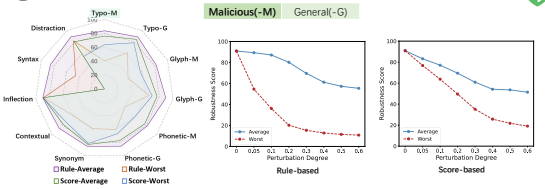


Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	95.2	95.1	94.1	90.9	85.9	80.5	77.8	76.2	*92.1*
Rule-Worst	95.2	91.4	80.4	51.6	29.9	21.3	16.4	13.1	*75.1*
Score-Average	95.2	92.6	91.0	87.9	82.2	78.4	76.3	75.5	*89.4*
Score-Worst	95.2	88.1	82.9	61.5	46.4	40.0	36.9	35.1	*77.5*

Figure 13: Robustness report for RoBERTa-large on AG's News.

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

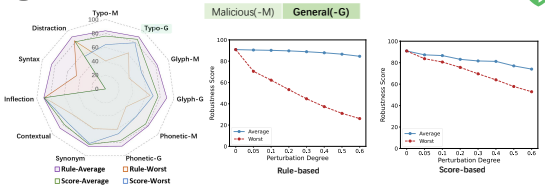
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	90.9	89.3	87.1	80.2	69.6	61.2	57.3	55.4	*84.0*
Rule-Worst	90.9	54.6	36.1	20.2	15.4	12.8	11.5	10.8	*40.5*
Score-Average	90.9	83.2	77.1	69.5	60.9	54.2	53.6	51.4	*76.3*
Score-Worst	90.9	76.8	63.8	49.6	35.2	25.7	21.8	19.1	*64.0*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

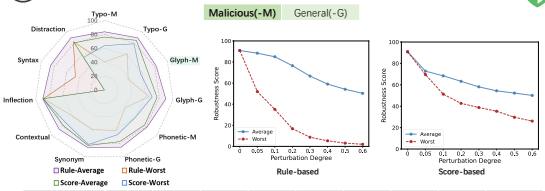
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	90.9	90.5	90.2	89.7	88.9	87.9	86.6	84.6	*89.3*
Rule-Worst	90.9	70.5	62.2	53.3	44.7	37.3	31.0	26.1	*62.1*
Score-Average	90.9	87.3	86.6	83.1	81.6	81.2	77.0	74.0	*85.1*
Score-Worst	90.9	83.7	80.6	75.6	69.6	64.0	57.8	52.9	*79.1*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

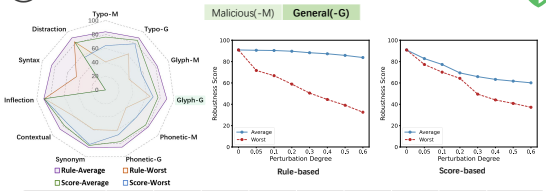
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	90.9	88.4	85.0	76.5	66.7	59.1	54.2	50.3	*82.3*
Rule-Worst	90.9	52.0	35.1	16.8	8.7	5.3	3.1	2.0	*37.6*
Score-Average	90.9	72.8	68.4	63.3	58.1	54.4	52.3	50.1	*68.0*
Score-Worst	90.9	69.6	51.3	42.7	38.8	35.3	29.7	26.1	*57.2*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

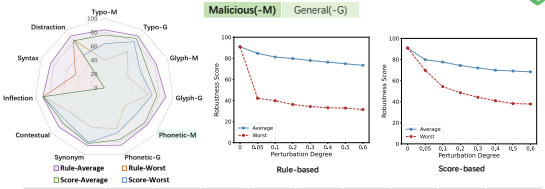
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	90.9	90.6	90.4	89.7	88.3	87.3	85.8	83.8	*89.4*
Rule-Worst	90.9	71.7	66.8	58.9	50.5	44.5	39.1	32.6	*65.3*
Score-Average	90.9	82.8	77.3	69.4	65.9	63.3	61.7	60.1	*76.9*
Score-Worst	90.9	77.3	70.1	64.3	49.5	44.1	40.8	37.2	*69.6*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

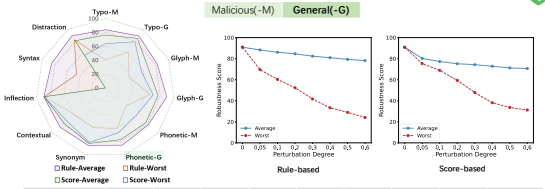
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	90.9	84.7	81.2	79.8	77.9	76.4	74.8	73.4	*81.6*
Rule-Worst	90.9	42.1	39.8	36.2	34.3	33.1	32.8	31.4	*39.5*
Score-Average	90.9	79.8	77.6	74.3	71.9	69.9	69.1	68.3	*76.9*
Score-Worst	90.9	69.7	54.3	48.5	44.2	40.9	38.2	37.8	*59.4*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

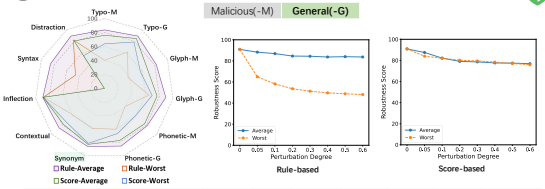
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	90.9	88.3	86.1	84.6	82.4	80.9	79.3	78.2	*85.8*
Rule-Worst	90.9	69.6	60.3	52.3	41.7	33.3	29.0	24.1	*60.7*
Score-Average	90.9	80.2	77.3	75.2	74.3	72.8	71.2	70.6	*77.4*
Score-Worst	90.9	75.3	68.9	59.4	47.8	38.1	33.6	31.2	*67.2*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

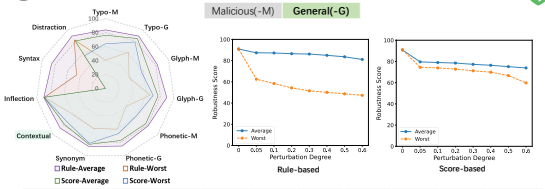
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	90.9	88.3	86.9	84.6	84.4	83.7	84.0	83.7	*86.3*
Rule-Worst	90.9	64.9	58.2	53.6	51.3	49.7	48.8	48.1	*59.6*
Score-Average	90.9	87.5	82.2	79.1	78.6	77.6	77.4	76.9	*83.3*
Score-Worst	90.9	84.0	82.0	80.2	79.5	78.4	77.1	75.9	*81.7*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

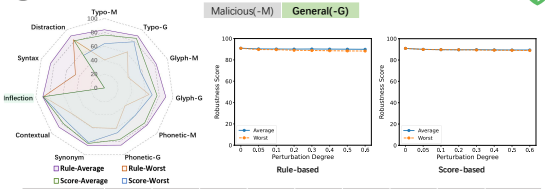
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	90.9	87.4	87.2	86.5	86.2	85.0	83.6	81.1	*86.3*
Rule-Worst	90.9	62.5	58.4	54.3	51.5	50.1	48.7	47.3	*58.6*
Score-Average	90.9	79.6	79.0	78.5	77.3	76.4	75.1	73.9	*78.3*
Score-Worst	90.9	74.5	73.9	72.8	71.2	69.9	66.7	59.8	*73.0*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

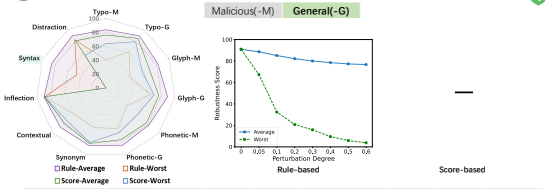
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	90.9	90.3	90.2	90.1	90.2	90.1	90.0	89.9	*89.5*
Rule-Worst	90.9	89.7	89.6	89.0	88.8	88.6	88.4	88.3	*88.8*
Score-Average	90.9	90.0	89.7	89.6	89.7	89.5	89.4	89.4	*89.1*
Score-Worst	90.9	89.9	89.7	89.6	89.4	89.2	89.1	88.9	*89.0*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

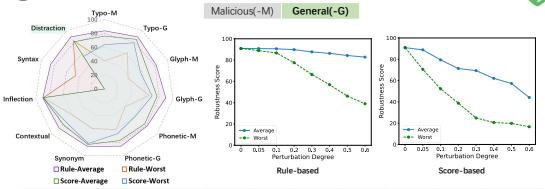
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	90.9	88.6	85.1	82.2	80.0	78.5	77.3	76.7	*85.1*
Rule-Worst	90.9	67.3	32.4	20.8	15.9	9.6	5.9	3.8	*45.8*
Score-Average	-	-	-	-	-	-	-	-	-
Score-Worst	-	-	-	-	-	-	-	-	-

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

Malicious(-M) | General(-G)

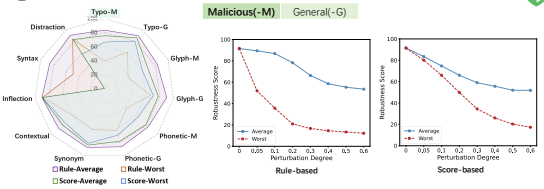


Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	90.9	90.8	90.6	89.8	87.7	86.3	84.3	82.8	*89.4*
Rule-Worst	90.9	88.9	86.7	77.5	66.5	56.9	46.2	39.0	*82.8*
Score-Average	90.9	88.8	79.4	71.3	69.3	62.1	57.2	44.1	*80.7*
Score-Worst	90.9	70.5	52.3	38.8	24.9	20.6	19.8	16.6	*55.8*

Figure 14: Robustness report for RoBERTa-base on Jigsaw.

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

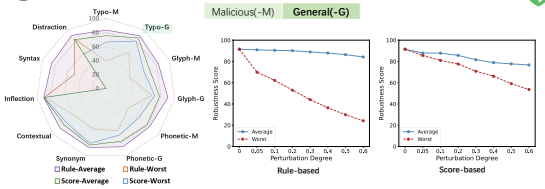
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	91.5	89.4	86.8	78.2	66.3	58.6	55.3	53.5	*83.4*
Rule-Worst	91.5	51.9	35.7	20.9	16.7	14.6	13.4	12.2	*39.3*
Score-Average	91.5	83.5	74.7	66.0	59.0	55.7	52.0	51.8	*75.3*
Score-Worst	91.5	80.1	66.0	49.9	34.5	26.0	20.3	17.3	*66.2*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

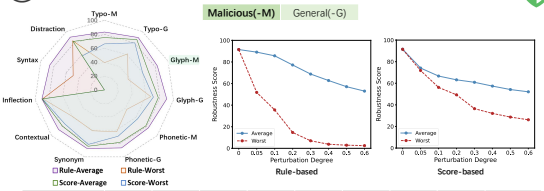
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	91.5	90.9	90.5	90.1	88.9	87.9	86.4	84.2	*89.6*
Rule-Worst	91.5	69.9	62.2	53.0	44.1	36.4	29.9	24.2	*61.7*
Score-Average	91.5	87.9	87.8	85.7	81.6	78.9	77.7	76.7	*86.0*
Score-Worst	91.5	85.6	81.0	77.6	70.8	66.1	59.1	53.6	*80.6*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

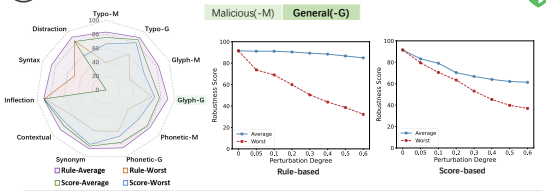
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	91.5	89.1	85.7	77.2	68.8	62.8	57.1	53.0	*83.2*
Rule-Worst	91.5	51.8	35.6	14.7	7.0	3.8	2.9	2.5	*37.3*
Score-Average	91.5	74.1	66.7	63.2	60.9	57.4	54.2	52.1	*68.5*
Score-Worst	91.5	71.8	56.2	49.3	36.5	32.1	28.7	26.2	*60.1*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

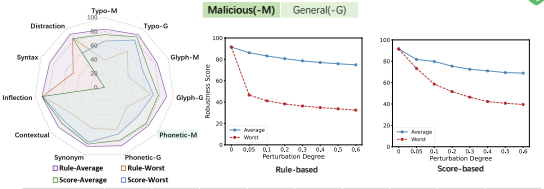
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	91.5	91.1	91.1	90.4	89.3	88.4	86.7	84.9	*90.0*
Rule-Worst	91.5	73.7	69.0	59.9	50.4	43.7	38.6	32.2	*67.0*
Score-Average	91.5	83.2	79.1	70.3	66.8	64.0	62.1	61.3	*77.8*
Score-Worst	91.5	79.5	70.5	63.4	53.1	45.3	39.8	36.9	*70.9*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

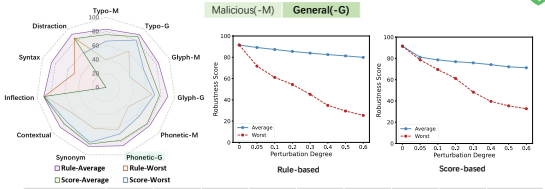
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	91.5	86.2	83.2	80.7	78.6	77.1	75.9	74.9	*83.1*
Rule-Worst	91.5	46.6	41.2	38.3	36.4	34.9	33.7	32.5	*42.5*
Score-Average	91.5	81.7	79.8	75.4	72.5	71.0	69.4	68.9	*78.6*
Score-Worst	91.5	73.3	58.6	51.6	46.4	42.3	40.7	39.5	*62.9*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

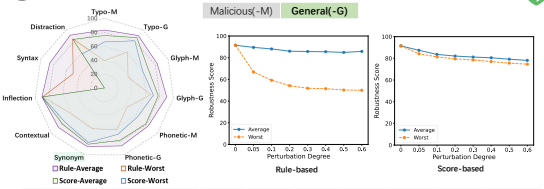
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	91.5	89.2	87.3	85.5	83.9	82.5	81.3	79.9	*86.8*
Rule-Worst	91.5	71.7	61.1	54.4	45.2	34.8	29.5	25.4	*62.5*
Score-Average	91.5	81.1	78.6	76.9	75.8	74.1	72.1	71.2	*78.5*
Score-Worst	91.5	78.6	69.5	61.2	48.3	39.6	35.4	32.7	*69.4*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

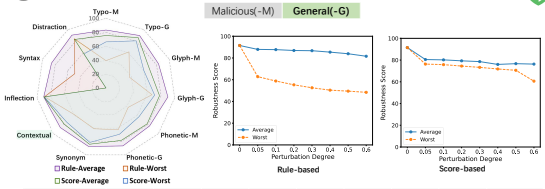
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	91.5	89.5	88.1	86.0	85.7	85.5	84.9	85.8	*87.6*
Rule-Worst	91.5	66.8	59.2	54.2	51.8	51.4	50.2	50.0	*61.0*
Score-Average	91.5	87.4	83.6	82.1	81.2	80.5	79.2	78.1	*84.3*
Score-Worst	91.5	84.2	81.3	79.5	78.5	77.0	75.6	74.6	*81.4*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

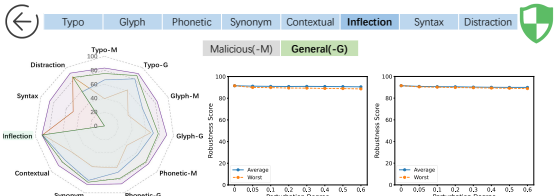
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	91.5	87.9	87.7	86.9	86.7	85.4	83.9	81.6	*86.8*
Rule-Worst	91.5	62.8	58.8	55.3	52.6	50.4	49.5	48.4	*59.0*
Score-Average	91.5	80.5	80.2	79.3	78.6	76.0	76.7	76.3	*79.3*
Score-Worst	91.5	76.3	75.8	74.5	73.3	71.8	70.6	60.6	*74.8*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

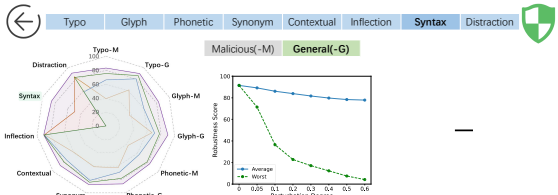
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	91.5	91.1	90.9	90.8	90.9	90.8	90.7	90.5	*90.3*
Rule-Worst	91.5	89.9	89.8	89.4	89.3	89.0	88.9	88.6	*89.0*
Score-Average	91.5	90.8	90.6	90.5	90.2	90.0	89.9	89.9	*89.9*
Score-Worst	91.5	90.4	90.1	89.8	89.6	89.3	89.1	88.8	*89.4*

Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

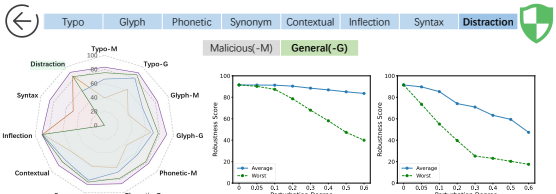
Malicious(-M) | General(-G)



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	91.5	89.2	86.1	83.9	81.7	79.8	78.4	77.9	*86.0*
Rule-Worst	91.5	71.4	36.5	22.8	17.1	12.3	7.5	4.2	*49.3*
Score-Average	-	-	-	-	-	-	-	-	-
Score-Worst	-	-	-	-	-	-	-	-	-

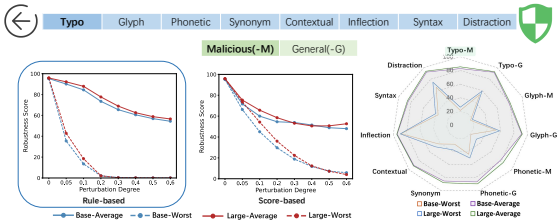
Typo | Glyph | Phonetic | Synonym | Contextual | Inflection | Syntax | Distraction

Malicious(-M) | General(-G)

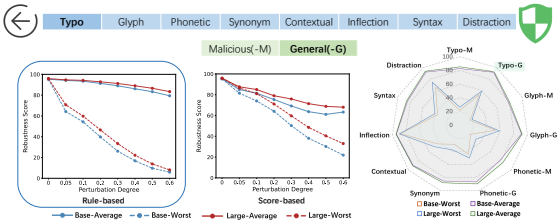


Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Rule-Average	91.5	91.4	91.3	90.4	88.5	86.9	85.1	83.6	*90.1*
Rule-Worst	91.5	90.2	87.4	78.6	68.0	58.1	47.3	40.0	*83.9*
Score-Average	91.5	89.7	85.3	74.1	70.9	63.2	59.4	47.3	*83.1*
Score-Worst	91.5	73.5	55.1	39.7	25.2	23.1	20.3	17.5	*58.2*

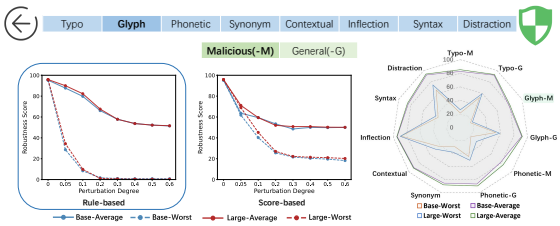
Figure 15: Robustness report for RoBERTa-large on Jigsaw.



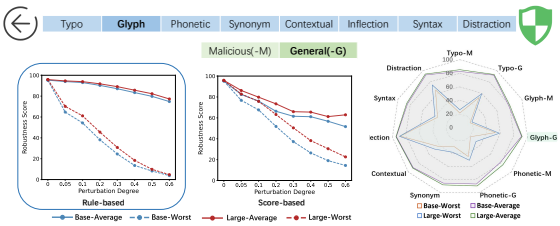
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	90.1	84.5	73.5	65.5	60.6	56.9	54.4	*82.7*
Base-Worst	95.3	35.5	13.5	2.1	0.4	0.2	0.1	0.1	*21.4*
Large-Average	95.9	92.2	87.9	77.7	68.9	62.6	58.8	56.6	*85.4*
Large-Worst	95.9	42.9	18.5	2.1	0.2	0.1	0.0	0.0	*26.4*



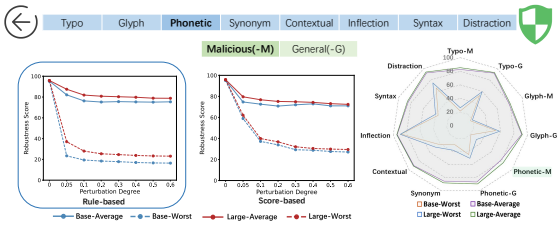
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	94.2	93.4	91.3	89.0	86.1	83.3	79.4	*92.0*
Base-Worst	95.3	64.2	54.3	39.8	26.1	16.8	9.7	5.9	*53.0*
Large-Average	95.9	94.7	94.2	92.8	91.2	89.0	86.6	83.4	*93.0*
Large-Worst	95.9	70.7	59.8	46.5	33.4	22.2	13.8	8.1	*59.2*



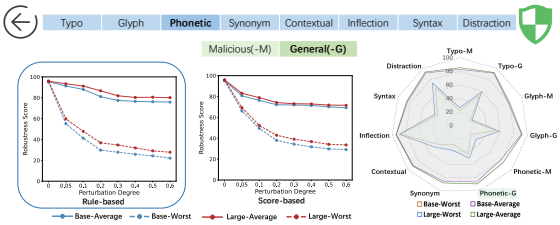
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	87.7	79.8	66.1	57.8	53.7	52.2	51.4	*78.6*
Base-Worst	95.3	28.6	8.4	1.7	1.0	0.8	0.8	0.8	*16.7*
Large-Average	95.9	90.0	82.3	67.6	57.8	53.7	52.2	51.6	*80.5*
Large-Worst	95.9	34.4	9.8	1.1	0.6	0.4	0.3	0.3	*19.8*



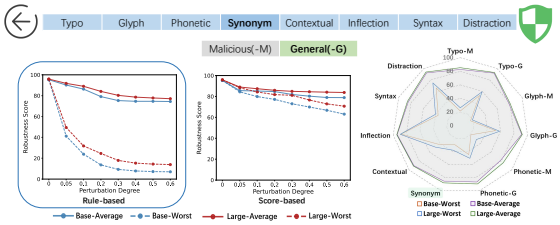
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	94.0	93.0	90.3	87.1	83.4	79.8	74.8	*91.4*
Base-Worst	95.3	64.7	54.3	38.1	24.5	13.5	8.2	3.8	*52.8*
Large-Average	95.9	94.6	93.9	91.8	89.1	85.7	82.3	77.3	*92.4*
Large-Worst	95.9	70.2	61.1	45.5	30.8	18.4	9.8	4.7	*58.8*



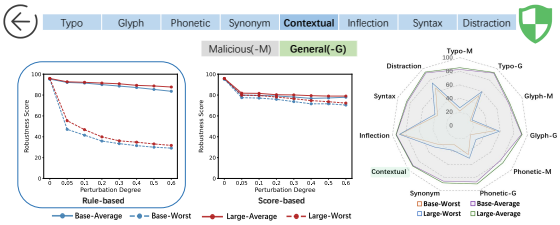
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	82.1	76.3	75.2	75.6	75.3	75.1	75.4	*78.4*
Base-Worst	95.3	23.4	19.4	18.4	17.8	17.0	16.6	16.4	*20.9*
Large-Average	95.9	87.5	81.9	80.8	80.3	79.8	78.9	78.8	*83.7*
Large-Worst	95.9	37.0	28.0	25.4	24.6	23.8	23.3	23.1	*31.5*



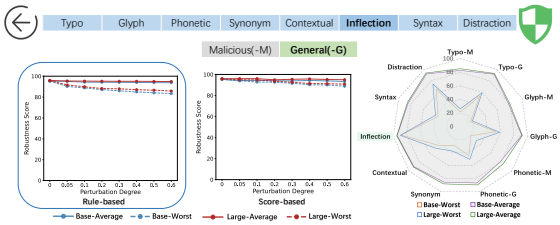
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	91.3	88.0	81.1	77.4	76.5	76.2	75.8	*86.8*
Base-Worst	95.3	55.1	41.2	29.8	27.8	25.8	24.3	22.1	*44.7*
Large-Average	95.9	93.4	91.2	86.7	81.9	80.3	80.5	80.1	*89.8*
Large-Worst	95.9	59.7	47.8	36.8	34.7	31.9	29.1	27.8	*50.2*



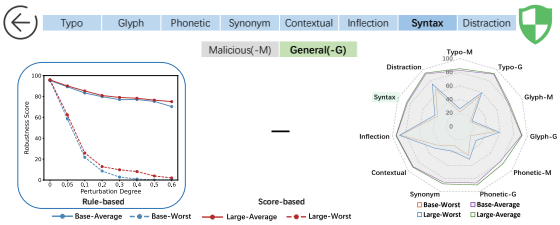
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	90.2	86.3	79.1	75.3	74.6	74.6	74.4	*85.3*
Base-Worst	95.3	41.2	23.9	13.6	9.3	7.8	7.2	7.0	*29.3*
Large-Average	95.9	91.8	89.0	84.1	80.3	78.6	77.8	77.1	*88.0*
Large-Worst	95.9	49.5	31.8	24.6	17.9	15.3	14.4	13.9	*37.7*



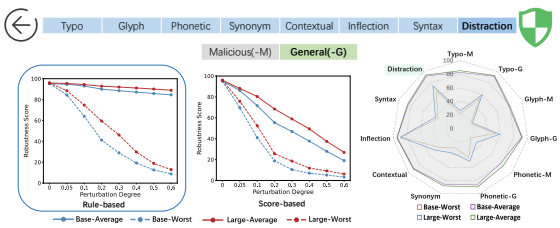
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	92.1	91.5	90.0	88.6	87.2	85.4	83.6	*90.4*
Base-Worst	95.3	47.1	41.5	35.9	33.4	31.5	30.0	29.1	*42.2*
Large-Average	95.9	92.7	92.2	91.5	90.8	89.3	88.7	87.7	*91.4*
Large-Worst	95.9	55.4	46.8	39.9	36.1	34.8	33.1	31.7	*48.5*



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	95.1	94.3	94.2	94.1	94.0	93.9	93.8	*93.9*
Base-Worst	95.3	90.2	88.8	87.1	86.0	84.9	84.1	83.4	*88.2*
Large-Average	95.9	95.4	95.3	95.2	95.1	95.1	95.0	94.9	*94.6*
Large-Worst	95.9	91.6	89.9	88.2	87.9	87.1	86.6	85.8	*89.5*

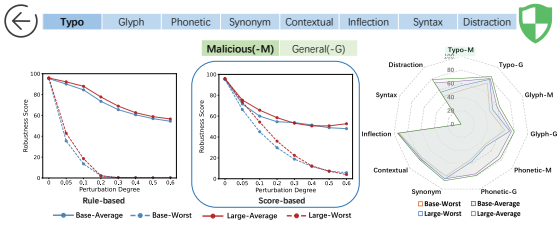


Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	89.3	83.3	79.8	77.0	77.1	75.3	70.3	*84.4*
Base-Worst	95.3	58.6	21.8	8.6	2.8	1.0	0.0	0.0	*36.0*
Large-Average	95.9	90.1	85.3	80.9	79.1	78.3	76.5	75.1	*85.7*
Large-Worst	95.9	62.4	25.9	12.8	9.8	8.1	3.9	2.0	*40.2*

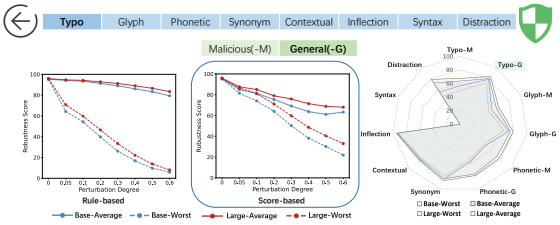


Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	94.7	93.0	90.0	88.5	87.1	85.8	84.5	*92.1*
Base-Worst	95.3	84.4	64.0	41.2	29.0	19.3	12.6	8.8	*66.0*
Large-Average	95.9	95.4	94.4	92.8	92.0	91.1	90.1	88.9	*93.6*
Large-Worst	95.9	88.5	74.7	59.5	46.1	29.8	18.9	12.9	*74.6*

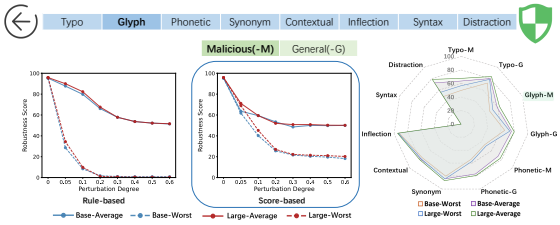
Figure 16: Robustness comparison report for rule-based evaluation on SST-2.



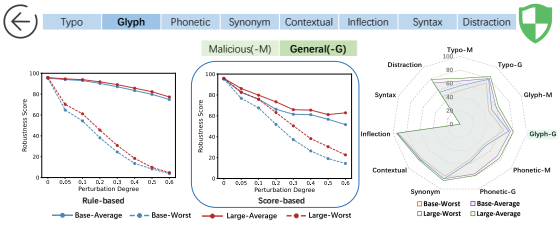
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	71.7	60.0	54.6	53.9	51.5	48.9	48.0	*63.8*
Base-Worst	95.3	66.3	45.0	29.7	18.8	11.8	7.3	5.8	*49.8*
Large-Average	95.9	75.4	65.7	58.5	53.2	50.6	50.7	52.7	*67.5*
Large-Worst	95.9	73.6	54.3	36.0	22.4	12.3	7.2	4.0	*56.8*



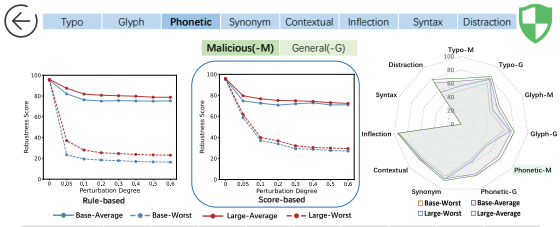
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	85.0	81.4	75.4	69.2	63.7	61.1	63.3	*80.0*
Base-Worst	95.3	81.5	74.0	64.1	50.3	38.1	30.2	21.9	*72.2*
Large-Average	95.9	87.6	85.0	79.0	75.9	71.4	68.8	68.0	*83.5*
Large-Worst	95.9	86.5	81.0	71.1	59.7	48.6	40.4	33.0	*78.5*



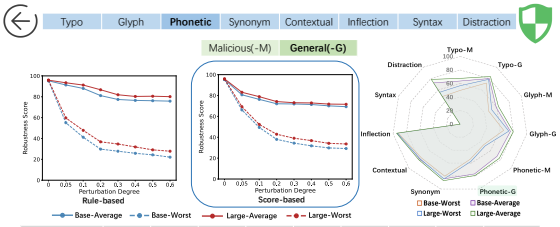
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	63.6	59.4	53.4	48.5	50.0	49.8	50.0	*59.1*
Base-Worst	95.3	61.8	40.2	25.6	21.6	20.4	19.6	18.2	*46.6*
Large-Average	95.9	71.1	59.4	52.0	50.8	50.8	50.2	50.1	*62.8*
Large-Worst	95.9	69.0	45.2	27.0	22.2	21.4	21.0	20.2	*51.7*



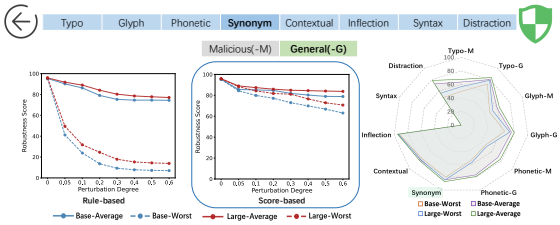
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	82.4	76.0	66.3	61.5	61.1	56.7	51.6	*75.5*
Base-Worst	95.3	76.8	67.6	51.8	37.2	26.4	19.0	14.4	*65.3*
Large-Average	95.9	86.2	79.8	73.4	65.9	65.5	61.2	62.9	*79.8*
Large-Worst	95.9	82.8	75.8	63.2	50.4	38.2	30.4	22.6	*73.2*



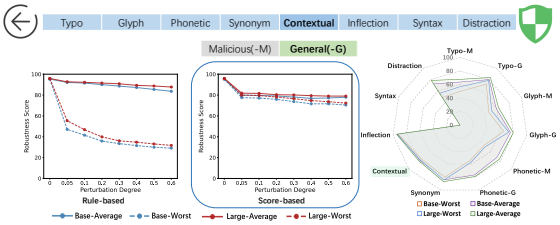
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	74.7	72.6	70.8	72.0	72.9	71.0	71.0	*72.8*
Base-Worst	95.3	58.9	37.2	33.9	29.3	28.8	27.6	27.1	*46.4*
Large-Average	95.9	79.6	76.8	75.2	74.9	74.3	73.1	72.4	*77.1*
Large-Worst	95.9	62.1	39.9	36.8	32.1	30.5	29.9	29.5	*49.3*



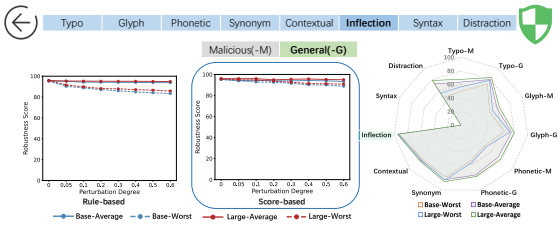
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	80.7	76.3	72.1	71.9	71.4	70.1	69.3	*76.8*
Base-Worst	95.3	66.2	49.4	38.0	34.4	31.8	29.8	29.2	*54.0*
Large-Average	95.9	83.1	78.9	74.2	73.1	72.9	71.8	71.6	*79.1*
Large-Worst	95.9	69.3	52.3	42.9	39.1	36.8	34.2	33.6	*57.5*



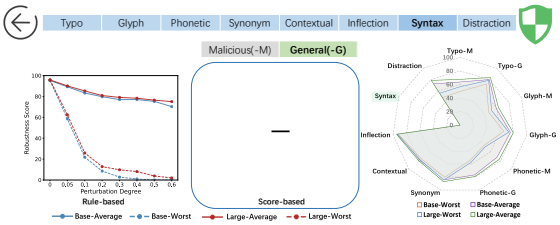
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	85.6	85.2	84.7	81.9	80.3	79.0	78.9	*84.2*
Base-Worst	95.3	84.4	79.9	77.2	73.0	69.9	66.8	63.1	*80.1*
Large-Average	95.9	89.0	87.4	85.9	84.9	84.5	84.1	83.8	*87.0*
Large-Worst	95.9	88.3	84.3	81.8	81.1	76.6	72.9	70.7	*84.6*



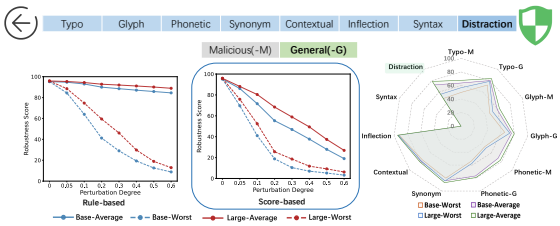
Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	79.9	79.6	79.1	78.1	76.8	77.2	77.9	*78.8*
Base-Worst	95.3	77.5	77.1	75.9	73.7	71.6	71.6	70.4	*76.0*
Large-Average	95.9	81.8	81.6	80.3	80.1	79.4	79.0	79.0	*80.7*
Large-Worst	95.9	79.8	79.6	78.1	76.5	74.7	73.6	72.3	*78.4*



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	94.2	94.4	94.0	93.9	94.3	93.9	93.3	*93.5*
Base-Worst	95.3	94.0	92.8	92.6	91.6	90.2	90.0	88.8	*92.4*
Large-Average	95.9	96.0	96.0	94.9	95.4	95.7	95.2	95.0	*95.0*
Large-Worst	95.9	95.0	94.8	93.6	92.6	91.4	91.2	90.8	*93.7*



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	-	-	-	-	-	-	-	-	-
Base-Worst	-	-	-	-	-	-	-	-	-
Large-Average	-	-	-	-	-	-	-	-	-
Large-Worst	-	-	-	-	-	-	-	-	-



Method | Degree	0	0.05	0.1	0.2	0.3	0.4	0.5	0.6	*Final Score*
Base-Average	95.3	86.1	71.6	55.3	46.9	37.7	27.9	19.0	*72.6*
Base-Worst	95.3	69.6	41.0	18.8	10.4	7.0	5.2	3.2	*48.4*
Large-Average	95.9	88.1	80.4	68.4	58.9	49.4	37.4	26.9	*78.7*
Large-Worst	95.9	75.8	52.4	25.6	18.6	11.8	9.2	6.2	*55.9*

Figure 17: Robustness comparison report for score-based evaluation on SST-2.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
the final section

☐ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

☐ B1. Did you cite the creators of artifacts you used?
No response.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
No response.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
No response.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.

☐ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
No response.

C ☑ Did you run computational experiments?

r

☐ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Not applicable. Left blank.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
w

☐ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Not applicable. Left blank.

☐ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Not applicable. Left blank.

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?
4

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Not applicable. Left blank.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Not applicable. Left blank.

☑ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Appendix

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.