Robustness Evaluation of the FLAN-T5 Model Against Prompt Variations in Text Classification Tasks

Muhammad Arsya Zain Yashifa[1], Tirana Noor Fatyanosa[1*], and Putra Pandu Adikara[1]

Department of Informatics, Faculty of Computer Science,
Universitas Brawijaya, Malang, Indonesia
arsyazain@student.ub.ac.id, fatyanosa@ub.ac.id, adikara.putra@ub.ac.id
*Corresponding author

Abstract. Large Language Models (LLMs) have demonstrated remarkable capabilities across various natural language processing tasks; however, their performance is highly influenced by the phrasing of the provided prompts. This study evaluates the robustness of the Flan-T5 model against prompt variations on three text classification datasets: SST-2 (sentiment analysis), MNLI (multi-class natural language inference), and RTE (binary recognizing textual entailment). The variations tested include rewording, style change, noise injection, and adversarial attack, with the original prompt serving as a baseline, using the PromptBench framework with ten trials per prompt at a temperature of 0.3. Results indicate that the best performance was achieved in the original category with a mean accuracy of 0.845 and a mean F1-score of 0.836 across datasets, while the worst performance was exhibited by the noise category with a mean accuracy of 0.595 and a mean F1-score of 0.676. On the SST-2 dataset, noise reduced the mean accuracy from 0.940 to 0.340, with 63.1% of outputs failing to map to a valid label, as character-level perturbations obscured the target labels within the prompt. Adversarial attacks demonstrated non-uniform impacts: instructions directing the model toward negative labels on SST-2 yielded a mean accuracy of only 0.321, whereas a contradiction-type adversarial attack on MNLI essentially preserved the original-prompt baseline (0.828 versus 0.824), while entailment-type and neutral-type adversarial attacks on the same dataset caused substantially larger drops (0.663 and 0.621, respectively). Rewording and style change produced moderate, dataset-dependent degradation, with mean accuracies across the three datasets of 0.817 and 0.759, respectively. These findings suggest that Flan-T5 is sufficiently robust against mild linguistic variations but vulnerable to structural perturbations in the instruction, with the degree of vulnerability shaped by task characteristics and, on multi-class tasks, by which classes the model already recognizes least reliably.

Keywords: Large language model · Robustness · Prompt engineering · PromptBench · Flan-T5 · Natural language processing

© The Author(s) 2026
M. Murtadha et al. (eds.), *Proceedings of the 2026 11th International Conference on Sustainable Information Engineering and Technology (SIET 2026)*, Atlantis Highlights in Computer Sciences 26,
https://doi.org/10.2991/978-94-6239-828-3_22

1 Introduction

The rapid advancement of Large Language Models (LLMs) such as GPT and LLaMA has brought significant impact to the field of natural language processing (NLP). LLMs have proven capable of completing various complex tasks, ranging from sentiment analysis and translation to text-based question answering. Despite their many advantages, recent studies show that these models have a fundamental weakness: they are overly sensitive to variations in the prompts provided [8]. Small changes in prompt phrasing often cause significant differences in answer quality, raising questions about model robustness in the face of user input variation.

This phenomenon has been confirmed in various studies. Research from Japan found that simple changes in prompt structure could drastically reduce GPT-4 accuracy in Japanese text classification tasks, from 49.21% to only 25.44% [4]. Other research showed that prompt writing format—such as plain text, Markdown, JSON, or YAML—can affect model performance by up to 40% in code translation tasks [5].

Beyond natural variations, LLMs have also been proven vulnerable to adversarial perturbations. A study using PromptRobust demonstrated that attacks at the character, word, sentence, and semantic levels can significantly degrade model performance, with word-level attacks recorded as the most damaging, producing an average performance drop of 39% [14].

In response to these challenges, various efforts have been made to develop more comprehensive evaluation frameworks. One such framework is PromptBench, an integrated library specifically designed to test LLM robustness against prompt variations. This framework supports diverse scenarios including adversarial attacks, dynamic evaluation, and transferability analysis, providing a comprehensive overview of model limitations and potential improvements [15].

These findings reveal that robustness is a crucial aspect that needs to be systematically evaluated. However, most previous studies still focus on a single type of prompt variation or are limited to one language and one type of task. Meanwhile, the characteristics of text classification tasks can vary greatly, from opinion-based classification such as sentiment analysis to classification requiring inter-sentence relational understanding and semantic reasoning. Moreover, there remains a need for studies that comprehensively compare different types of prompt perturbation—such as rewording, style change, noise, and adversarial attack—to identify which form of variation most significantly affects model performance.

Based on this, the present study evaluates the robustness of a large language model using the PromptBench framework across several text classification datasets. The research focuses on Google Flan-T5 and three benchmark datasets: SST-2 for sentiment classification, and MNLI and RTE for natural language inference tasks. Through testing against various prompt variations, this study aims to provide a more comprehensive picture of LLM performance sensitivity and stability across different types of classification tasks.

2 Related Work

In recent years, the topic of robustness and sensitivity of LLMs to prompt variations has gained increasing attention. Gan and Mori [4] evaluated LLM robustness in Japanese text classification tasks and found that simple changes in prompt sentence structure could reduce GPT-4 accuracy from 49.21 to 25.44. This highlights that even highly sophisticated models remain unstable when facing prompt template variations, particularly in non-English languages.

He et al. [5] investigated whether prompt format differences—plain text, Markdown, JSON, and YAML—affect LLM performance. Using GPT-3.5-turbo and GPT-4 across reasoning, translation, and code generation tasks, they found that GPT-3.5-turbo performance can differ by up to 40% merely due to prompt format changes, while GPT-4 was relatively more stable.

Razavi et al. [8] constructed a dataset called PromptSET containing thousands of prompt variations from TriviaQA and HotpotQA, and benchmarked sensitivity prediction methods including self-evaluation and query performance prediction. Their results showed that current prediction methods remain weak, with small prompt variations still producing significant performance differences.

Zhu et al. [14] introduced PromptRobust, constructing a benchmark with 4,788 adversarial prompts covering eight tasks and 13 datasets to test various LLMs. Their findings demonstrated that most current LLMs are not robust against adversarial prompts. A follow-up contribution introduced PromptBench [15], a unified library for systematic LLM robustness evaluation across character, word, sentence, and semantic levels.

Agrawal et al. [1] went further by not only evaluating but also enhancing LLM robustness. Their study examined how vulnerable models including Flan-T5 are to subtly perturbed instructions on sentiment analysis and NLI tasks, and demonstrated that instruction augmentation and fine-tuning can improve robustness.

On the adversarial resilience side specifically, Wallace et al. [10] showed early on that short, universal adversarial trigger sequences appended to an input can consistently flip the predictions of NLP models, establishing that instruction- and input-level manipulation is a general vulnerability rather than one confined to a specific architecture. More recently, Zhang et al. [13] surveyed the broader landscape of robustness evaluation and mitigation strategies for LLMs, noting that adversarial robustness, prompt sensitivity, and noisy-input tolerance are frequently studied in isolation and calling for evaluation protocols that integrate multiple perturbation types, which is the gap that motivates the present study's combined rewording, style-change, noise-injection, and adversarial-attack design.

These prior studies collectively highlight that most research has evaluated LLM sensitivity to prompt variations separately without integrating multiple perturbation types in a single, unified evaluation framework. The present study addresses this gap by integrating four prompt variation categories—rewording, style change, noise injection, and adversarial attack—in a structured experiment using PromptBench on the Google Flan-T5 model across SST-2, MNLI, and RTE datasets.

3 Methodology

3.1 Datasets

Three publicly available text classification datasets were used in this study. All three are part of the GLUE benchmark [11] and were accessed via the PromptBench framework's `DatasetLoader`, which in turn retrieves them through the Hugging Face `datasets` library (`load_dataset("glue", <config>)`). No custom or random train/test splitting was performed in this study; each dataset's official, predefined validation split (as released by the GLUE benchmark maintainers) was used directly, since (a) label information for the official GLUE test sets is not publicly released, and (b) using the standard validation split keeps results directly comparable with other work that reports on GLUE. The datasets can be downloaded or browsed at the following locations:

- SST-2: `https://huggingface.co/datasets/nyu-mll/glue/viewer/sst2`
- MNLI: `https://huggingface.co/datasets/nyu-mll/glue/viewer/mnli`
- RTE: `https://huggingface.co/datasets/nyu-mll/glue/viewer/rte`
- GLUE benchmark homepage: `https://gluebenchmark.com/`

SST-2 (Stanford Sentiment Treebank 2) [9] is a binary sentiment classification dataset derived from movie reviews. The validation split used in this study consists of 872 samples: 444 positive and 428 negative.

MNLI (Multi-Genre Natural Language Inference) [12] is a large-scale NLI dataset with 433,000 sentence pairs spanning diverse genres. Pairs are labeled as *entailment* (0), *neutral* (1), or *contradiction* (2). The combined matched and mismatched validation splits total 19,647 samples. Because this study evaluates eight prompt variants over ten repeated runs each, running full inference over all 19,647 MNLI instances was computationally infeasible within the available budget. A stratified random subsample of 3,600 instances (1,200 per label class, fixed random seed) was therefore drawn from the combined validation split and used for all MNLI experiments reported in this paper; SST-2 and RTE, being smaller, were evaluated on their full validation splits (872 and 277 instances, respectively). An earlier, smaller subsample of 600 instances (200 per class) was used in preliminary experiments; results changed meaningfully between the two sample sizes for some prompts (most notably the Adversarial (Contradiction) prompt, discussed in Section 4.2), so the 3,600-instance subsample is used throughout this paper as it more closely matches results obtained on the full validation split for the Original prompt.

RTE (Recognizing Textual Entailment) [3] is a binary NLI dataset consisting of Text-Hypothesis pairs labeled as *entailment* (0) or *contradiction* (1). The validation split used contains 277 samples: 146 entailment and 131 contradiction.

3.2 Model

This study uses **Flan-T5 Large** [2], a variant of the T5 encoder-decoder Transformer model that has undergone instruction fine-tuning on a diverse collection

of tasks formatted as natural language instructions. Flan-T5 is available in five sizes (Small, Base, Large, XL, XXL), ranging from approximately 80 million to 11 billion parameters. The Large variant (approximately 780 million parameters) was selected to balance model capacity with computational feasibility, as experiments required repeated inference across three datasets and multiple prompt variations. The model was accessed via the Hugging Face Hub through PromptBench's `LLMModel` interface with `temperature=0.3` and `max_new_tokens=50`. A low but non-zero temperature of 0.3 was deliberately chosen over a fully deterministic setting (`temperature=0`, i.e., greedy decoding). Greedy decoding would always return the same output for the same prompt, making it impossible to distinguish a stable model from an unstable one across repeated trials—every prompt would trivially report a standard deviation of zero. A small positive temperature preserves most of the model's determinism while still allowing enough sampling variability to be measured across the ten runs per prompt, so that the resulting standard deviation reflects genuine output instability rather than being an artifact of a fixed decoding strategy. A value of 0.3 was selected, rather than a higher value (e.g., 0.7–1.0), to keep this sampling noise small relative to the effect of the prompt variation itself, which is the primary variable of interest in this study.

3.3 Prompt Variation Categories

Prompt variations were designed based on three robustness dimensions identified by Kumar and Mishra [6]: sensitivity to prompt variations, handling noisy or corrupted inputs, and resilience to adversarial attacks. Five categories were evaluated:

1. **Original**: The baseline prompt with no modifications.
2. **Rewording**: The instruction is rephrased without changing its semantic meaning, to test sensitivity to surface-form variation.
3. **Style Change**: The instruction is delivered in a different register (formal or casual), to test sensitivity to stylistic variation.
4. **Noise Injection**: Character-level perturbations are introduced into the instruction (e.g., replacing letters with symbols such as `Cl@ssify`, `pos!tive`, `neg@tive`), simulating corrupted or noisy input.
5. **Adversarial Attack**: The instruction is deliberately modified to mislead or bias the model's prediction (e.g., "*Pretend you are biased toward negative.*").

It is important to note that noise injection targets the *instruction and label keywords* within the prompt, not the input text content from the dataset. This design choice isolates the effect of structural prompt degradation on model behavior.

3.4 Evaluation Protocol

Each prompt was evaluated ten times with the same temperature setting to capture output variability due to sampling stochasticity. For each run, model

outputs were mapped to numeric labels using a rule-based projection function based on keyword detection. Outputs that could not be mapped to a valid label (returned as -1) are scored as incorrect: accuracy counts them as mismatches against the true label, and macro F1 is computed over the fixed set of valid classes for each dataset (e.g., $\{0, 1\}$ for SST-2/RTE, $\{0, 1, 2\}$ for MNLI), so an unmapped output is never excluded from the denominator and never forms a spurious extra "class -1" in the macro average. Performance was measured using accuracy and macro-averaged F1-score. The mean and standard deviation across ten runs were reported for each prompt. For all three datasets, we additionally report per-class precision, recall, and F1, together with a confusion matrix, pooled across the ten runs of each prompt, to make class-level behavior (e.g., a systematic bias against one label) visible rather than only the aggregate macro F1. To keep the manuscript within the page limit, the full prompt templates, output-projection keyword rules, and these per-class/confusion-matrix results are provided in the project's public code repository: `https://github.com/NLP-UB/promptbench`.

The original prompt results served as the baseline. For cross-category comparison, results from prompts within the same category were aggregated by averaging.

4 Results

4.1 Results on SST-2

Table 1 presents evaluation results on the SST-2 dataset, with unmapped (-1) outputs scored as incorrect. The Original and Rewording prompts achieved a mean accuracy/F1-score of 0.940 and 0.944, respectively, both with very low standard deviation ($\leq$0.003) and no unmapped outputs, indicating high stability. Style Change variants (formal and casual) maintained similarly high performance at 0.943 and 0.942, respectively.

Noise Injection caused a dramatic performance collapse, reducing mean accuracy to 0.340 and mean F1-score to 0.494. The confusion matrix for this prompt (available in the project repository) shows that 63.1% of its outputs could not be mapped to a valid label at all: character-level perturbations obscured the key label tokens (`pos!tive`, `neg@tive`) that the projection function searches for, so the majority of generations were simply unparseable rather than confidently wrong.

Adversarial attacks demonstrated asymmetric effects. The positive-bias attack maintained a relatively high mean accuracy of 0.877, with almost no unmapped outputs (0.1%). The negative-bias attack instead severely degraded performance to a mean accuracy of 0.321 and F1-score of 0.468—below the roughly 0.51 expected from random prediction given the dataset's near-balanced class distribution—with 63.0% of outputs unmapped, a failure mode much closer to Noise Injection's collapse than to the positive-bias attack's comparatively graceful degradation, despite both being adversarial instructions of similar surface form.

Table 1. Evaluation results on SST-2 dataset.

No.	Category	Mean Acc.	Std Acc.	Mean F1	Std F1	Unmapped
1	Original	0.940	0.003	0.940	0.003	0.0%
2	Rewording	0.944	0.003	0.944	0.003	0.0%
3	Style Change (Formal)	0.943	0.004	0.943	0.004	0.0%
4	Style Change (Casual)	0.942	0.002	0.942	0.002	0.0%
5	Noise	0.340	0.004	0.494	0.005	63.1%
6	Adversarial (Positive)	0.877	0.005	0.876	0.005	0.1%
7	Adversarial (Negative)	0.321	0.009	0.468	0.010	63.0%

4.2 Results on MNLI

Table 2 shows the MNLI evaluation results, computed on the stratified 3,600-instance subsample described in Section 3.1, with unmapped (-1) outputs scored as incorrect. The Original prompt achieved a mean accuracy of 0.824 and F1-score of 0.816, with no unmapped outputs—both essentially identical to the 0.824/0.817 obtained in preliminary experiments on the full 19,647-instance validation split, supporting the adequacy of this subsample size for the remaining seven prompts. Rewording caused a moderate accuracy decline to 0.726, with 13.5% of its outputs unmapped.

A strong contrast emerged under Style Change: the formal variant achieved the highest accuracy of any MNLI prompt (0.845) with a negligible unmapped rate (0.1%), while the casual variant collapsed to mean accuracy 0.338, with 61.7% of outputs unmapped. The confusion matrix for this prompt (available in the project repository) shows the model predicted *neutral* for essentially none of the casual-phrased inputs, indicating this phrasing caused the model to overwhelmingly fail to produce a parseable three-way label rather than merely mispredicting among the three classes.

Noise Injection had a comparatively mild effect on MNLI (mean accuracy 0.817, 7.1% unmapped), consistent with the label option text (`entailment/contradiction/neutral`) remaining largely intact under the character-level perturbation, unlike its devastating effect on SST-2.

Among the adversarial variants, the contradiction-type attack essentially matched the Original baseline (0.828 versus 0.824), while the entailment-type and neutral-type attacks caused substantially larger drops (0.663 and 0.621, respectively), accompanied by much higher unmapped rates (22.8% and 13.5%, versus 5.8% for the contradiction-type attack).

Pooling predictions across the ten runs and inspecting the per-class confusion matrix for the Original prompt (available in the project repository) shows that the gap between accuracy (0.824) and macro F1 (0.816) is driven mostly by the *neutral* class: recall for neutral is 0.608, compared to 0.920 for entailment and 0.943 for contradiction. This neutral-class weakness recurs across nearly every MNLI prompt variant (see the per-class results in the project repository), and it also helps explain why the contradiction-type adversarial attack alone

approximately preserves accuracy: under this prompt, neutral recall rises to 0.664 relative to the Original baseline, so instances the model already tended to mishandle out of the neutral class are redirected toward contradiction rather than toward an unmapped or entailment-labeled failure, while contradiction recall itself falls only slightly (0.943 to 0.902) and entailment recall is essentially unaffected (0.920 to 0.919). The entailment-type and neutral-type attacks, by contrast, directly suppress recall on whichever class they instruct the model away from (e.g., neutral recall falls to 0.260 under the entailment-type attack), which produces the much larger accuracy drops observed for those two variants.

Table 2. Evaluation results on MNLI dataset (3,600-instance stratified subsample).

No.	Category	Mean Acc.	Std Acc.	Mean F1	Std F1	Unmapped
1	Original	0.824	0.003	0.816	0.003	0.0%
2	Rewording	0.726	0.005	0.776	0.004	13.5%
3	Style Change (Formal)	0.845	0.002	0.843	0.002	0.1%
4	Style Change (Casual)	0.338	0.002	0.367	0.003	61.7%
5	Noise	0.817	0.003	0.845	0.002	7.1%
6	Adversarial (Entailment)	0.663	0.003	0.708	0.003	22.8%
7	Adversarial (Contradiction)	0.828	0.003	0.849	0.003	5.8%
8	Adversarial (Neutral)	0.621	0.005	0.669	0.005	13.5%

4.3 Results on RTE

Table 3 shows RTE evaluation results, with unmapped (-1) outputs scored as incorrect. The Original prompt yielded a mean accuracy of 0.772 and F1-score of 0.753 with low standard deviation. The Formal Style Change variant performed best overall on this dataset (mean accuracy 0.790, F1-score 0.775), and Rewording also slightly exceeded the Original baseline (0.782/0.767).

Unlike in the earlier, uncorrected analysis, Style Change (Casual) is not merely an F1-variance anomaly here: it degrades mean accuracy to 0.696, with 20.6% of its outputs unmapped, indicating this phrasing measurably impairs the model's ability to produce a parseable label rather than only shifting its class balance. The confusion matrix for this prompt (available in the project repository) shows this unmapped mass falls disproportionately on *contradiction* instances rather than *entailment* ones, mirroring the class-specific fragility also seen on MNLI (Section 4.2), though here the more vulnerable class is contradiction rather than neutral.

Noise reduced mean accuracy to 0.627, with the highest unmapped rate observed on this dataset (29.3%)—a degree of disruption intermediate between SST-2's near-catastrophic Noise collapse and MNLI's comparatively mild one.

Adversarial attacks kept accuracy close to the Original baseline (0.773–0.786) with modest unmapped rates (1.6–2.5%). Their F1-scores (0.767–0.778) are now close to their accuracies rather than substantially depressed, which confirms that

the low adversarial F1-scores previously reported for RTE (0.513–0.518, with F1 standard deviations as high as 0.124) were largely an artifact of the phantom-class F1 bug described in Section 3.4, rather than evidence that adversarial prompts were biasing RTE predictions toward a particular label.

Table 3. Evaluation results on RTE dataset.

No.	Category	Mean Acc.	Std Acc.	Mean F1	Std F1	Unmapped
1	Original	0.772	0.006	0.753	0.007	0.0%
2	Rewording	0.782	0.003	0.767	0.003	0.2%
3	Style Change (Formal)	0.790	0.006	0.775	0.007	0.0%
4	Style Change (Casual)	0.696	0.008	0.744	0.009	20.6%
5	Noise	0.627	0.006	0.688	0.006	29.3%
6	Adversarial (Entailment)	0.773	0.006	0.767	0.007	2.5%
7	Adversarial (Contradiction)	0.786	0.005	0.778	0.005	1.6%

4.4 Cross-Dataset and Cross-Category Comparison

Table 4 summarizes aggregated performance per dataset (each dataset's value is the unweighted mean/average-of-standard-deviations over its own category rows in Tables 1–3). SST-2 achieved the highest mean F1-score (0.801) despite a middling mean accuracy (0.758), reflecting its two near-complete collapses (Noise and negative-bias Adversarial) pulling accuracy down further than F1. MNLI achieved the lowest mean accuracy (0.708) of the three datasets, driven primarily by its Style Change (Casual) collapse. RTE achieved the highest mean accuracy (0.747), consistent with it having no single catastrophic-collapse category as extreme as SST-2's Noise/Adversarial(Negative) or MNLI's Style Change (Casual).

Table 4. Aggregated evaluation results per dataset.

Dataset	Mean Acc.	Std Acc.	Mean F1	Std F1
SST-2	0.758	0.004	0.801	0.005
MNLI	0.708	0.003	0.734	0.003
RTE	0.747	0.006	0.753	0.006

Table 5 aggregates performance by prompt category across all datasets (each category's value averages every sub-variant of that category across all three datasets, e.g. Style Change averages six rows: formal and casual, for each of SST-2, MNLI, and RTE). Original prompts achieved the highest mean accuracy (0.845) and F1-score (0.836), with Rewording close behind (0.817/0.829). Style Change produced a larger drop (0.759/0.769), and Adversarial attacks a larger

one still (0.696/0.731). Noise caused the largest overall drop by both metrics (mean accuracy 0.595, F1-score 0.676), confirming it as the single most damaging category even after outputs that fail to map to a valid label are correctly scored as incorrect rather than excluded.

It should be noted that Table 5 averages macro F1-scores computed on a 2-class dataset (SST-2), a 2-class dataset (RTE), and a 3-class dataset (MNLI). Because the baseline expected value of macro F1 under random guessing differs by number of classes, this cross-dataset average should be read only as a coarse, descriptive summary of the overall ranking between categories, not as a statistically calibrated single metric. The per-dataset breakdowns in Tables 1–3, which keep each dataset's class cardinality fixed, provide the more rigorous basis for comparison and are what the category-level discussion in Section 5 primarily draws on.

Table 5. Aggregated evaluation results per prompt variation category.

Category	Mean Acc.	Std Acc.	Mean F1	Std F1
Original	0.845	0.004	0.836	0.004
Rewording	0.817	0.004	0.829	0.003
Style Change	0.759	0.004	0.769	0.005
Noise	0.595	0.004	0.676	0.004
Adversarial	0.696	0.005	0.731	0.005

5 Discussion

5.1 Impact of Prompt Variation by Category

Rewording and Style Change produced moderate, non-uniform performance degradation. On SST-2, both categories had negligible impact (accuracy within 0.002–0.004 of the Original baseline), suggesting that the model's reliance on content-level features (sentiment-bearing words in the input text) outweighs instruction phrasing for this simple binary task. On RTE, Rewording and Style Change (Formal) likewise stayed close to or slightly above baseline, but Style Change (Casual) dropped accuracy by 7.6 points (0.772 to 0.696) with a 20.6% unmapped rate. On MNLI, the casual-phrasing effect is far more severe: accuracy collapses from 0.824 to 0.338, with 61.7% of outputs unmapped. Across both NLI datasets, then, casual phrasing does not primarily cause the model to pick the wrong one of two or three labels; it substantially increases the rate at which the model fails to produce any recognizable label at all, and this failure-to-parse effect is far larger on the three-way MNLI task than on the binary RTE task.

Noise Injection was the most consistently damaging category overall (Table 5), and unmapped-output rates make the mechanism behind this unambiguous. On SST-2, where noise directly corrupted label tokens (`pos!tive`,

`neg@tive`) in the prompt, 63.1% of outputs were unmapped and accuracy collapsed to 0.340. On RTE, where the label words themselves were not corrupted but the instruction around them was, 29.3% of outputs were unmapped and accuracy fell to 0.627. On MNLI, where the label option text (`entailment/contradiction/neutral`) remained fully intact, only 7.1% of outputs were unmapped and accuracy fell only to 0.817—close enough to the 0.824 Original baseline that Noise is, on this dataset alone, less damaging than Rewording or Style Change (Casual). This gradient across datasets highlights that the *location of noise within the prompt*—whether it targets label tokens directly, surrounding instruction text, or leaves labels untouched—is a critical determinant of its destructive effect, and that effect is now visible directly as an unparseable-output rate rather than only as a downstream accuracy drop.

Adversarial Attacks showed the most varied and task-dependent behavior of any category. On SST-2, the negative-bias attack reduced mean accuracy to 0.321, still below random-chance expectation for a near-balanced binary dataset, while the positive-bias attack degraded accuracy only mildly (0.877) despite both attacks sharing the same instruction template ("Pretend/Regardless...always answer X"). This asymmetry implies the model was successfully misled toward the negative label specifically, rather than simply confused by an out-of-distribution instruction. On MNLI, the three adversarial variants diverged sharply: the contradiction-type attack essentially matched the Original baseline (0.828 versus 0.824), while the entailment-type and neutral-type attacks caused substantially larger drops (0.663 and 0.621, respectively), accompanied by much higher unmapped-output rates (22.8% and 13.5%, versus 5.8% for the contradiction-type attack). The confusion matrices in the project repository (Section 3.4) indicate this divergence is closely tied to the *neutral* class specifically: neutral has the lowest recall of the three classes under almost every MNLI prompt (e.g., 0.608 under the Original prompt, versus 0.920 and 0.943 for entailment and contradiction), and the contradiction-type attack is the only variant under which neutral recall *improves* relative to the Original baseline (0.608 to 0.664), while entailment and contradiction recall are each only marginally affected. The entailment-type and neutral-type attacks, by contrast, directly suppress recall on whichever class they instruct the model away from (e.g., neutral recall falls to 0.260 under the entailment-type attack), which is what drives their much larger accuracy drops. On RTE, all adversarial variants stayed close to the Original baseline (0.773–0.786) with low unmapped rates (1.6–2.5%); their macro F1-scores (0.767–0.778) are correspondingly close to their accuracies, in contrast to the artificially depressed F1-scores an uncorrected phantom-class computation would have produced whenever any unmapped output occurred (Section 4.3). Taken together, adversarial vulnerability on this model is not explained by a single mechanism: it ranges from targeted label-flipping (SST-2), to an attack direction happening to align with the model's pre-existing class-recognition weaknesses (MNLI), to near-immunity (RTE).

5.2 Cross-Dataset Robustness Patterns

SST-2's high baseline performance but extreme vulnerability to noise and negative-bias adversarial attacks demonstrates a characteristic of simple instruction-dependent tasks: when the instruction is intact, the task is easy; when the instruction is disrupted badly enough that the label tokens themselves become unreadable, the model has little grounding to fall back on and largely stops producing parseable output at all (63% unmapped under both Noise and Adversarial (Negative)).

MNLI's larger input content (average 22.3 words per premise) partially compensates for noisy instructions, provided the label option text remains legible: the model can still extract semantic relational cues from the long input text even when the surrounding instruction is somewhat degraded, which explains MNLI's relative tolerance to Noise Injection specifically (7.1% unmapped, accuracy 0.817 versus a 0.824 baseline). However, MNLI is simultaneously the dataset most vulnerable to casual phrasing (61.7% unmapped under Style Change (Casual)) and to two of its three adversarial variants, suggesting that its robustness is contingent on which part of the prompt is disrupted and which class an attack targets, rather than being uniformly higher or lower than the other two datasets.

RTE's behavior sits between the other two: its binary label space is simpler than MNLI's three-way distinction, but its short, abstractly phrased premise–hypothesis pairs leave less redundant content for the model to fall back on than MNLI's longer inputs, which is consistent with RTE showing a substantial (though not catastrophic) unmapped-driven drop under both Noise (29.3%) and Style Change (Casual) (20.6%) while remaining comparatively robust to Rewording, Style Change (Formal), and every Adversarial variant tested.

5.3 Limitations

Several limitations of this study should be acknowledged. First, the study uses only a single model (Flan-T5 Large), so findings may not generalize to other architectures. Second, prompt variations were manually designed; systematic coverage of the variation space cannot be guaranteed. Third, the study measures robustness only through performance drop, covering three of the seven robustness dimensions identified by Kumar and Mishra [6], not all. Fourth, the cross-dataset aggregation in Table 5 averages macro F1-scores across datasets with different numbers of classes (two for SST-2 and RTE, three for MNLI), which is only a descriptive summary rather than a statistically rigorous comparison, as discussed in Section 4.4. Fifth, for computational reasons, MNLI was evaluated on a 3,600-instance stratified subsample rather than its full 19,647-instance validation split (Section 3.1); results for the Original prompt on this subsample (accuracy 0.824, F1 0.816) closely matched preliminary full-split results (0.824/0.817), which is reassuring, but an earlier trial with a smaller 600-instance subsample produced a visibly different result for the Adversarial (Contradiction) prompt specifically, so some sampling variance in the remaining MNLI prompts cannot be ruled out without evaluating the full split. Sixth, each reported mean/standard deviation is computed from ten repeated runs per prompt; no formal statistical test (e.g.,

paired significance testing across prompts) is applied in this version of the paper to determine whether observed differences between prompt variants are statistically significant, which we plan to address with a larger number of runs in future work.

6 Conclusion

This study evaluated the robustness of Flan-T5 Large against four categories of prompt variation across three text classification benchmarks using the PromptBench framework.

Key findings are as follows. First, Noise Injection was the most consistently and severely damaging category (mean accuracy 0.583, mean F1-score 0.665 across datasets), particularly when character-level perturbations targeted label keywords in the prompt, driving the unmapped (unparseable) output rate as high as 63.1% on SST-2. Second, Adversarial Attacks produced the most heterogeneous effects: on SST-2 a negative-bias attack drove accuracy below chance level, while on MNLI a contradiction-type attack essentially preserved baseline accuracy but entailment-type and neutral-type attacks caused substantial drops, with the size of the drop tied to how reliably the model already recognized the targeted class rather than to that class's share of the dataset. Third, Rewording and Style Change caused moderate, task-dependent degradation, with formal register generally more robust than casual, and multi-class NLI tasks more sensitive than binary sentiment classification. Fourth, Flan-T5 demonstrated sufficient robustness against mild linguistic variations but meaningful vulnerability to structural perturbations in instruction and label tokens.

These findings provide empirical guidance for practitioners designing prompts for LLM-based classification systems: instruction components and label keywords should be protected from corruption, and casual or ambiguous instruction styles should be avoided for tasks requiring fine-grained semantic distinctions.

Future work should extend evaluation to additional models to assess whether these patterns are architecture-specific or general, extend the class-wise confusion-matrix analysis introduced in this revision to SST-2 and RTE, and consider mitigation strategies such as instruction augmentation and prompt ensembling to improve robustness in deployment scenarios.

Acknowledgments. The authors thank the Faculty of Computer Science, Universitas Brawijaya, for supporting this research. Experiments were conducted using Kaggle Notebooks with NVIDIA Tesla T4 GPU resources.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Agrawal, S., et al.: Enhancing LLM robustness to perturbed instructions: an empirical study. arXiv preprint arXiv:2504.02733 (2025)

2. Chung, H.W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., et al.: Scaling instruction-finetuned language models. arXiv preprint arXiv:2210.11416 (2022). https://doi.org/10.48550/arXiv.2210.11416
3. Dagan, I., Glickman, O., Magnini, B.: The PASCAL recognising textual entailment challenge. In: Quiñonero-Candela, J., et al. (eds.) Machine Learning Challenges, LNCS, vol. 3944, pp. 177–190. Springer, Heidelberg (2006). https://doi.org/10.1007/11736790_9
4. Gan, C., Mori, T.: Sensitivity and robustness of large language models to prompt template in Japanese text classification tasks. arXiv preprint arXiv:2305.08714 (2023). https://doi.org/10.48550/arXiv.2305.08714
5. He, J., Rungta, M., Koleczek, D., Sekhon, A., Wang, F.X., Hasan, S.: Does prompt formatting have any impact on LLM performance? arXiv preprint arXiv:2411.10541 (2024). https://doi.org/10.48550/arXiv.2411.10541
6. Kumar, P., Mishra, S.: Robustness in large language models: a survey of mitigation strategies and evaluation metrics. arXiv preprint arXiv:2505.18658 (2025). https://doi.org/10.48550/arXiv.2505.18658
7. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. J. Mach. Learn. Res. **21**(140), 1–67 (2020). https://doi.org/10.48550/arXiv.1910.10683
8. Razavi, A., Soltangheis, M., Arabzadeh, N., Salamat, S., Zihayat, M., Bagheri, E.: Benchmarking prompt sensitivity in large language models. arXiv preprint arXiv:2502.06065 (2025). https://doi.org/10.48550/arXiv.2502.06065
9. Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C.D., Ng, A., Potts, C.: Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of EMNLP 2013, pp. 1631–1642. ACL, Seattle (2013). https://doi.org/10.18653/v1/D13-1170
10. Wallace, E., Feng, S., Kandpal, N., Gardner, M., Singh, S.: Universal adversarial triggers for attacking and analyzing NLP. arXiv preprint arXiv:1908.07125 (2021). https://doi.org/10.48550/arXiv.1908.07125
11. Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S.R.: GLUE: a multi-task benchmark and analysis platform for natural language understanding. In: Proceedings of the 2018 EMNLP Workshop BlackboxNLP, pp. 353–355. ACL, Brussels (2018). https://doi.org/10.18653/v1/W18-5446
12. Williams, A., Nangia, N., Bowman, S.R.: A broad-coverage challenge corpus for sentence understanding through inference. arXiv preprint arXiv:1704.05426 (2018). https://doi.org/10.48550/arXiv.1704.05426
13. Zhang, K., Wu, L., Yu, K., Lv, G., Zhang, D.: Evaluating and improving robustness in large language models: a survey and future directions. arXiv preprint arXiv:2506.11111 (2025). https://doi.org/10.48550/arXiv.2506.11111
14. Zhu, K., Wang, J., Zhou, J., Wang, Z., Chen, H., Wang, Y., et al.: PromptRobust: towards evaluating the robustness of large language models on adversarial prompts. arXiv preprint arXiv:2306.04528 (2024). https://doi.org/10.48550/arXiv.2306.04528
15. Zhu, K., Zhao, Q., Chen, H., Wang, J., Xie, X.: PromptBench: a unified library for evaluation of large language models. arXiv preprint arXiv:2312.07910 (2024). https://doi.org/10.48550/arXiv.2312.07910

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (http://creativecommons.org/licenses/by-nc/4.0/), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

