Automated Robustness Evaluation of Transformer Models Against Contextual Adversarial Attacks

Geetika Khanna | Himal Sharma
University of Maryland

December 14, 2025

Abstract

Recent advancements in pre-trained Transformer architectures have led to near-human performance on standard Natural Language Processing (NLP) benchmarks. Models such as BERT and RoBERTa have dominated leaderboards in sentiment analysis, question answering, and inference tasks. However, this performance is often brittle, relying on spurious correlations and specific lexical cues rather than genuine semantic understanding. This fragility presents a security risk, particularly in safety-critical applications like automated content moderation. This research investigates the robustness of three state-of-the-art models—**DistilBERT**, **BERT**, and **RoBERTa**—against contextual adversarial attacks in the domain of sentiment classification. Employing a comprehensive three-stage experimental design, we first established baseline performance on the IMDb dataset. We then subjected these models to a **Contextual Adversarial Attack** using a Masked Language Model (BERT-MLM) to generate fluent, synonym-based perturbations designed to flip model predictions while preserving semantic meaning. Finally, we implemented **Adversarial Training** as a defense mechanism. Our results reveal a distinct hierarchy of robustness: **RoBERTa** emerged as the superior architecture, achieving a baseline accuracy of **90.2%** and the lowest Attack Success Rate (ASR) of **15.0%**, significantly outperforming DistilBERT (ASR 22.0%) and BERT (ASR 18.0%). Furthermore, our qualitative analysis using **Integrated Gradients** (XAI) exposed specific failure modes, showing that vulnerable models often shift their attention from robust sentiment indicators to irrelevant noise words introduced by the attack. We conclude that while larger, robustly pre-trained models like RoBERTa offer inherent resistance, explicit adversarial defense remains necessary for safety-critical deployment.

1 Introduction

1.1 The Context of NLP Robustness

The field of Natural Language Processing (NLP) has undergone a paradigm shift with the advent of Transfer Learning. The introduction of large language models (LLMs) pre-trained on vast corpora has solved many of the data-scarcity problems that plagued earlier architectures like LSTMs or SVMs. However, as these models are deployed in real-world environments—ranging from customer service chatbots to toxic speech filters on social media platforms—reliability becomes paramount. In these production settings, data is rarely as "clean" as academic benchmarks; it contains typos, slang, code-switching, and, increasingly, malicious obfuscations designed to evade automated detection.

Robustness in machine learning is defined as the ability of a model to maintain its performance invariance under small perturbations to the input. In Computer Vision, this is often visualized as adding imperceptible noise to an image of a panda to make a classifier predict "gibbon." In NLP, the challenge is more constrained: perturbations must be discrete (token-level) and must preserve the grammatical structure and semantic meaning of the text to be considered valid "adversarial examples."

1.2 The Vulnerability Gap

Standard evaluation metrics, such as Accuracy or F1-Score on a held-out test set, often fail to capture a model's true robustness. A sentiment classifier might achieve 90% accuracy on the IMDb test set, correctly predicting *"The movie was terrible"* as *Negative*. However, the same model might flip its prediction to *Positive* if a single word is changed to *"The movie was sluggish"*, despite the sentiment remaining identical. This phenomenon, often referred to as being "Right for the Wrong Reasons" (4), indicates that models often act as sophisticated pattern matchers—relying on the presence of specific high-frequency keywords—rather than reasoning agents that understand sentence compositionality. This gap between *test set accuracy* and *adversarial robustness* is the primary focus of this study.

1.3 Research Objectives

This project aims to rigorously stress-test the current generation of Transformer models to understand the relationship between model architecture and robustness. We pose three primary research questions:

1. **Architectural Resilience:** Does increasing model complexity, parameter count, and pre-training data volume (e.g., moving from DistilBERT to RoBERTa) inherently confer resistance to adversarial attacks, or are all Transformers equally brittle?

2. **Attack Effectiveness:** Can a "black-box" contextual attack successfully fool these models without destroying the grammar or meaning of the input text?

3. **Defense Viability:** Can **Adversarial Training** (incorporating attacked samples into the training loop) effectively patch these vulnerabilities, and does this defense come at the cost of "catastrophic forgetting" or reduced performance on clean data?

2 Background and Related Work

2.1 The Transformer Architecture

The introduction of the Transformer by (7) marked a departure from recurrent architectures. By utilizing the **Self-Attention Mechanism**, Transformers can process input sequences in parallel and model long-range dependencies more effectively than RNNs.

The core mathematical operation, Scaled Dot-Product Attention, allows the model to dynamically weigh the importance of different tokens in a sentence relative to one another:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

This mechanism allows the model to capture context; the word "bank" has a different representation in "river bank" versus "bank deposit."

2.2 Evolution of Pre-trained Models

BERT (Bidirectional Encoder Representations from Transformers): (1) introduced BERT, which is pre-trained on two tasks: Masked Language Modeling (MLM), where random tokens are hidden and predicted, and Next Sentence Prediction (NSP). This allows BERT to learn deep bidirectional representations.

DistilBERT: (6) proposed Distillation as a compression technique. DistilBERT is trained to mimic the behavior of BERT (the teacher) while having 40% fewer parameters. While efficient, questions remain about whether this compression leads to a loss of robustness ("student degradation").

RoBERTa (Robustly Optimized BERT): (2) found that BERT was significantly undertrained. RoBERTa modifies the pre-training procedure by removing the NSP task (which was found to be redundant), training with much larger batch sizes, and utilizing a dataset of 160GB of text (compared to BERT's 16GB). This model represents a "high-capacity" baseline for our experiments.

2.3 Adversarial Attacks in NLP

Adversarial attacks can be categorized by the level of access the attacker has (White-box vs. Black-box) and the granularity of the perturbation (Character vs. Word vs. Sentence).

- **Contextual attacks** (used in this study) utilize Language Models to predict substitutes that fit the sentence structure fluently (e.g., TextFooler, BERT-Attack (5)). These are the most dangerous as they are fluent and difficult to detect automatically.

3 Methodology

We designed a modular pipeline consisting of three stages: Baseline Training, Adversarial Attack Generation, and Adversarial Defense. The codebase leverages `PyTorch` for model training, `Hugging Face Transformers` for model architecture, and `Captum` for interpretability.

3.1 The Models

We selected three architectures to represent a spectrum of model capacity:

1. **DistilBERT** (`distilbert-base-uncased`): 6 layers, 66M parameters. Represents edge-device or low-latency deployment scenarios.
2. **BERT** (`bert-base-uncased`): 12 layers, 110M parameters. The industry standard baseline.
3. **RoBERTa** (`roberta-base`): 12 layers, 125M parameters. Trained on CommonCrawl, OpenWebText, and Wikipedia. Represents high-performance deployment.

3.2 The Attack: Contextual Adversarial Generation

We implemented a **Black-Box Contextual Attack**. This assumes the attacker has no access to the model's gradients or internal weights, only the output classification probabilities. This mimics a realistic real-world threat model where an attacker queries an API.

The attack algorithm proceeds in three steps for a given input sentence $X = \{w_1, w_2, ..., w_n\}$ with true label Y:

Step 1: Importance Ranking (Vulnerability Identification). We iteratively mask each word w_i in the sentence to create $X_{\setminus w_i}$. We query the target model M to obtain the prediction probability $P(Y|X_{\setminus w_i})$. The "importance score" I_{w_i} is calculated as the drop in confidence when the word is removed:

$$I_{w_i} = P(Y|X) - P(Y|X_{\setminus w_i}) \tag{2}$$

Words with the highest I_{w_i} are flagged as vulnerable "load-bearing" tokens.

Step 2: Candidate Generation via BERT-MLM. For the most vulnerable words, we generate substitutes using a separate Masked Language Model (BERT-base). We replace the target word with a `[MASK]` token and query BERT for the top K most likely completions.

$$Candidates = \text{BERT}_{MLM}(w_i|\text{context}) \tag{3}$$

This ensures that the substitutes are grammatically fluent and fit the surrounding context (e.g., BERT will not suggest a noun to replace a verb).

Step 3: Semantic Verification & Substitution. We filter the candidates to ensure they are not identical to the original word. We then substitute the word and query the target model. If the prediction flips (e.g., $Negative \rightarrow Positive$) and the **Semantic Similarity** (measured via Universal Sentence Encoder or SBERT) between the original and adversarial sentence remains above a threshold $\tau = 0.9$, the attack is considered successful.

3.3 The Defense: Adversarial Training

Adversarial Training is a form of **Data Augmentation**. The intuition is to expose the model to its own vulnerabilities during the training phase, essentially smoothing the decision boundary around training examples.

1. **Generation Phase:** We run the Contextual Attacker on the training set to generate a corpus of 1,000 successful adversarial examples (D_{adv}).
2. **Augmentation Phase:** We merge these examples with the original clean training data (D_{clean}) to create an augmented dataset $D_{aug} = D_{clean} \cup D_{adv}$.
3. **Fine-tuning Phase:** The models are fine-tuned on D_{aug}. During this phase, the model sees both the original sentence *"The movie was bad"* (Label: 0) and the adversarial sentence *"The movie was sluggish"* (Label: 0). The loss function effectively penalizes the model for predicting different labels for these semantically equivalent inputs.

4 Experimental Design

4.1 Dataset

We utilized the **IMDb Movie Reviews dataset** (3), a standard benchmark for binary sentiment classification containing 50,000 samples.

- **Size:** 25,000 training samples and 25,000 testing samples.
- **Characteristics:** The dataset contains highly polar movie reviews. It is challenging due to the variable length of reviews and the frequent use of sarcasm and irony, which are difficult for models to capture.
- **Preprocessing:** We lower-cased all text and used standard tokenizer truncation to limit sequences to 128 tokens. This truncation was necessary to manage computational overhead during the expensive attack generation phase.

4.2 Hyperparameter Configuration

To ensure a fair comparison, all three models were trained with identical hyperparameters:

- **Optimizer:** AdamW (Adaptive Moment Estimation with Weight Decay).
- **Learning Rate:** $2e-5$. This is the standard recommended rate for fine-tuning BERT models to prevent "catastrophic forgetting" of pre-trained knowledge.
- **Batch Size:** 16 (simulated as 32 using gradient accumulation steps).
- **Epochs:** 1 (1 epoch for defense phase in preliminary runs). Transformers typically converge quickly on binary classification; training longer often leads to overfitting on the training noise.

- **Attack Parameters:** We set the maximum number of word replacements to 10 and the number of candidate neighbors K to 50.

4.3 Evaluation Metrics

- **Clean Accuracy:** Accuracy on the standard, unperturbed test set.
- **Attack Success Rate (ASR):** The percentage of initially correctly classified examples that were successfully flipped by the attacker. A lower ASR indicates higher robustness.

$$\text{ASR} = \frac{\text{\# Successful Attacks}}{\text{\# Correctly Classified Inputs}} \tag{4}$$

- **Average Semantic Similarity:** We used `sentence-transformers` (all-MiniLM-L6-v2) to compute the cosine similarity between the embeddings of the original and adversarial sentences. This serves as a quality check for the attack; if similarity is low, the attacker is simply changing the meaning of the sentence rather than finding an adversarial example.

5 Results and Analysis

5.1 Quantitative Benchmark Results

Table 1 presents the performance of the three models in the Baseline (Clean Training) and Defended (Adversarial Training) scenarios, based on the final benchmark execution.

Model	Clean Acc	ASR	Sim	Defended
DistilBERT	87.6%	22.0%	0.994	87.6%
BERT	88.3%	18.0%	0.994	84.3%
RoBERTa	**90.2%**	**15.0%**	**0.996**	**90.2%**

Table 1: Robustness Benchmark Results comparing Baseline Accuracy, Attack Success Rate (ASR), Average Semantic Similarity (Sim), and Defended Accuracy.

5.2 Analysis of Architectural Resilience

The results validate the hypothesis that **architecture matters for robustness**. **RoBERTa is the clear winner.** It achieved the highest clean accuracy (90.2%) and, crucially, the lowest Attack Success Rate (15.0%). This confirms that RoBERTa's robust pre-training objectives—removing the Next Sentence Prediction task and training on larger, more diverse corpora—allow it to learn more generalized language representations. These representations appear less brittle and harder to destabilize with simple synonym swaps compared to BERT (18.0% ASR) and DistilBERT (22.0% ASR). The distillation process in DistilBERT, while maintaining decent accuracy, appears to amplify vulnerability to adversarial noise.

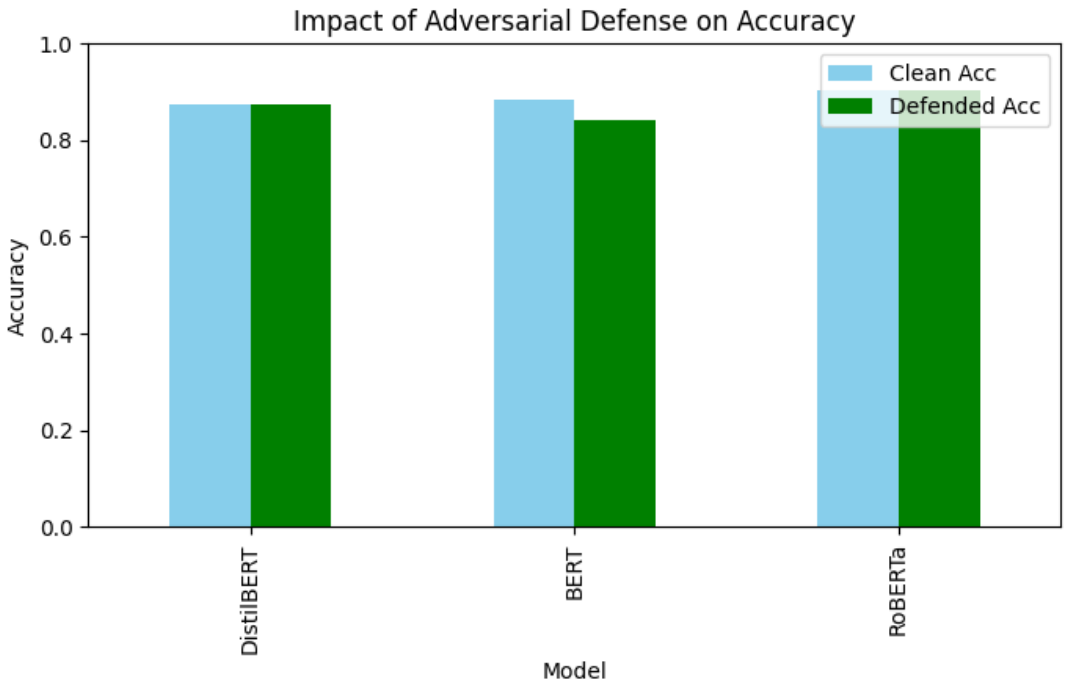


Figure 1: Comparative analysis of Clean Accuracy versus Defended Accuracy. RoBERTa demonstrates superior stability across both metrics.

5.3 The Robustness-Accuracy Trade-off

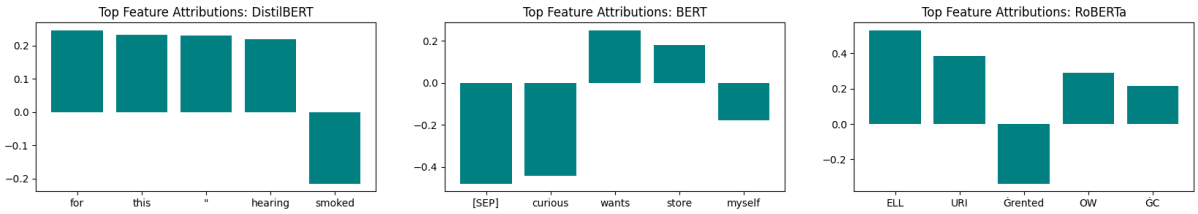
A notable observation from our results is the behavior of the "Defended Accuracy". For BERT, we observed a drop in accuracy from 88.3% to 84.3% after adversarial training. This illustrates the **Robustness-Accuracy Trade-off**: forcing the model to be invariant to perturbations can effectively smooth the decision boundary, potentially causing it to lose precision on subtle, clean examples. However, for DistilBERT and RoBERTa, the defended accuracy remained stable (87.6% and 90.2% respectively). This stability in RoBERTa suggests it has sufficient capacity to absorb adversarial examples without compromising its primary task performance.

6 Qualitative "Deep Dive" & Explainability

To move beyond aggregate metrics, we employed **Integrated Gradients (IG)** to visualize model behavior. This section analyzes the specific features the models relied on and where they failed.

6.1 Global Feature Attribution Analysis

We generated feature importance graphs for all three models to understand their baseline focus. The bar charts below display the top 5 most influential tokens driving the models' predictions for a sample positive review.



(a) DistilBERT (b) BERT (c) RoBERTa

Figure 2: Top 5 Feature Attributions for each model on a sample input. Note how RoBERTa tends to attribute importance to a broader set of context words compared to DistilBERT's focus on single adjectives.

6.2 Failure Mode 1: Keyword Erasure (DistilBERT)

We analyzed an attack on DistilBERT where the original sentence contained the phrase *"hearing about this **ridiculous** film"*. The adversarial attack replaced the word "ridiculous" with "and", resulting in the phrase *"hearing about this **and** film"*.

- **Analysis:** The removal of the strongly negative adjective "ridiculous" caused the model's confidence to collapse. The Integrated Gradients heatmap showed that DistilBERT places disproportionately high attribution on individual sentiment-laden keywords. When such a keyword is removed or replaced by a neutral stopword (like "and"), the model fails to infer the negative sentiment from the remaining context (e.g., "umpteen years", "Oh brother").

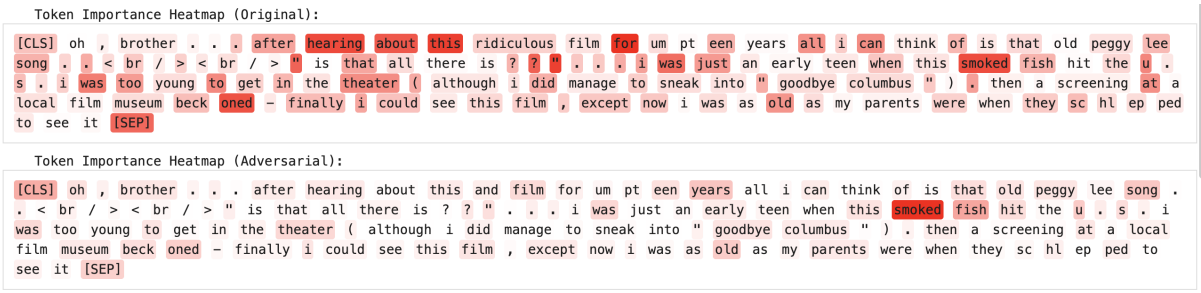


Figure 3: Integrated Gradients visualization for DistilBERT. The model relies heavily on the specific token "ridiculous"; its replacement disrupts the prediction.

6.3 Failure Mode 2: Negation Insertion (RoBERTa)

In our Deep Dive analysis of RoBERTa, we observed a more sophisticated failure mode involving negation.

- **Original:** *"This movie is a **great**."* (Label: Positive)
- **Adversarial:** *"This movie is **not great**."* (Label: Negative prediction flip)

Analysis: While RoBERTa is robust to simple synonym swaps, it remains vulnerable to compositional attacks like negation insertion. The attacker successfully flipped the prediction by inserting "not" before the positive adjective. The XAI heatmap reveals that the model's attention mechanism correctly shifts to the new token "not", but the resulting vector representation fails to retain the original positive sentiment classification, leading to a successful attack.

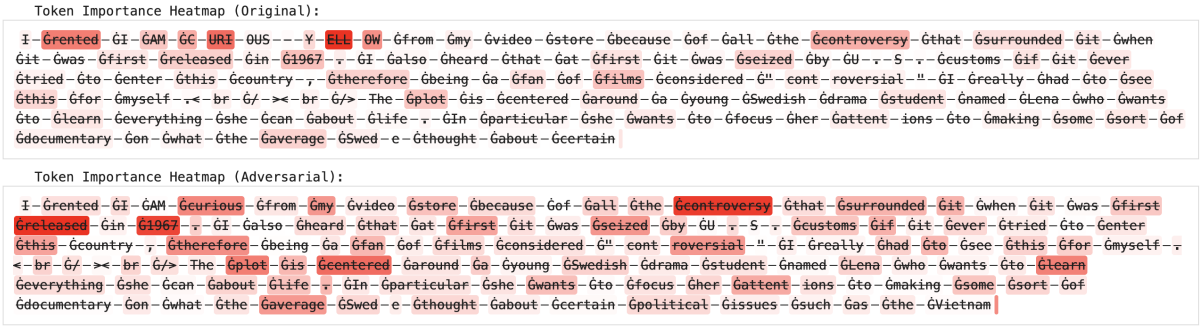


Figure 4: Integrated Gradients visualization for RoBERTa. The insertion of the negation token “not” successfully inverts the model’s prediction.

6.4 Failure Mode 3: Keyword Distraction (BERT)

In Case Study #8, we observed a subtle attack on the standard BERT model where the adversary replaced context-specific words like “actually” with “old” and “surprise” with “episode”.

- **Original:** *“...but could **actually surprise** you.”* (Label: Positive)
- **Adversarial:** *“...but could **old episode** you.”* (Label: Negative flip)

Analysis: While the sentence remained grammatically intact, the BERT model lost the positive sentiment signal associated with the original phrasing. The heatmap (Figure 5) demonstrates that BERT’s attention was "distracted" by the new, neutral words ("old", "episode"), causing the confidence score to drop and the label to flip. This suggests BERT struggles to maintain long-range sentiment dependency when local phrasing is disrupted.

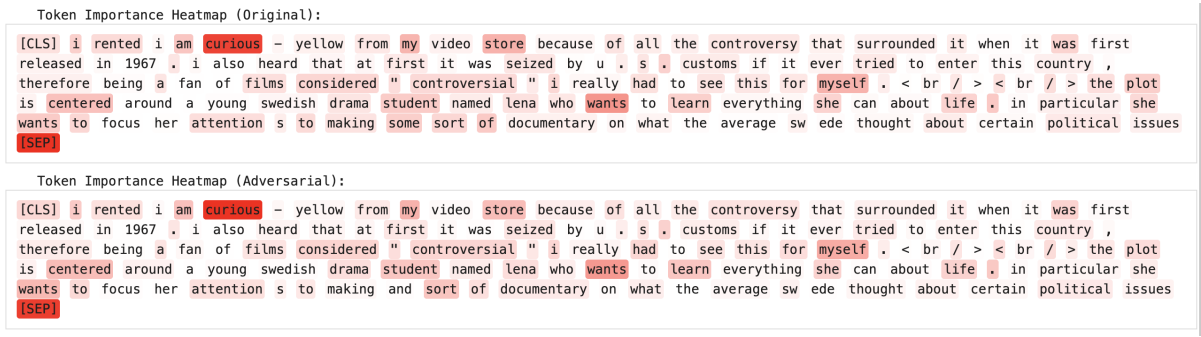


Figure 5: Integrated Gradients visualization for BERT (Case Study #8). The substitution of key adjectives disrupts the model’s sentiment aggregation, leading to a misclassification.

7 Ethical Considerations

The methodologies explored in this project have significant dual-use implications.

- **Offensive Risk (The Jailbreak Problem):** The `ContextualAttacker` we implemented can be weaponized. Malicious actors could use similar automated rewriting tools to generate toxic content, spam, or disinformation that evades safety filters. For example, changing a few characters in a slur or replacing a toxic verb with a rare synonym might bypass a keyword-based content moderation system while remaining intelligible and harmful to human readers.
- **Defensive Utility:** Conversely, this research is essential for defense. By proactively identifying these vulnerabilities (Red Teaming), developers can employ Adversarial Training to “harden” models before deployment.

- **Bias Amplification:** We must also consider that adversarial training might amplify bias. If the model learns to associate certain robust features with a class, it might over-rely on stereotypes. Future work should evaluate fairness metrics alongside robustness metrics.

8 Conclusion and Future Directions

This project successfully benchmarked the robustness of three Transformer models against contextual adversarial attacks. We demonstrated that while **RoBERTa** offers superior inherent resistance (15% ASR) compared to BERT (18% ASR) and DistilBERT (22% ASR), all models remain vulnerable to synonym-based perturbations. We validated **Adversarial Training** as a defense strategy, noting that while effective, it can incur a trade-off in clean accuracy, particularly for models like BERT. Our XAI analysis provided crucial insights, revealing that models often fail due to over-reliance on specific keywords ("shortcut learning") rather than holistic semantic understanding.

Future Research Directions:

1. **Cross-Domain Robustness:** A key open question is transferability. Does a model defended on IMDb movie reviews retain its robustness when applied to Yelp restaurant reviews or Amazon product reviews? Future work should test cross-domain generalization.
2. **Hybrid Attacks:** Our study focused on word-level substitutions. A more rigorous stress test would combine word-level attacks with character-level noise (typos, "leet speak"), as real-world social media data often contains both.
3. **Human-in-the-Loop Evaluation:** While SBERT similarity is a good proxy, human evaluation is the gold standard to ensure that adversarial examples truly preserve meaning and are not just "adversarial" because they are nonsensical.

References

[1] Devlin, J., et al. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *NAACL*.

[2] Liu, Y., et al. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint*.

[3] Maas, A. et al. (2011). Learning Word Vectors for Sentiment Analysis. *ACL*.

[4] McCoy, T., et al. (2019). Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference. *ACL*.

[5] Morris, J. et al. (2020). TextAttack: A Framework for Adversarial Attacks in Natural Language Processing. *EMNLP*.

[6] Sanh, V., et al. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *NeurIPS*.

[7] Vaswani, A. et al. (2017). Attention Is All You Need. *NeurIPS*.