Advancing Text Adversarial Example Generation Using Large Language Models

SEIO 2025

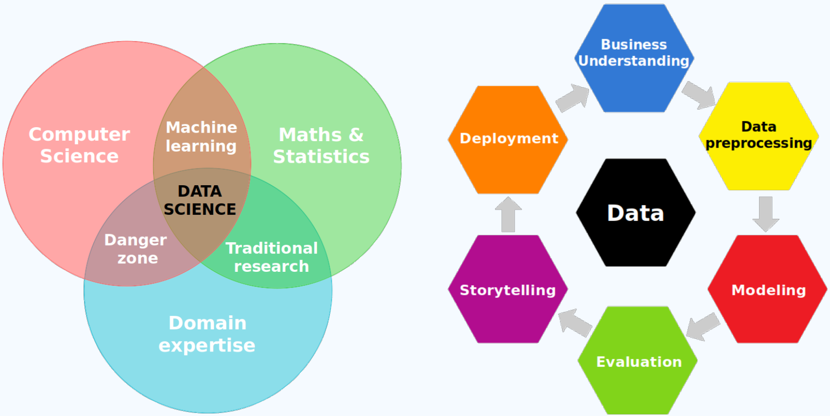
Natalia Madrueño [1] Alberto Fernández-Isabel [1]
Rubén R. Fernández [1] Isaac Martín de Diego [1]

(1) Rey Juan Carlos University, Data Science Laboratory

June 10, 2025

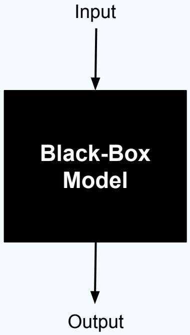
Table of Contents

Table of Contents



(a) Foundations (b) Applications

- **Lack of explainability poses several challenges**
  - Conceals weaknesses that degrade model quality and robustness
  - Complicates vulnerability detection, raising legal and ethical concerns
- **Recent advances in NLP rely on black-box models**
  - No access to or knowledge of their inner workings
  - Typically, only prediction scores are accessible
- **Adversarial examples for analyzing black-box score-based models**



Adversarial examples

- **Crafted inputs designed to fool victim models**
  - Introduce subtle perturbations to the original input
  - Similar to original input from human perspective
- **In text and NLP models**
  - Perturbations at different text levels (char, word, sentence...)
  - Preserve the semantic meaning of the original text
- **Existing state-of-the-art methods have limitations**
  - Focus on LLM perturbations at a single text level
  - Can be significantly improved

Original

The **movie** was **great**!
The **actor** was good
The director was nice

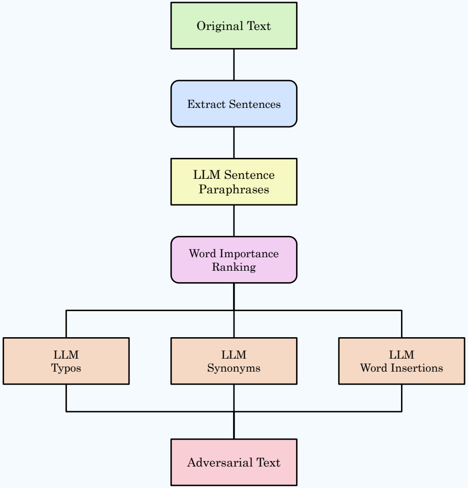
Adversarial

The **film** was **gr8**!
The **actor then** was good
What a nice director

Table of Contents

Proposed method

- **Leverages LLMs' text generation capabilities to produce adversarial examples**
- **Perturbations introduced in a 2-step process**
  1. Sentence-level perturbations
  2. Character- and word-level perturbations
- **Currently under revision at a JCR journal**



Sentence-level perturbations

- **Segment original text into sentences using a sentencizer**
  - $X \to S = (s_1, s_2, \ldots, s_n)$
- **Generate paraphrases for each sentence** s_i
  - $P_i = (p_{i1}, p_{i2}, \ldots, p_{im})$
- **Evaluate paraphrases based on their ability to deceive models**
  - Effect of replacing s_i with p_{ij} in victim model prediction scores

Sentence-level perturbations - LLM Paraphrases

- **Generates LLM paraphrases for each sentence**
  - Different syntax + word choice

```
Generate a list of paraphrases for the target sentence. Limit to
bullet points for each suggested paraphrase.
Sentence: "{}"
Answer: -
```

- **For each paraphrase p_{ij}, replace s_i in X**
  - If model completely deceived
    - Return adversarial example
  - If model deception is increased
    - Replace s_i with p_{ij} and continue iterating
- **If no adversarial examples has been found**
  - Continue with character- and word-level perturbations

```
Input  : Original input text X
Output: Sentence-level perturbed text X'
X' ← X;
for s_i ∈ ExtractSentences(X) do
    for p_ij ∈ GenerateParaphrases(s_i) do
        if ModelIsDeceived(p_ij, X') then
            return ReplaceSentence(p_ij, X');
        else if DeceptionIsIncreased(p_ij, X's) then
            X' ← ReplaceSentence(p_ij, X')
return X'
```

Character- and word-level perturbations

- **Segment modified text into words using a tokenizer**
  - $X' \rightarrow W = \{w_1, w_2, \dots, w_l\}$
- **Identify most vulnerable words using WIR**
  - Rank words based on the effect in model prediction scores of omitting w_l in X'
  - Extract 40% most vulnerable words
- **Generate typos, synonyms and word insertions for the most vulnerable words**
  - $Q_k = T_k \frown Z_k \frown L_k \frown R_k = (q_{k1}, \dots, q_{ke})$
- **Evaluate typos, synonyms and word insertions based on their ability to deceive models**
  - Effect of replacing w_k with q_{kh} in victim model prediction scores

- **Generates LLM typos for each vulnerable word**
  - Typographical variations

> Generate a list of common typos for the target word. Include extra whitespaces, random additional characters, and misplaced characters. Limit to bullet points for each suggested typo.
> Word: "{}"
> Answer: -

Character- and word-level perturbations - LLM Synonyms

- **Generates LLM synonyms for each vulnerable word**
  - Semantically similar words

Generate a list of synonyms for the target word in the context of the text below. Limit to bullet points for each suggested synonym.
Text: "{}"
Word: "{}"
Answer: -

- **Inserts LLM neutral words for each vulnerable word**
  - Do not affect the overall semantic meaning
  - Either to the left or right of the target word

Generate a list of neutral words that could naturally be inserted at the position marked by [INSERTION] in the text below. Limit to bullet points for each suggested insertion. Text: "{}" Answer: -

- **For each character- and word-level perturbation q_{kh}, replace w_k in X'**
  - If model completely deceived
    - Return adversarial example
  - If model deception is increased
    - Replace w_k with q_{kh} and continue iterating

```
Input  : Sentence-level perturbed text X'
Output: Text adversarial example X_adv
X_adv ← X';
W ← ExtractWords(X');
for w_k ∈ WIR(W, X') do
    for q_kh ∈ GenerateTyposSynonymsInsertions(w_k, X_adv)
      do
        if ModelIsDeceived(q_kh, X_adv) then
            return PerturbWord(q_kh, X_adv);
        end
        else if DeceptionIsIncreased(q_kh, X_adv) then
            X_adv ← PerturbWord(q_kh, X_adv);
        end
    end
end
return X_adv
```

Table of Contents

- **Two binary sentiment classification problems**
  - Classify text sentiment in positive or negative
- **Problem 1: Binary Stanford Sentiment Treebank (SST-2)**
  - Movie reviews from Rotten Tomatoes
  - Short texts consisting of a individual sentence
- **Problem 2: The Internet Movie Database (IMDB)**
  - Movie reviews from The Internet Movie Database
  - Longer texts consisting of several sentences

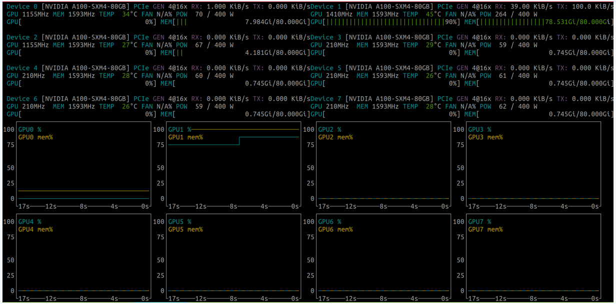
Model Setup

- **Victim models: open-weight LLMs**
  - Instruct versions of Gemma 2 9B, Llama 3.1 8B, Qwen 2.5 7B, and Yi 1.5 6B
- **Adversarial attacks: GPT-4o mini**
  - Proposed adversarial method
    - Combines multiple LLM perturbations
  - SOTA adversarial methods
    - One single LLM perturbation

```
Determine whether the sentiment of the following text is positive or negative. Answer only with the word "Positive" or "Negative".
Text: "{}"
Answer:
```

Model Setup

- **Compute Cluster**
  - 8 $\times$ NVIDIA A100 80 GB GPUs
- **OpenAI Integration**
  - API requests for text generation



Evaluation

- **Attack Success Rate (ASR)**
  - % of adversarial examples that fool the models
- **Semantic preservation**
  - Uses GPT-4o to assess semantic preservation
  - % of similar pairs between original text and the corresponding adversarial example

```
Determine whether the following two texts are semantically similar. Answer "YES" if they are semantically similar, or "NO" otherwise.
Text 1: "{}"
Text 2: "{}"
Answer:
```

Evaluation - ASR

SST-2 Adversarial Attack	ASR Gemma	Llama	Qwen	Yi
Paraphrases	0.13	0.15	0.15	0.20
Typos	0.47	0.71	0.68	0.72
Synonyms	0.60	0.64	0.62	0.65
Insertions	0.57	0.66	0.61	0.63
Paraphrases + Typos	0.62	0.80	0.79	0.83
Paraphrases + Synonyms	0.71	0.76	0.76	0.77
Paraphrases + Insertions	0.72	0.77	0.74	0.77
Typos + Synonyms	0.75	0.83	0.83	0.83
Typos + Insertions	0.76	0.87	0.85	0.88
Synonyms + Insertions	0.77	0.80	0.80	0.80
Paraphrases + Typos + Synonyms	0.81	0.90	0.89	0.89
Paraphrases + Typos + Insertions	0.84	0.93	0.92	0.91
Paraphrases + Synonyms + Insertions	0.83	0.88	0.87	0.88
Typos + Synonyms + Insertions	0.85	0.90	0.90	0.90
Proposed method (all perturbations)	**0.90**	**0.96**	**0.95**	**0.93**

IMDB Adversarial Attack	ASR Gemma	Llama	Qwen	Yi
Paraphrases	0.17	0.16	0.21	0.22
Typos	0.23	0.45	0.44	0.45
Synonyms	0.48	0.60	0.56	0.59
Insertions	0.46	0.51	0.52	0.47
Paraphrases + Typos	0.52	0.80	0.72	0.65
Paraphrases + Synonyms	0.70	0.81	0.77	0.73
Paraphrases + Insertions	0.71	0.79	0.76	0.67
Typos + Synonyms	0.57	0.84	0.73	0.70
Typos + Insertions	0.58	0.84	0.73	0.65
Synonyms + Insertions	0.69	0.81	0.71	0.71
Paraphrases + Typos + Synonyms	0.79	0.96	0.89	0.81
Paraphrases + Typos + Insertions	0.81	0.96	0.89	0.78
Paraphrases + Synonyms + Insertions	0.86	0.92	0.87	0.82
Typos + Synonyms + Insertions	0.72	0.92	0.82	0.77
Proposed method (all perturbations)	**0.89**	**0.98**	**0.91**	**0.87**

Tables: ASR for the evaluated adversarial example generation techniques that attack on the SST-2 and IMDB datasets the instruct versions of Gemma 2 9B, Llama 3.1 8B, Qwen 2.5 7B, and Yi 1.5 6B.

Evaluation - Semantic preservation

SST-2 Semantic Preservation	Semantically Similar			
	Gemma	Llama	Qwen	Yi
Paraphrases	0.97	0.98	0.98	1.00
Typos	0.98	0.99	0.97	0.99
Synonyms	0.95	0.95	0.95	0.96
Insertions	0.87	0.90	0.90	0.92
Paraphrases + Typos	0.98	0.99	0.99	0.99
Paraphrases + Synonyms	0.95	0.96	0.95	0.97
Paraphrases + Insertions	0.94	0.94	0.96	0.96
Typos + Synonyms	0.94	0.96	0.95	0.98
Typos + Insertions	0.89	0.96	0.92	0.96
Synonyms + Insertions	0.89	0.90	0.91	0.93
Paraphrases + Typos + Synonyms	0.96	0.98	0.97	0.99
Paraphrases + Typos + Insertions	0.95	0.97	0.96	0.98
Paraphrases + Synonyms + Insertions	0.93	0.94	0.94	0.96
Typos + Synonyms + Insertions	0.90	0.94	0.91	0.95
Proposed method (all perturbations)	0.94	0.97	0.94	0.98

IMDB Semantic Preservation	Semantically Similar			
	Gemma	Llama	Qwen	Yi
Paraphrases	0.95	0.95	0.94	0.97
Typos	0.97	0.99	0.99	0.99
Synonyms	0.99	0.99	0.98	0.98
Insertions	0.97	0.97	0.96	0.97
Paraphrases + Typos	0.97	0.97	0.97	0.97
Paraphrases + Synonyms	0.97	0.97	0.97	0.97
Paraphrases + Insertions	0.96	0.97	0.97	0.98
Typos + Synonyms	0.99	0.99	0.99	0.98
Typos + Insertions	0.99	0.97	0.96	0.95
Synonyms + Insertions	0.97	0.98	0.98	0.98
Paraphrases + Typos + Synonyms	0.97	0.97	0.97	0.97
Paraphrases + Typos + Insertions	0.97	0.98	0.98	0.99
Paraphrases + Synonyms + Insertions	0.97	0.97	0.96	0.98
Typos + Synonyms + Insertions	0.96	0.99	0.97	0.99
Proposed method (all perturbations)	0.97	0.98	0.99	0.98

Tables: Percentage of adversarial examples from the instruct versions of Gemma 2 9B, Llama 3.1 8B, Qwen 2.5 7B, and Yi 1.5 6B that preserve semantic similarity on the SST-2 and IMDB datasets according to GPT-4o.

Table of Contents

Discussion

- **Effectiveness of the proposed method**
  - Significantly higher ASR than previous SOTA
  - Semantic preservation similar to previous SOTA
- **Strengths and limitations**
  - The integration of multiple perturbations exploits several weaknesses
  - High computational costs despite using WIR

Table of Contents

- **Conclusions**
  - Presented a new adversarial example generation method based on LLMs
  - Integrates perturbations at different text-levels
  - Validated in short and long texts
- **Future work**
  - Use the proposal for model explainability and adversarial training
  - Reduce the computational cost of generating adversarial examples

Questions

Advancing Text Adversarial Example Generation Using Large Language Models

SEIO 2025

Natalia Madrueño [1] Alberto Fernández-Isabel [1]
Rubén R. Fernández [1] Isaac Martín de Diego [1]

(1) Rey Juan Carlos University, Data Science Laboratory

June 10, 2025