2025 NIST GenAI Text Challenge Evaluation Plan

NIST GenAI Team

September 16, 2025

Abstract

The NIST Generative AI (GenAI) Text Challenge is an evaluation program designed to probe the capabilities and limitations of generative AI models from three complementary perspectives: Generator, Prompter, and Discriminator. The challenge will focus on two core aspects: **indistinguishability** from human writing and the **believability** of generated narratives. In the **Generator task**, participants are to build models that produce text that is indistinguishable from human-authored passages. In the **Prompter task**, participants craft prompts that elicit two types of narratives from a generator: (a) highly accurate narratives and (b) misleading, but persuasive, narratives. In the **Discriminator task**, participants develop detectors that not only help decide whether a passage is AI-generated or human-written, but also output a score that is a prediction of how believable the content will be to a lay audience. Participation is open to all individuals and organizations that agree to follow the established rules and procedures.

Contents

DISCLAIMER

Certain commercial equipment, instruments, software, or materials are identified in this document to specify the experimental procedure adequately. Such identification is not intended to imply recommendation or endorsement by NIST, nor necessarily the best available for the purpose. The descriptions and views contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of NIST or the U.S. Government. This project has been reviewed by the Research Protections Office (RPO) under the reference number ITL-2023-0644.

Revision History

2025-09-03. First version.

1 Introduction

In recent years, the quality of digital content generated by artificial intelligence (AI) has advanced considerably across various modalities, including image, video, audio, and text. This surge in generative AI capability presents both opportunities and challenges—generative AI has facilitated creative expression and production, enabling artists, designers, and writers to create digital content at a much faster pace, but has also raised concerns regarding the authenticity and integrity of digital media, especially as it has become increasingly difficult to distinguish AI-generated content from human-generated content.

The NIST Generative AI (GenAI) program[1] supports research in generative AI with a structured series of evaluations testing the capabilities of multiple AI technologies in various modalities, beginning with text generation. The 2025 GenAI Text Challenge invites teams from academia, industry, and the research community to test AI capabilities by acting in different roles across the spectrum of AI actors: "generators", "prompters", and "discriminators."

- **Generators** build AI systems that produce text indistinguishable from human writing.
- **Prompters** craft two types of prompts: (a) those that elicit accurate, credible narratives; and (b) those that elicit intentionally misleading yet believable content.[2]
- **Discriminators** create AI systems that classify text as human- or AI-authored and output a believability score estimating the proportion of readers who will believe the text's main message.

1.1 Research Objective

The purpose of the upcoming GenAI Text Challenge is three-fold:

- Evaluate the ability of generative AI models to produce text that is indistinguishable from human writing.
- Assess the effectiveness of AI content discriminators in detecting AI-generated narratives as well as evaluating their believability.
- Explore the ability of prompters to generate credible as well as misleading content.

An analysis of prompting strategies will also help understand factors that influence one to believe in inaccurate or misleading AI-generated content.

[1] https://ai-challenges.nist.gov/genai

[2] The purpose of asking prompters to craft prompts for generative AI models that will produce believable but misleading narratives is that the resulting data can be used to train detectors to recognize such narratives.

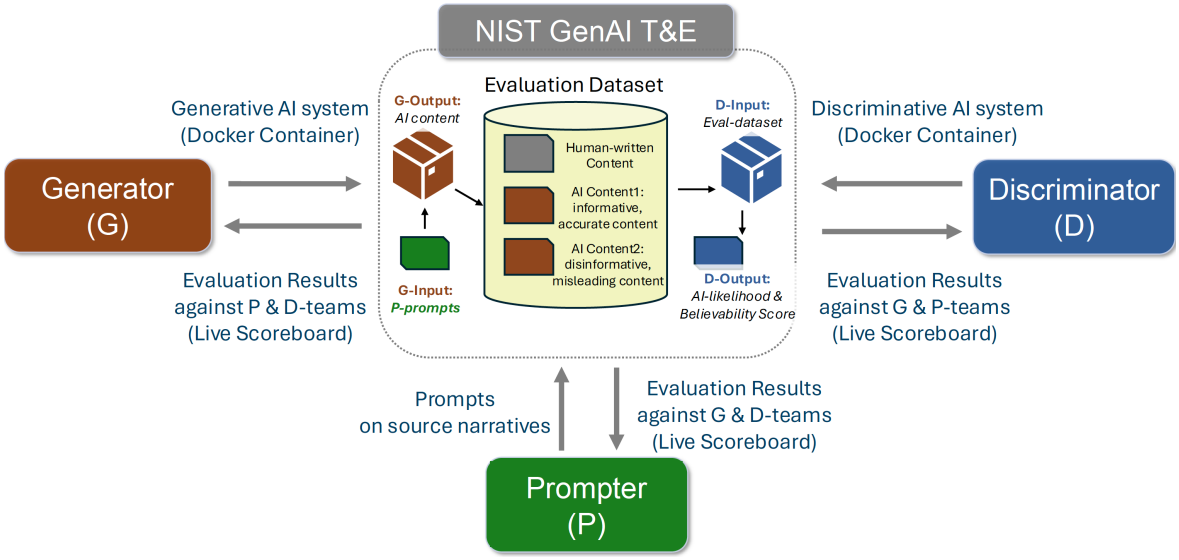

Figure 1: 2025 NIST GenAI Text Challenge - Evaluation Framework Overview.

Figure 1 provides an overview of the 2025 NIST GenAI text challenge framework. Through participation, teams will contribute to:

- **Driving AI innovation** by expanding knowledge of the capabilities and limitations of current models and content discriminators
- **Creating a responsible AI ecosystem** by understanding the complex interactions between models, prompts, and content discriminators
- **Informing AI standards** by evaluating and benchmarking diverse models and prompting strategies in a controlled setting.

2 Task

The 2025 GenAI Text evaluation supports the following three tasks:

- **Text Generators (Text-G)**: develop Generative AI models capable of generating high-quality text content. Submissions of the Text-G AI models should be in the form of a Docker container. These models will be used by Prompter teams (Text-P) to generate text content indistinguishable from human writing.
- **Text Prompters (Text-P)**: using submitted Text-G AI models and a given topic/source document as input, create two sets of prompts:

  1: The first set of prompts should generate accurate, relevant information in a brief, well-organized and fluent manner.

  2: The second set of prompts should generate a well-organized and fluent narrative that contains inaccurate, non-credible, and/or misleading information, crafted to be believable to the general public.

- **Text Discriminators (Text-D)**: given a text narrative, which may be AI-generated or human-generated, develop AI models to perform two main tasks:

  1: Assess the likelihood of the content being AI-generated by assigning a score between 0 and 1 (inclusive), such that scores closer to 1 suggest the content is AI-generated and scores closer to 0 suggest the content is human-generated (0.5 represents total indecision).

  2: Predict the proportion of the general public that would believe the main message of the text narrative by assigning a believability score between 0 (0% of the public would believe it) and 1 (100% of the population would believe it).

3 Evaluation Metrics

3.1 Performance Metrics

3.1.1 AUC-ROC

The AUC-ROC score reflects a detector's ability to distinguish between classes in general (for instance, target-class = AI-generated, non-target-class= human-generated). An AUC value of 1 means the detector is able to perfectly separate the classes with higher detection scores for the target-class and lower detection scores for the non-target class. An AUC value near 0.5 indicates that the detector performance is no better than random guessing. In this challenge, we place greater value on the indistinguishability of AI-generated content from human writing. So generators with AUC values close to 0.5 would be deemed successful (indistinguishable).

An AUC value close to 0 suggests the detector tends to assign higher detection scores to non-target class and lower detection scores to target class, contrary to what it is supposed to do. This will lead to classifiers that are consistently wrong, essentially predicting the incorrect class with high confidence. If the goal is to "fool the detectors" into classifying AI-generated content as human-generated and vice versa, then successful generators are those with AUC values less than or equal to 0.5. Such generator/prompter combinations are capable of deception.

3.1.2 Brier Score

The Brier score measures the accuracy of probabilistic predictions by calculating the mean squared difference between the predicted probability (i.e., detection score) and the actual outcome. A lower Brier score indicates better calibration and predictive accuracy, while a higher Brier score reflects greater error. For a well-calibrated detector, a score near 0.5 represents total uncertainty, or indistinguishability between AI and human written content, signaling confusion. In this case, the Brier score will be

$$(1 - 0.5)^2 = (0 - 0.5)^2 = 0.25.$$

However, if the detector assigns a detection score close to 0 for AI-generated content (i.e., high confidence that the content is human-generated when it is actually AI-generated), the generator has not merely confused the detector but has completely fooled it. If the detector is incorrectly confident — for example, assigning detection scores higher than 0.5 to human-written content or lower than 0.5 to AI-generated content — then the Brier score will exceed 0.25. At the extreme, a Brier score of 1 indicates the detector has been entirely fooled, showing 100% confidence in the wrong classification.

3.1.3 Believability score

Believability scores are intended to predict the proportion of general population who would believe the main message in the text. Higher scores (closer to 1) suggest that the main message of the text narrative is highly believable. Evaluation will consider the maximum and average believability scores across each set of narratives and prompts, as well as the overall distribution of scores. Whether these scores truly reflect human judgment will ultimately be validated using human evaluation over a diverse corpus where each human participant will assign their degree of belief using a number between 0 and 1 (inclusive).

Table 1: Believability Score and Human judgment

Human judgment	Degree of Belief
I strongly disbelieve it	0.0 - 0.2
I'm leaning toward not believing it	0.2 - 0.4
Unsure	0.4 - 0.6
I'm leaning toward believing it	0.6 - 0.8
I strongly believe it	0.8 - 1.0

Table 1 shows how a human reader might quantify their degree of belief in the main message of the narrative. A system description should be submitted including details of detection methods and any additional tools (e.g., search engines) or AI models utilized.

3.2 Definition of Success for Each Track

3.2.1 Text Generator (Text-G)

A generator G is considered successful when it produces content that is difficult for a discriminator D to distinguish from human writing. Specifically:

- AUC-ROC values less than or equal to 0.5, indicating either (a) the detector is unable to distinguish the AI-generated content from human writing and is randomly guessing, or (b) the detector is fooled into assigning lower detection scores to AI-generated content than to human-generated content.
- Brier scores greater than or equal to 0.25 reflecting that D is guessing or confidently misled.
- Higher believability scores (close to 1), showing that outputs guided by the prompt are convincing and human-like. Evaluation will consider the maximum and average believability scores across each set of narratives and generator systems, as well as the overall distribution of scores.

3.2.2 Text Prompter (Text-P)

A prompter P is considered successful when the prompts it generates lead to convincing outputs from the generator. Success is measured by:

- AUC-ROC values closer to 0.5, indicating that D cannot reliably separate human-written from AI-generated responses. Evaluations will consider the average AUC-ROC across all prompts submitted with a given generator.
- Higher Brier scores, reflecting that D is confidently misled.

- Higher believability scores (close to 1), showing that outputs guided by the prompt are convincing and human-like. Evaluation will consider the maximum and average believability scores across each set of narratives and generator systems, as well as the overall distribution of scores.

Importantly, prompter teams do not generate content themselves. Rather, they influence the quality, credibility, and plausibility of the generated responses through prompt design. Consequently, the evaluation of a prompter system is based on its downstream impact on both generator and discriminator outputs.

3.2.3 Text Discriminator (Text-D)

A discriminator D is considered successful when it can effectively distinguish AI-generated content from human writing. Indicators of success include:

- AUC-ROC values closer to 1, reflecting stronger discriminative power.
- Brier scores closer to 0, indicating accurate and well-calibrated probabilities.
- Believability scores that correlate well with the proportion of the population expressing high degrees of belief in the main message of the generated content.

4 Data Resources

4.1 Development Set

NIST will release development datasets to participating teams to support system design, calibration, and internal testing. The development dataset will include a set of statements and corresponding text narratives, paired with ground-truth scores: AI-likelihood labels and human-annotated believability scores. These examples will illustrate the types of topics and domains covered in the evaluation, giving each team a general sense of expected input and output.

4.2 Test Set

The test dataset will remain blind to participants throughout the evaluation period. Submitted systems will be executed on the NIST servers using this blind test dataset.

NIST reserves the right to not release the full test dataset following the conclusion of the evaluation. If any data is to be released, participants will be notified.

5 System Requirements and Specifications

Prompter teams must submit their prompts in a structured `JSON` format (see Appendix A).

Generator and **Discriminator** teams must package their systems as Docker images to ensure portability and consistent execution across evaluation environments. For detailed requirements and specifications, see Appendix B.

Additionally, Generator and Discriminator teams are required to provide comprehensive system descriptions. Refer to Appendix C for the Generator and Appendix D for the Discriminator.

6 Protocol and Rules

Rules and Restrictions

- Participants may use publicly available data, provided it complies with all applicable laws and regulations, to train their models.
- All systems must complete training, model selection, and tuning prior to submission to NIST.
- NIST will cap the number of submissions per week based on feedback from potential participants and the availability of NIST resources.

Advertising and Endorsement

- Participants may not make advertising claims about their standing in the evaluation or claim NIST endorsement of their system(s).
- The following language from the U.S. Code of Federal Regulations (15 C.F.R. §200.113 (d)) must be respected: NIST does not approve, recommend, or endorse any proprietary product or proprietary material.

Reporting and Publication

- NIST will generate a report summarizing the system results with participant team names.
- Participants may publish their research, provided they include proper references to the original source.

7 Agreement

All potential participants who wish to submit their systems and outputs will be required to complete and sign an agreement included in a registration form before uploading their submissions. This registration form will be provided to potential participants during the registration process for the evaluation.

8 Tentative Schedule

The schedule for the GenAI Text Challenge is:

September 18 , 2025	Evaluation Plan Posted
October 15, 2025	Registration Open
December 31, 2025	Registration Close
January 21, 2026	Round-table with Registrants
January 28, 2026	Dry-run with Sample Set
March 25, 2026	Dry-run Close
April 22, 2026	Evaluation Start
June 17, 2026	Evaluation Close

A Appendix: Prompter Submission Format

Submissions must be formatted in JSON and include all required fields. For the `truthfulness` field, use "0" to indicate prompts intended to produce factual and accurate narratives, and "1" to indicate prompts intended to produce misleading or inaccurate narratives. For the `submissionindex` field, assign each submission a value from 1 to 3, corresponding to its order of submission to the system.

```
{
  "team": "team ID",
  "version": "1.0",
  "submissionindex": "1",
  "prompt_list": [
    {
      "statement_id": "00001",
      "truthfullness": "0",
      "prompt": "Your prompt here",
    },
    {
      "statement_id": "00001",
      "truthfullness": "1",
      "prompt": "Your prompt here",
    },
    {
      "statement_id": "00002",
      "truthfullness": "0",
      "prompt": "Your prompt here",
    },
    ...
  ]
}
```

B Appendix: Container and Submission Requirements

All participants should provide systems packaged as Docker containers and submit with the required documentation and artifacts. This section outlines both the container specifications and the submission checklist.

1. Container Build and Runtime Requirements

B.0.1 Base Image

Use an official and immutable base image such as `nvidia/cuda:11.8-runtime` or `python:3.12-slim`, pinned to an exact digest or SHA.

B.0.2 Filesystem Layout

- `/app/prompts/` — read-only input directory.

- `/app/outputs/` — write-only output directory.
- `/app/models/` — optional model weights or cache.

B.0.3 Entrypoint and CLI

The container must set:

- `ENTRYPOINT = ["/app/run.sh"]`
- Required flags:

  `--prompt, --output.`

B.0.4 I/O Formats

- The NIST-validated input JSON file should be fed into the generator system.
- The output JSON should be validated by the required format. Each generator system must validate its `output.json` before producing artifacts. Failure to produce an artifact in the required format should be considered as an incorrect submission.

B.0.5 Resource Constraints

Containers should respect evaluator-imposed limits on CPU, GPU, and memory (e.g., `-cpus="8"`, `-gpus all`, `-memory="64g"`).

B.0.6 Logging and Health Checks

Logs should be written to `stdout` in a consistent format. An optional HTTP health-check may be exposed on port 8080.

B.0.7 Example Invocation

```
docker run --rm --gpus all \
  --cpus="8" \
  --memory="64g" \
  -v /prompts:/app/prompts:ro \
  -v /outputs:/app/outputs:rw \
  genarator-participant-system1:latest \
  --prompt /app/prompts/submission1.json \
  --output /app/outputs/output1.json
```

B.0.8 Output JSON format

```
{
  "team": "team ID",
  "prompt_team": "prompt_team_ID_parsed_from_input",
  "submissionindex": "index_parsed_from_input",
  "version": "1.0",
```

```
  "narrative_list": [
    {
      "statement_id": "00001",
      "truthfullness": "0",
      "narrative": "Generated output here",
    },
    {
      "statement_id": "00001",
      "truthfullness": "1",
      "narrative": "Generated output here",
    },
    {
      "statement_id": "00002",
      "truthfullness": "0",
      "narrative": "Generated output here",
    },
    ...
    }
  ]
}
```

2. Submission Package Requirements

The following should be included when submitting your system:

- **Container Artifacts:** Docker image (by name/tag) plus `Dockerfile`, `Docker-Compose` and any build scripts.
- **Dependencies:** Manifest file `environment.yml`.
- **System Description:** Architecture summary, model versions, hardware mapping, and citations. See the Appendix C and D
- **Documentation:** README with build steps, configuration options, invocation examples, and a compliance checklist.
- **Resource Usage Report:** Expected CPU, memory, and GPU requirements.

C Appendix: System Description Template (Generator)

Suggested template: https://www.ieee.org/conferences/publishing/templates.html

C.1 Section 1. Submission Identifier(s)

List your **team ID** and the **submission IDs** for which system outputs were submitted.

C.2 Section 2. System Description

Provide a concise technical and architectural overview of your generator system.

- **Foundation LLM:** e.g., GPT-4o, Claude Sonnet 4, etc.
- **Model Architecture:** Off-the-shelf vs. customized (hybrid, fine-tuned, etc.), Any novel modules or pipelines (RAG, MCP, etc.)
- **Unique Features / Modifications:** Custom pre- or post-processing, Specialized prompt engineering strategies, Any adapter layers, retrieval augmentation, etc.

C.3 Section 3. Docker Image/Container Specification

Describe the submitted docker image names & tags, a list of base images, container size, entrypoint & CMD, dependencies

C.4 Section 4. Training Data and Knowledge Sources

- **Training Datasets:** Dataset A (description, size, source), Dataset B (description, size, source)
- **Runtime Knowledge Sources (if used):** External APIs (e.g., Wikipedia, news feeds), Knowledge bases or retrieval corpora, Other dynamic data sources
- **Curation / Augmentation:** Filtering criteria, Data augmentation methods, Any alignment or cleaning procedures

C.5 Section 5. References

[1] A. Author, "Title of the paper," in *Proc. XYZ*, 2023.

[2] Dataset Name, Version, URL, Year.

D Appendix: System Description Template (Discriminator)

Suggested template: https://www.ieee.org/conferences/publishing/templates.html

D.1 Section 1. Submission Identifier(s)

List your **team ID** and the **submission IDs** for which system outputs were submitted.

D.2 Section 2. System Description

Provide a concise technical and architectural overview of your discriminator system.

- **Base discriminator model:** e.g., Vicuna-RADAR, OpenAI detector, etc.
- **Model Architecture:** Highlight unique features or modifications that differentiate your system from an off-the-shelf discriminator model.
- **Unique Features / Modifications:** If applicable, specify the model architecture (e.g., hybrid systems, fine-tuning only).

D.3 Section 3. Docker Image/Container Specification

Describe the submitted docker image names & tags, a list of base images, container size, entrypoint & CMD, dependencies

D.4 Section 4. Training Data and Knowledge Sources

- **Training Datasets:** Dataset A (description, size, source), Dataset B (description, size, source)
- **Runtime Knowledge Sources (if used):** External APIs (e.g., Wikipedia, news feeds), Knowledge bases or retrieval corpora, Other dynamic data sources
- **Curation / Augmentation:** Filtering criteria, Data augmentation methods, Any alignment or cleaning procedures

D.5 Section 5. References

[1] A. Author, "Title of the paper," in *Proc. XYZ*, 2023.

[2] Dataset Name, Version, URL, Year.