EURECOM
Sophia Antipolis

LettRAGraph

Hallucination or Creativity:

How to Evaluate AI-Generated Scientific Stories?

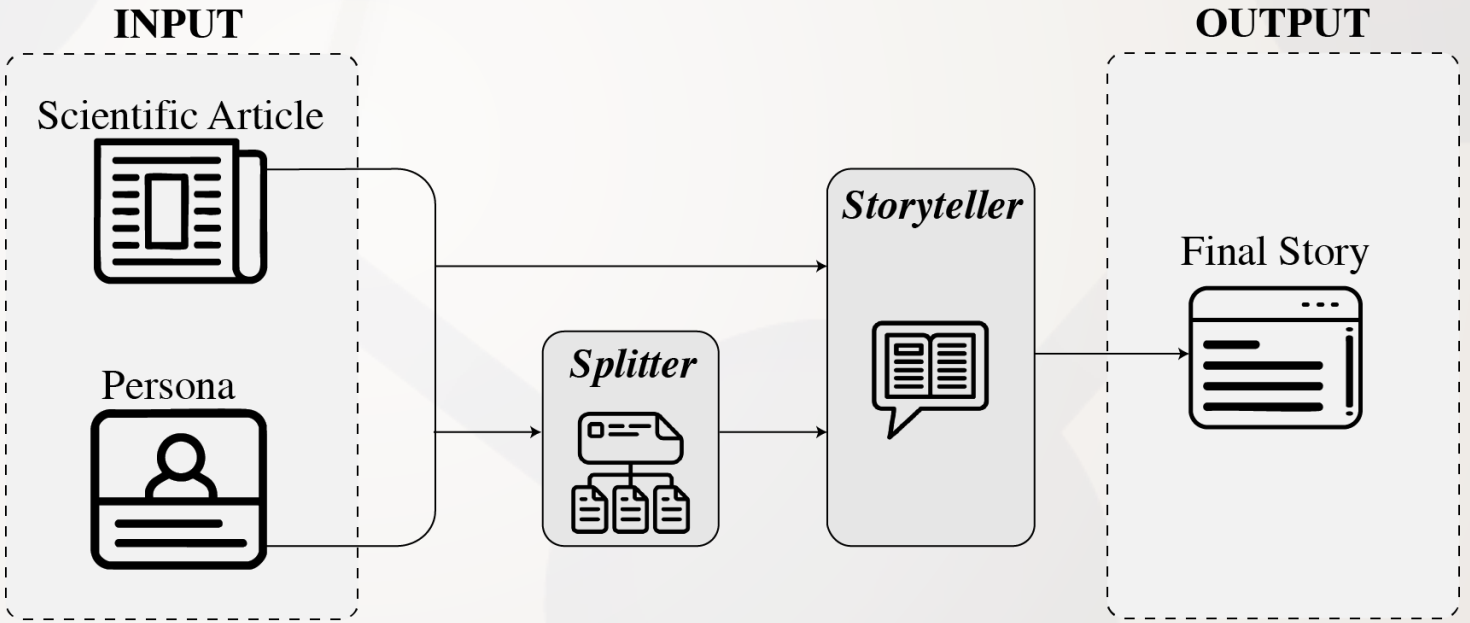
Alex Argese, Pasquale Lisena, Raphaël Troncy

ECIR 2026

Text2Story @ ECIR 2026
Delft, 29 March 2026

AI Scientist Storyteller

- Turn scientific papers into stories
- Adapt to different personas:
  - Researchers & Engineers
  - Student
  - Teacher
  - Policy Maker
  - Jounalist
  - Investor
  - General pulic



https://sciteller.tools.eurecom.fr/

Why evaluating scientific stories is challenging?

A good scientific story must preserve the paper's meaning while adapting structure and tone to a target audience.

- Traditional metrics alone (ROUGE, BERTScore, etc.) are not enough
- We want stories that are factual accurate, but have freedom in structure, for engagement and creativity

Example

Paper

"Attention Is All You Need" Vaswani et al., 2017

Factual Summary

The paper introduces the Transformer, a model based on attention mechanisms, removing recurrence and convolution.

Story for General Public

Imagine a reader that doesn't follow words one by one, but looks at the whole sentence at once, focusing on what matters most. That's the idea behind the Transformer.

Our Contribution

- A new metric for AI-generated scientific stories: the ***StoryScore*** metric

- A comparison of different hallucination detectors for AI-generated scientific stories

StoryScore

Composite evaluation metric

$$StoryScore = (0.3 \cdot ContextRecall) + (0.2 \cdot BERTScore) + (0.2 \cdot PromptCleanliness) + (0.1 \cdot TitleCoverage) + (0.1 \cdot NoRedundancy) + (0.1 \cdot NoHallucination)$$

Context Recall

- Measures vocabulary overlap between paper and story
- $Jaccard\ recall = \frac{|tokens\_story \cap tokens\_paper|}{|tokens\_paper|}$

→ i.e.: how much of the paper's vocabulary is reused in the story.

Scale: 0–1

High value = large lexical coverage of the paper

StoryScore

Composite evaluation metric

$$StoryScore = (0.3 \cdot ContextRecall) + (0.2 \cdot BERTScore) + (0.2 \cdot PromptCleanliness) + (0.1 \cdot TitleCoverage) + (0.1 \cdot NoRedundancy) + (0.1 \cdot NoHallucination)$$

BERTScore (semantic similarity)

- Measures semantic alignment with the source paper
- BERTScore between concatenated story sections and paper context
- Model: roberta-large

Scale: 0–1

High value = story semantically close to the paper

StoryScore

Composite evaluation metric

$$StoryScore = (0.3 \cdot ContextRecall) + (0.2 \cdot BERTScore) + (0.2 \cdot PromptCleanliness) + (0.1 \cdot TitleCoverage) + (0.1 \cdot NoRedundancy) + (0.1 \cdot NoHallucination)$$

Prompt Cleanliness
(Prompt Leakage Control)

- Verifies absence of prompt or control artefacts in the output
- Flags residual instructions, schema markers, or malformed structure

Scale: 0–1

High value = correct constraint following

StoryScore

Composite evaluation metric

$$StoryScore = (0.3 \cdot ContextRecall) + (0.2 \cdot BERTScore) + (0.2 \cdot PromptCleanliness) + (0.1 \cdot TitleCoverage) + (0.1 \cdot NoRedundancy) + (0.1 \cdot NoHallucination)$$

Title Coverage
(structural consistency)

- Comparison after normalisation of the Storyteller titles and Splitter titles.
- Necessary to check the consistency of the structure

Binary scale: 0 or1

Score = 1 if all 5 match exactly → 0 otherwise

StoryScore

Composite evaluation metric

$$StoryScore = (0.3 \cdot ContextRecall) + (0.2 \cdot BERTScore) + (0.2 \cdot PromptCleanliness) + (0.1 \cdot TitleCoverage) + (0.1 \cdot NoRedundancy) + (0.1 \cdot NoHallucination)$$

No Redundancy
(fluency and avoidance of loops)

- Tokenisation of the story text
- Generation of all 3-grams.
- Calculation of the frequency of the most common.
- $Repetition_{rate} = \frac{freq_{max}}{total_{3grams}}$
- $NoRepetition = 1 - repetition_{rate}$

Scale: 0–1

High value = fewer repeated phrases or words

StoryScore

Composite evaluation metric

$$StoryScore = (0.3 \cdot ContextRecall) + (0.2 \cdot BERTScore) + (0.2 \cdot PromptCleanliness) + (0.1 \cdot TitleCoverage) + (0.1 \cdot NoRedundancy) + (0.1 \cdot NoHallucination)$$

No Hallucination
(no invented entities)

- Extraction of candidate entities (Named Entities wih Spacy «PERS» and «ORG») from the paper and the story.
- $Hallucination_{rate} = \frac{\left(\begin{matrix} n.\ entities\ in\ the\ story \\ NOT\ present\ in\ the\ paper \end{matrix}\right)}{(n.\ entities\ in\ the\ story)}$
- $NoHallucination = 1 - hallucination_{rate}$

Scale: 0–1

High value = almost no fictional entities

Our Contribution

- A new metric for AI-generated scientific stories: the *StoryScore* metric

- A comparison of different hallucination detectors for AI-generated scientific stories

Hallucination Detection in Scientific Storytelling

Why it matters?

- A scientific story must remain **faithful** to the paper
- Invented entities, wrong names, or unsupported claims **reduce trust**
- Hallucination detection is therefore a key **evaluation dimension**

Why it is hard in a creative setting?

- Stories may use **metaphors and analogies**
- Stories may **simplify terms** for non-expert audiences
- **Not every wording difference is a hallucination**

Hallucination Detection in Scientific Storytelling

Example A - acceptable creative reformulation

Paper

"The model transfers information between modules.»

Story

"It acts like a messenger passing information forward."

Not a hallucination

metaphor, but meaning remains supported

Example B - real hallucination

Paper

Anonymized affiliation

Story

"The method was developed at the University of Birmingham."

Hallucination

invented factual entity

Hallucination detection
Methods Considered

- **NER-based**
  - Candidate entities extraction (words with capitalised initial letters) from the paper and the story.
  - $Hallucination_{rate} = \frac{\binom{n.\,entities\,in\,the\,story}{NOT\,present\,in\,the\,paper}}{(n.\,entities\,in\,the\,story)}$,

    $NoHallucination = 1 - hallucination_{rate}$

Evaluation examples

- *"University of Birmingham"* → not in paper → **correctly detected**
- story uses "AI", paper uses "Artificial Intelligence" → **incorrectly flagged**

- **MIRAGE***
  - MIRAGE re-generates multiple rewrites of the story
  - Computes alignment between the original generation and its own rewrites
  - If a concept is unstable or unsupported, the model assigns high hallucination probability

* https://github.com/bVendeville/MIRAGE

Evaluation examples

- to explain what is Robustness in AI, the story used a simpler example about object detection in images
  → **MIRAGE considers the metaphor as a hallucination**

Hallucination detection
Methods Considered

- **LLM-as-a-judge (Qwen 7B & GPT 5.1)**
  - A Large model (Qwen-7B) receives:
    - CONTEXT: the paper text
    - ANSWER: the story section
  - Generates a JSON containing:
    - name_accuracy
    - numeric_accuracy
    - overall_faithfulness
    - hallucinated_names / numbers

Evaluation examples

- *"Robust-kit, developed at the **University of Birmingham…**»*
  → **hallucination NOT detected**
- Correct story (human checked) LLM-as-judge incorrectly outputs hallucinated names:
  ["SwinUnetR", "MedSAM", "SAM-Med2D"]
  these appear in the paper but not in that section
  → **incorrect flagged**

Hallucination detection
Methods Considered

- Hybrid Hallucination Detection (HHD)

Step 1 — Entity extraction with SpaCy
- Identify "technical tokens": Capitalised words, acronyms, numbers…
- Ignore metaphorical or generic words → no false positives

Step 2 — Semantic support check
For each sentence:
- Find top-k most similar paper sentences (MiniLM embeddings)
- Check if each technical token appears in those contexts

Step 3 — Hallucination rule
A token is hallucinated only if:
- It does NOT appear in any top-k similar paper sentences
- AND the similarity < 0.6

Evaluation examples

Hermes, the messenger god of ancient Greece, was known for his speed and efficiency. Similarly, the HERMES system acts as a swift messenger between the initial prompt and the final, refined medical image segmentation.

→ it is a consistent creative simile. **Incorrect detection.**

The proposed solution, is an automated framework designed to enhance the accuracy of flash memory (FM)-based segmentation.

→ In the paper, FM stands for Foundation Model, not Flash Memory. **Incorrect detection.**

Hallucination detection
Methods Considered

Need a metric for detecting hallucinations in creative storytelling tasks in the world of scientific research.

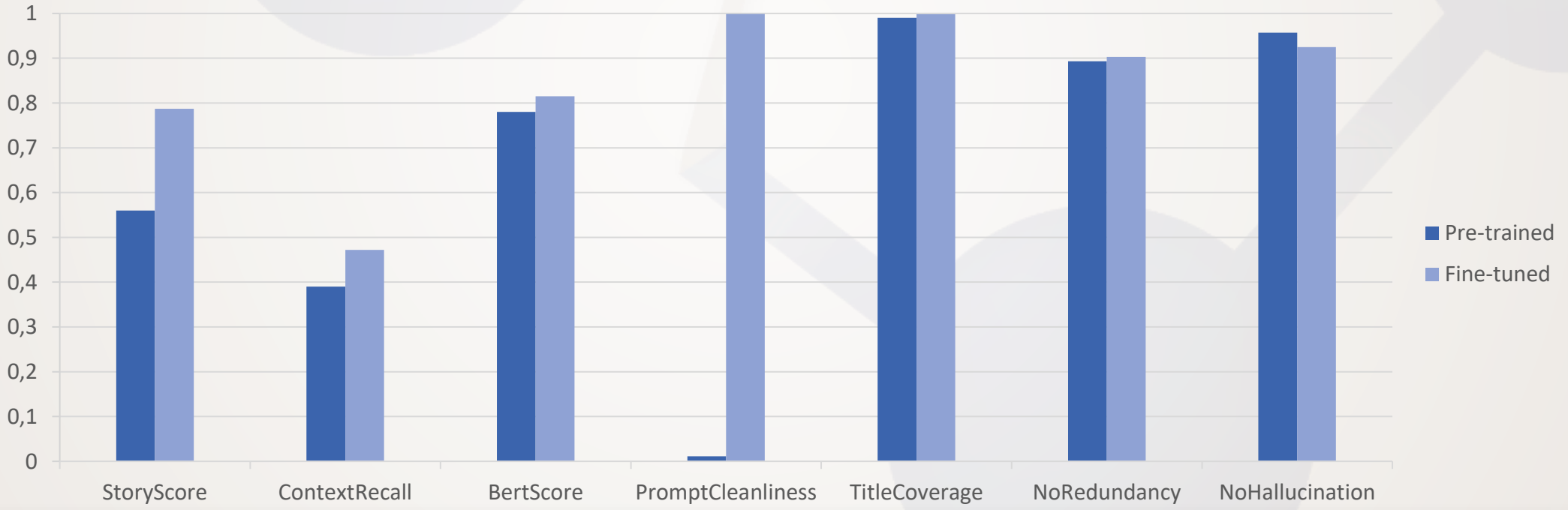
↓

Kept the simplest and most effective detector (NER) with SpaCy

METHOD	Capitalised Words	SpaCy NER	MIRAGE	LLM-Judge (Qwen 7B)	LLM-Judge (GPT 5.1)	HHD (hybrid)
WHAT IT DETECTS	Surface-form mismatch	Incorrect PERSON/ORG entities	Rewrite-consistency instability	Factual consistency	High-level reasoning errors	Entity + retrieval alignment
KEY WEAKNESS	Flags abbreviations & creative capitalisations as errors	Misses conceptual errors (wrong claims, invented datasets)	Penalises analogies and audience-adapted rephrasing	"Hallucinates hallucination" labels correct facts as errors	Overcautious: flags benign contextual expansions	Dominant false positives; threshold unstable
VERDICT	Too noisy	✓ Chosen	Too rigid	Unstable	Too strict	Unreliable

Preliminary Evaluation on a story set

- 76 generated stories (Pre-trained vs Fine-tuned pipeline)
- Fine-tuning strongly improves overall StoryScore
- The biggest gain is Prompt Cleanliness: prompt leakage is essentially removed
- Better human readability confirmed by StoryScore: 0.560 → 0.787



Conclusion and Future Works

- These metrics provide approximations rather than ground-truth guarantees
- Improve grounding mechanisms and hallucination mitigation
- Scale up evaluation with larger and more diverse evaluator

Thank you for your attention

Repository
bit.ly/ai-sci-storyteller

Mail
alex.argese@eurecom.fr