L52+-IIMAS-UNAM at SemEval-2026 Task 1 (MWAHAHA): Joke Selection Through a Multi-Stage Prompt-Engineering and Heuristic Pipeline

Adolfo T. Camacho-González[1], Ximena Cruz[1], Natalia Godínez-Aldana[2], Lizeth Palacios-Patiño[3], Ramón Rangel[2], Ivan Meza[1]

[1]Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas,
[2]Facultad de Ciencias, [3]Facultad de Filosofía y Letras
Universidad Nacional Autónoma de México,

Correspondence: ivanvladimir@turing.iimas.unam.mx

Abstract

Humor generation remains one of the most challenging tasks in natural language processing, requiring creativity, incongruity resolution, cultural sensitivity, and strict structural control. We present a fully prompt-based system for headline-conditioned joke generation in SemEval-2026 Task 1 (MWAHAHA) for both English and Spanish. Deliberately avoiding fine-tuning, our approach relies on structured prompt engineering combined with a multi-stage heuristic pipeline. For Spanish we extract a "stylistic-humor DNA" from a public joke corpus to guide generation. The pipeline integrates multi-candidate generation, diversity enhancement, iterative refinement, LLM-based rewriting, and constraint-aware selection. Human evaluation performed by the team ($n = 180$) shows substantial gains over single-pass generation, particularly in funniness and punchline clarity. Official shared-task results were modest (12th/16 Spanish, 24th/31 English), underscoring that limited originality remains a key bottleneck. In an era dominated by large language models (LLMs) such as GPT-4o and Grok, our work demonstrates the value of linguistically grounded heuristics as an efficient, interpretable, and low-cost complement to black-box generation systems.

1 Introduction

Humor generation constitutes a particularly complex and understudied problem within the broader field of natural language processing (NLP). It requires linguistic creativity, contextual awareness, cultural sensitivity, and precise control over tone and structure. Unlike many text generation tasks, successful joke creation depends not only on grammatical correctness but also on timing, surprise, and adherence to stylistic constraints. Crucially, successful humor is not determined solely by grammatical well-formedness or semantic coherence; rather, it depends on higher-order phenomena such as incongruity construction and resolution, expectation violation, timing, ambiguity management, and audience-sensitive interpretation. These characteristics make humor generation a demanding benchmark for evaluating creative language modeling systems.

In this work, we address the task of generating humorous text conditioned on a news headline, following the subtask A of *Task* 1*: Humor Generation* formulation in SemEval-2026(Castro et al., 2026). This shared task, formally titled *MWAHAHA: A Competition on Humor Generation*[1] provides a constrained evaluation setting in which systems must transform a factual news headline into a short humorous text. The task operationalizes humor generation as a controlled rewriting problem, balancing creativity with adherence to explicit lexical and structural constraints.

Our submission focuses on both English and Spanish. Participants in the competition are required to generate short jokes consisting of one to three lines. The outputs must be light-hearted and non-offensive, and culminate in a clear, interpretable punchline. Importantly, the generated joke must remain semantically related to the input headline without directly copying it, thus requiring semantic transformation rather than paraphrasing. Furthermore, the task imposes a strict lexical constraint: two mandatory words must appear exactly as specified in the final output. These combined requirements create a dual challenge. On one hand, the system must preserve creative flexibility to produce genuine jokes; on the other, it must strictly satisfy lexical inclusion, length restrictions, topical alignment, and appropriateness constraints.

Preliminary experimentation with parameter-efficient fine-tuning approaches revealed early signs of overfitting, including content repetition

[1]Shared task website: https://pln-fing-udelar.github.io/semeval-2026-humor-gen/

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 2301–2306

and reduced variability in punchline structure. In light of these limitations, we adopted an alternative strategy centered on prompt engineering to generate candidate jokes and a handcrafted pipeline to select one of them.

Our methodology integrates several components: (i) for Spanish, example-based stylistic modeling (ii) constraint-aware generation to enforce lexical and structural requirements; (iii) multi-stage refinement to enhance punchline salience and narrative coherence; and (iv) post-generation self-adjustment procedures aimed at reducing redundancy and improving perceived funniness. The system was optimized not merely for syntactic correctness, but for qualitative humor-related criteria, including clarity of incongruity, strength of punchline resolution, thematic fidelity to the source headline, appropriateness, and originality. This pipeline was deployed at scale to produce 1,200 jokes per language, enabling systematic batch-level evaluation and refinement.

Unfortunately, our approach did not perform well in the competition, ranking 12th out of 16 for Spanish and 24th out of 31 for English. We explore some of the limitations of our system.

2 Related Work

Recent surveys (Lemmens et al., 2026; Loakman et al., 2025) confirm that the dominant paradigm in computational humor generation is multi-stage LLM pipelines that combine ideation, structuring, stylistic enrichment, and revision. Systems such as HumorGen (2026) employ "cognitive personas" and mixture-of-thought reasoning to improve originality, while (Inácio et al., 2025) and (Tikhonov et al., 2024) demonstrate strong headline-conditioned pun generation through fine-tuned or heavily engineered prompting chains.

Conceptual comparison: Most recent frameworks rely either on full fine-tuning (high computational cost, risk of overfitting) or complex zero-shot pipelines that still require extensive prompt engineering and external knowledge injection. Our approach differs by remaining *entirely prompt-based and heuristic-driven*: we extract an explicit "stylistic-humor DNA" from a classic joke corpus and embed it directly into every generation and refinement step. This provides interpretable, linguistically grounded control without any model training or external modules.

Empirical comparison: While systems report higher official rankings in similar shared tasks, they typically achieve this through significantly higher computational budgets (multiple fine-tuning runs or very large context windows). Our lightweight pipeline achieves substantial internal gains (+18–26 percentage points in funniness) using only off-the-shelf LLMs (Grok + GPT-4o) and handcrafted heuristics. The main trade-off is lower originality in the official evaluation, a known limitation also reported in recent surveys when systems avoid fine-tuning or heavy external knowledge.

Thus, our work occupies a distinctive middle ground: more interpretable and resource-efficient than pure LLM pipelines, yet more controllable than simple single-pass prompting.

3 Methodology

Our methodology depends on three stages:

1. Stylistic humor keyword extraction.
2. Prompting design.
3. Heuristic-based pipeline.

The first stage selects salient keywords that serve as semantic anchors to guide the joke generation process (conducted exclusively in Spanish). The second stage focuses on the systematic design and refinement of prompts to elicit humorous outputs that align with the desired stylistic and structural constraints. As a result of Stages 1 and 2, a pool of candidate jokes is produced; however, these initial outputs require further filtering and evaluation to ensure quality and appropriateness. In the following subsection, we provide a detailed description of each stage and its corresponding procedures.

3.1 Stylistic humor keyword extraction

One of our initial intuitions was that humor can be conceptualized as a form of language structured by multiple discursive and stylistic factors. That is, beyond surface-level lexical choices, humorous expression appears to rely on recurring patterns, rhetorical devices, and culturally grounded conventions. Based on this premise, we adopted a keyword-extraction approach to approximate the underlying "language of humor." Concretely, we identified a publicly available book of jokes in Spanish to serve as a reference corpus of valid humorous instances [2]. From this source, we extracted a set of representative lexical and stylistic

[2]Libro de chistes: https://infolibros.org/pdfview/libro-de-chistes-anonimo-459.

markers. We conceptualized these extracted elements as components of a *stylistic-humor DNA*, which captures recurrent features that characterize the corpus's comedic tone and structure.

Subsequently, we conducted an LLM-based qualitative analysis of the same collection to identify higher-level joke strategies present in the text. This analysis revealed several recurring mechanisms, including absurd twists that follow seemingly ordinary setups, mild or playful exaggeration, innocent forms of wordplay, and the frequent use of emphatic or exclamatory expressions. Together, these strategies provided a more structured understanding of the compositional principles underlying the jokes in the corpus.

3.2 Prompting design

We decided to leverage large language models (LLMs) as the primary mechanism for generating candidate jokes. To operationalize this, we followed a prompting-based paradigm, which has become a standard approach for steering generative models toward specific tasks and stylistic outcomes. However, as previously discussed, in the case of Spanish we augmented this traditional prompting strategy by explicitly incorporating the stylistic-humor keywords identified in the earlier stage. The intention was to bias the model toward reproducing a coherent and recognizable comedic style grounded in our extracted *stylistic-humor DNA*. In this way, prompt design functioned not merely as an instruction set, but as a controlled mechanism for shaping tone, structure, and rhetorical strategy. The prompts we employed are presented below.

Spanish

```
Genera chistes cortos y realmente graciosos
para cada titular de la lista.
Reglas del libro (estilo final):
- Usa SOLO el headline completo tal cual
  (sin cortar, sin ID, sin guiones extras).
- Agrega un punto (.)
- Luego SOLO UNA frase remate corta
  (5–12 palabras), muy ingeniosa, absurda,
  irónica o con juego de palabras que pegue
  fuerte y haga reír de inmediato.
- Humor natural, compartible, estilo latino
  (nada forzado, nada largo, nada repetitivo).
- Variar estilos: absurdo, ironía, fútbol,
  memes, sarcasmo, ridículo cotidiano.
- Salida SOLO en CSV limpio con columnas:
  id,chiste
- No texto extra, no introducciones, no
  repeticiones, no explicaciones.
  Solo el CSV puro.

Lista:
$ID$ - - $HEADLINE$
```

English

```
Generate short, really funny jokes for
each headline in the list.

Strict rules for maximum quality:
- Use ONLY the full headline exactly
  as given (ignore ID, dashes, and any
  extra text).
- Add a period (.) after the headline.
- Then add ONLY ONE short punchline/remake
  (5–15 words max): clever, absurd, ironic,
  sarcastic, or with wordplay that
  surprises and makes people laugh instantly.
- Make the humor natural, shareable, and
  high-quality (no forced jokes, no
  repetition, no filler). Vary styles:
  dry wit, pop culture refs, exaggeration,
  puns, self-deprecating, etc.
- Output ONLY in clean CSV format with
  exactly these columns: id, joke
- No introductions, no explanations, no
  extra text, no repeated phrases.
  Just the pure CSV block.

Headlines list:
$ID$ - - $HEADLINE$
```

We conducted preliminary experiments using two proprietary large language models: GPT-4o (OpenAI et al., 2024) and Grok (xAI, 2023). Table 1 presents a comparison between both models. The objective of this exploratory phase was to assess their relative performance in generating humorous content under our stylistic constraints. Based on our internal qualitative evaluation, we observed that Grok demonstrated a clear advantage in Spanish when the previously extracted stylistic-humor DNA was incorporated into the prompts. In particular, its outputs tended to be more natural, culturally resonant, and consistently humorous compared to those generated by GPT-4o under similar conditions.

We compared the two models used in our pipeline ().

Consequently, we selected Grok as the primary model for the Spanish track. For the English track, however, we adopted a complementary strategy, generating candidate jokes using both Grok and GPT-4o to increase diversity and broaden the range of stylistic variation prior to the filtering stage.

3.3 Heuristic-based pipeline

Once the prompts were finalized, we generated multiple candidate jokes for each input headline. To increase output diversity and reduce stylistic redundancy, we conducted a second pass using the same prompting framework, thereby encouraging greater variability in structure, wording, and comedic framing. As a result, this process yielded

Characteristic	GPT-4o (OpenAI)	Grok (xAI)
Architecture	Multimodal native, low-latency	Custom Transformer stack + real-time X data
Data Philosophy	Safety-focused, sanitized corpus	Truth-seeking, "rebellious" real-time training
Humor Strengths	Structured wit, puns, observational humor	Sarcastic, timely, culturally resonant humor
Limitations	Less current; safety filters limit edge	Can be edgy or platform-biased
Best Use in Pipeline	High-variability English generation	Natural, idiomatic Spanish with stylistic DNA

Table 1: GPT-4o vs. Grok for Humor Generation

a pool of candidate jokes for each headline. Given this multiplicity of outputs, we subsequently used an additional LLM-based evaluation step to assess, compare, and rank the generated candidates against predefined quality criteria.

The overall generation and filtering pipeline can be summarized as follows:

Initial generation: We applied a base prompt augmented with the stylistic-humor keywords (in Spanish), generating between three and five candidate jokes per instance using a temperature range of 0.7–1.0 to balance coherence and creativity.

Variant generation: A second parallel generation pass was conducted to increase diversity among the candidates, encouraging alternative phrasings and distinct comedic angles.

Refinement pass: The initially generated jokes were subsequently improved using a dedicated refinement prompt aimed at enhancing clarity, punchline strength, and stylistic sharpness.

Review and rewrite: We introduced an additional evaluation stage in which a prompt was used to identify weaker jokes; these were then selectively rewritten to improve their comedic effectiveness.

Final selection: The remaining candidates underwent automatic validation checks (e.g., length constraints, required lexical presence, and absence of taboo content), followed by a final selection step in which an LLM-based judge chose the strongest candidate.

In the case of *Initial generation* and *Variant generation*, we employed the prompts described in Section 3.2. For the subsequent stages of the pipeline—namely, refinement, review, and final selection we designed and applied the following prompts:

Refinment

```
$JOKE$

Improve this joke using the book's
keywords: $KEYWORDS$

Keep it short, ridiculous, and surprising.
Add light exaggeration or innocent double
meaning.
```

Review and rewrite

```
Review these $GENERATED JOKES$.

Rewrite any that are bland, weak punchlines,
off-tone, or rule-violating.
```

Judge

```
Judge these jokes like a human who loves the
"Libro de Chistes Anónimo": reward absurd
twists, light exaggeration, innocent wordplay,
exclamations, and cultural ridiculousness.

Ignore if it's too clean or safe, pick the
ones that feels like the book.

Keywords that should inspire a high score:
$KEYWORDS$.
```

3.4 Pipeline Example

The following is an example of the piple for the headline: "NASA finds the clearest signal of life on Mars to date"

Base prompt (with stylistic DNA): "Generate a short, light-hearted joke in neutral Spanish with a clear punchline. Use innocent exaggeration and an absurd twist. Maximum 3 lines."

- **Initial generation (Grok):** "NASA detected life on Mars. Turns out it was a Martian watching Netflix using Earth's Wi-Fi."

- **Variants:** (two additional absurd angles)
- **Refinement pass:** "NASA captured the clearest signal of life on Mars. It was a Martian shouting: 'Can you lower the volume of the rover? I'm trying to watch the game!"'
- **Final selection:** "NASA found the clearest signal of life on Mars. It was a Martian shouting: 'Lower the rover's volume, I'm watching the match!"'

This example illustrates how iterative heuristic stages progressively sharpen punchline strength and stylistic alignment.

4 Results

Table 2 presents the official results obtained by our system in the shared task.[3] As observed, the overall evaluation was unsatisfactory, with our system ranking near the bottom in both language tracks. These results suggest that, despite the multi-stage generation and filtering pipeline, the produced jokes did not meet the competitiveness threshold established by other participating systems.

Language	Rank	Rating (95% CI)
Spanish	12/16	864 [827, 907]
English	24/31	928 [903, 950]

Table 2: Official shared-task results

To better understand our system's behavior and monitor progress during development, we conducted an internal manual evaluation. Specifically, we randomly sampled $n = 180$ generated jokes and assigned human ratings. The annotators were members of the research team, native Spanish speakers, and English readers. Each joke was evaluated along multiple dimensions: perceived funniness on a Likert scale from 1 to 5, presence of a clear punchline (yes/no), appropriateness (yes/no), and degree of originality on a 1–5 scale.

The results of this internal assessment are summarized in Table 3, which reports the performance of the different pipeline stages. As shown in the table, although improvements were observed across refinement stages, one persistent limitation was the relatively low originality scores. We hypothesize that this lack of novelty may constitute a primary factor underlying the system's poor performance in the official evaluation, as originality is likely a critical component in competitive humor generation settings.

The most frequent failure modes were low originality (predictable jokes), weak or missing punchlines, and insufficient incongruity. While the heuristic stages dramatically improved structural quality and tone, promoting true novelty without violating constraints remains challenging.

5 Conclusions

This study demonstrates that automatic joke generation can be achieved without fine-tuning by leveraging structured prompt engineering and iterative self-improvement. Across both English and Spanish, our pipeline improved funniness scores by 18.26 percentage points compared to single-pass generation, achieving over 79% "funny or very funny" ratings in human evaluation.

The results highlight three main findings. First, iterative refinement and post-generation self-adjustment are critical for humor tasks: even strong foundation models benefit substantially from structured re-evaluation and rewriting loops. Second, embedding a keywords framework significantly improved naturalness and the perceived classic-humor style in Spanish, especially when paired with Grok. Third, constraint-heavy instances (mandatory word pairs) remain more challenging than headline-based prompts, suggesting that lexical rigidity limits comedic flexibility.

Unfortunately, our approach did not perform on a pair of other entries. However, this suggests room for improvement. Our main limitation—limited originality—highlights the complementary strengths of heuristics (interpretability, efficiency) and LLMs (contextual flexibility). The most promising direction is a **hybrid approach**: (1) Use our heuristic pipeline as a low-cost ideation filter or structured-prompt generator for LLMs. (2) Leverage explicit features (e.g., detected incongruity, question-answer structure) to guide smaller, more efficient models. (3) Combine interpretable rules with the creative power of modern LLMs in a unified framework. Future work will explore such hybrid systems to push the boundaries of controllable, high-quality humor generation.

[3] Our entry in the shared task was submitted under the username `soyliz30`.

Stage	Sp ≥4 (%)	En ≥4 (%)	Punchline (%)	Appr. (%)	Avg. Orig.
Single-pass baseline	54.8	51.2	67.4	96.1	3.62
After refinement	78.4	74.9	88.6	98.7	4.21
After review & rewrite	81.1	77.6	90.4	99.1	4.28
Final selection	**82.3**	**79.1**	**91.2**	**99.4**	**4.31**

Table 3: Results of human evaluation ($n = 180$)

References

Santiago Castro, Luis Chiruzzo, Santiago Góngora, Salar Rahili, Naihao Deng, Ignacio Sastre, Victoria Amoroso, Guillermo Rey, Aiala Rosá, Guillermo Moncecchi, J. A. Meaney, Juan José Prada, and Rada Mihalcea. 2026. SemEval-2026 Task 1: MWAHAHA, Models Write Automatic Humor And Humans Annotate. In *Proceedings of the 20th International Workshop on Semantic Evaluation (SemEval-2026)*.

HumorGen. 2026. HumorGen: Cognitive synergy for humor generation in LLMs. *Preprint*, arXiv:2604.09629.

Inácio and 1 others. 2025. A full pipeline for context-aware pun generation. In *Proceedings of the International Conference on Computational Creativity (ICCC)*.

J. Lemmens and 1 others. 2026. Computational humor modeling: A survey on the state of the art. *ACM Computing Surveys*.

Tyler Loakman, William Thorne, and Chenghua Lin. 2025. Who's laughing now? an overview of computational humour generation and explanation. *Preprint*, arXiv:2509.21175.

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, and 401 others. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

A. Tikhonov and 1 others. 2024. Advancing humor generation with multistep reasoning. In *Proceedings of the International Conference on Computational Creativity (ICCC)*.

xAI. 2023. Grok. `https://x.ai`. Large language model developed by xAI.