GenSIE: General-purpose Schema-guided Information Extraction

Task Proposal for IberLEF 2026

Yudivian Almeida Cruz Suilan Estévez Velarde
Alejandro Piad Morffis Isabel Espinosa Zaragoza
María Miró Maestre Alba Pérez Montero
Lucía Sevilla Requena Ernesto Estevanell Valladares

This proposal introduces GenSIE (General-purpose Schema-guided Information Extraction), a novel task for IberLEF 2026 designed to evaluate the ability of systems to extract nested, structured information (JSON) from general-domain Spanish texts. A key challenge is the zero-shot schema adherence, where the extraction schema is provided only at inference time. Focusing on small, open-weight language models, GenSIE aims to stimulate methodological innovation in inference-time techniques. We will construct a high-quality, human-curated dataset of 1,000 annotated examples across diverse domains, strictly enforcing grounding to penalize hallucinations. The task emphasizes structural validity and semantic accuracy, with evaluation based on a Flattened Schema Scoring metric. GenSIE seeks to establish a transparent and reproducible benchmark for robust, cost-effective, and sustainable structured information extraction in Spanish.

1 Introduction and Motivation

The field of Information Extraction (IE) has traditionally been dominated by task-specific models trained on fixed schemas (e.g., classic Named Entity Recognition or Relation Extraction with pre-defined classes). However, the advent of Large Language Models (LLMs) has shifted the paradigm toward generative extraction, where models are expected to extract complex, structured information based on natural language instructions and dynamic schemas.

While LLMs excel at generation, they often struggle with structural reliability (outputting invalid JSON) and zero-shot schema adherence (extracting fields they have not seen during training). Furthermore, rigorous benchmarks for this capability in Spanish are scarce, as most "instruction-following" benchmarks focus on reasoning or creative writing rather than strict structured data extraction.

The rise of Agentic Workflows has created a massive demand for systems that can communicate via structured protocols. To identify user intent, invoke external tools (API calls), or exchange information between autonomous sub-agents, an AI must be able to output rigid, error-free structured data (JSON) across a vast variety of dynamic formats. However, these agentic workflows are computationally intensive. A single user request often triggers multiple inference steps—reasoning loops, self-correction, and tool execution—multiplying the cost and latency of the system. Relying exclusively on massive, proprietary models for these "control flow" operations is often cost-prohibitive and inefficient. This reality creates an urgent need for Small Language Models (SLMs) that can run on commodity hardware yet still perform complex structured extraction with high reliability.

Our proposal explicitly targets the evaluation of Small Language Models (sub-14B parameters) within an Open Source ecosystem for several strategic reasons. First, while massive proprietary models (like GPT-5) might solve many extraction tasks through raw scale, SLMs often require clever engineering to perform at the same level. This creates a valuable *innovation gap* where participants must explore inference-time techniques—such as Chain-of-Thought (CoT), ReAct loops, and self-consistency ensembles—to boost performance. By focusing on this gap, we aim to stimulate methodological innovation rather than simply rewarding whoever has access to the largest computational budget.

Furthermore, we aim to prioritize efficiency and sustainability to ensure that high-performance extraction pipelines remain deployable in real-world scenarios. By focusing on models that run on consumer-grade hardware, we promote sustainable AI and cost-effective solutions that are accessible to smaller research groups and industry practitioners. Finally, this approach safeguards reproducibility and sovereignty. Relying on closed-source, black-box APIs undermines scientific rigor, as these models often change silently behind the scenes. By standardizing on open weights, we ensure that results are reproducible, transparent, and permanent, fostering a stable foundation for future research.

In this context, we thus propose **GenSIE (General-purpose Schema-guided Information Extraction)**, a novel and challenging task for IberLEF 2026 evaluating the ability of systems to extract nested, structured information (JSON) from general-domain Spanish texts, given a schema that is only provided at inference time (Zero-Shot), based only on inference-time techniques, and with a focus on small, open-weight language models.

2 Antecedents and Previous Experience

GenSIE is not a standalone initiative but the evolution of a successful line of research and evaluation campaigns organized by this consortium. It directly extends the **eHealth-KD** (eHealth Knowledge Discovery) challenges, which ran at **IberLEF/TASS from 2018 to 2021**.

The eHealth-KD series focused on the automatic extraction of semantic information from Spanish electronic health documents. Over four editions, it established a robust benchmark for extracting:

- **Entities:** Key concepts (Concepts, Actions, References, etc.).
- **Relations:** Semantic links between entities (Causality, Property-of, Same-as, etc.).

GenSIE represents a paradigm shift designed for the era of Generative AI, extending the eHealth-KD methodology in three critical dimensions:

1. **From Domain-Specific to General Purpose:** While eHealth-KD was strictly medical, GenSIE embraces a general domain scope (Legal, Scientific, News, Technical), requiring systems to be robust across widely varying vocabularies and contexts.
2. **From Fixed to Dynamic Schemas:** eHealth-KD relied on a static ontology defined by the organizers. GenSIE introduces a **Zero-Shot Schema** challenge, where the equivalent of an "ontology" is provided at inference time via a JSON Schema. This mirrors real-world applications where an LLM agent must adapt to new APIs or data formats on the fly.
3. **From Structural Tagging to Generative Structure:** Previous tasks **were not** typically solved with sequence labeling (BIO tagging). GenSIE demands the generation of syntactically valid, deeply nested JSON objects, shifting the challenge from simple classification to **constrained generation and structural reasoning**.

GenSIE thus stands on the shoulders of this extensive track record. It leverages our proven expertise in the manual annotation and rigorous evaluation of complex semantic corpora, adapting these established methodologies to address the stochastic nature, structural demands, and infinite variability characteristic of the Generative AI era.

3 Task Description

The objective of GenSIE is to extract structured knowledge from a given text fragment according to a specific, arbitrary schema provided in the input.

Crucially, the extraction schemas in GenSIE are designed to elicit a high level of semantic analysis and reasoning. This task goes beyond the mere retrieval of surface-level strings; it requires the model to synthesize and interpret the text to satisfy the schema constraints. For example, in a legal domain context, a schema might request a boolean field indicating whether a lawsuit was `successful` or `failed`. Correctly populating this field requires the model to understand the verdict and its implications, rather than simply copying a substring. To rigorously evaluate this capability, our dataset includes tasks with varied levels of semantic complexity, ranging from direct entity extraction to such high-level inferential fields.

3.1 Grounding & Hallucination Traps

A critical aspect of GenSIE is evaluating the model's ability to remain strictly grounded in the source text. To test this, we will design specific schema fields that ask for information **not present** in the input context—even if that information is widely known (e.g., asking for the specific date of a famous historical event when the text only mentions the year). In such cases, the system must explicitly return a `null` value. The frequency of these "null" targets will be kept intentionally low to prevent systems from artificially inflating their scores by defaulting to empty outputs, but their presence is vital for penalizing parametric hallucinations. This design enforces a strict "retrieval-only" behavior essential for reliable, trustworthy downstream applications.

3.2 Input

For each instance, the system receives:

1. **Context:** A text fragment in Spanish (sourced from Wikipedia, news, scientific abstracts, or blogs).
2. **Instruction:** A natural language description of what needs to be extracted.
3. **Target Schema:** A JSON Schema definition following a **strict, non-recursive subset of the OpenAPI 3.0** specification (serialized via Pydantic). This restriction ensures compatibility with standard grammar-constrained decoding techniques while providing clear definitions of types, enums, and required fields.

3.3 Output

The system must generate a **valid JSON object** that:

1. Strictly adheres to the provided `Target Schema`.
2. Contains information faithfully extracted from the `Context`.
3. **Does not** hallucinate information not present in the text (returns `null` for missing data).

3.4 Example Instance

To illustrate the task, consider the following example sourced from Wikipedia:

Input Context:

> “El ensayo clínico aleatorizado de Fase 3 de la vacuna mRNA-1273 (Moderna) evaluó a 30,420 participantes. Los resultados primarios mostraron una eficacia del 94.1% en la prevención de la enfermedad sintomática por COVID-19 en comparación con el placebo. Los eventos adversos solicitados fueron principalmente leves o moderados, incluyendo dolor en el sitio de inyección, fatiga y cefalea, resolviéndose en 2-3 días.”

Input Instruction:

> “Extrae el nombre del medicamento, su categoría, el tamaño de la muestra, la eficacia reportada, los efectos secundarios, clasifica el resultado clínico del ensayo y extrae la temperatura de almacenamiento requerida.”

Input Target Schema:

```
{
  "type": "object",
  "properties": {
    "medication_name": {
      "type": "string",
      "description": "Name of the drug or vaccine"
    },
    "medication_category": {
      "type": "string",
      "description": "Type of medication (e.g. vaccine, analgesic)"
    },
    "sample_size": {
      "type": "integer",
      "description": "Total number of participants"
    },
    "efficacy_rate": {
      "type": "string",
      "description": "Extract verbatim percentage"
    },
    "side_effects": {
      "type": "array",
      "items": { "type": "string" },
      "description": "List of reported adverse events"
    },
    "clinical_outcome": {
      "type": "string",
      "enum": ["POSITIVE", "NEGATIVE", "INCONCLUSIVE"],
      "description": "Infer the semantic success of the trial"
    },
    "storage_temperature": {
      "type": ["string", "null"],
      "description": "Required storage condition"
    }
  },
  "required": [
    "medication_name",
    "medication_category",
    "sample_size",
    "efficacy_rate",
    "side_effects",
    "clinical_outcome",
    "storage_temperature"
  ]
}
```

Expected Output (Gold Standard):

```
{
  "medication_name": "mRNA-1273 (Moderna)",
  "medication_category": "vacuna",
  "sample_size": 30420,
  "efficacy_rate": "94.1%",
  "side_effects": [
    "dolor en el sitio de inyección",
    "fatiga",
    "cefalea"
  ],
  "clinical_outcome": "POSITIVE",
  "storage_temperature": null
}
```

This instance demonstrates two critical challenges beyond simple entity recognition:

1. **Semantic Reasoning:** The `clinical_outcome` field requires the model to interpret the narrative ("94.1% efficacy", "prevention of disease") and map it to the specific enum `POSITIVE`, a label that never appears explicitly in the text.
2. **Hallucination Check (Grounding):** The schema asks for `storage_temperature`, a well-known fact for the Moderna vaccine (-20°C). However, this information is absent from the provided context snippet. A reliable model must suppress its pre-trained parametric knowledge and correctly return `null`, whereas a hallucinating model might incorrectly fill in the external fact, failing the grounding constraint.

3.5 The Zero-Shot Challenge

A crucial aspect of the GenSIE challenge is that it is a **zero-shot schema** task. In the **Development Phase** participants will be given examples with a set of schemas (e.g., `Event`, `Biography`, `Recipe`). However, in the **Test Phase**, the private evaluation set will contain entirely new schemas (e.g., `LegalContract`, `MedicalProcedure`, `ProductSpec`) that the model has not seen in the training data, along some schemas that were present in the development set. This forces participants to build systems that can generalize to *any* structure, rather than overfitting to specific entity types.

4 Methodology and Resources

4.1 Dataset Construction (Human-in-the-Loop)

We will employ a rigorous Human-in-the-Loop (HitL) methodology to construct a high-quality Silver-to-Gold dataset. This process begins with the curation of diverse source texts from high-quality Spanish repositories, including Wikipedia, news sources, public domain books, and open-access scientific journals. We target coverage of approximately 10 distinct domains (e.g., Legal, Medical, Scientific) paired with around 100 unique extraction schemas.

Once sources are selected, a commercial LLM (e.g., Gemini 3 Pro) will generate initial “Silver” annotations. These preliminary annotations will then undergo a strict double-blind review process.

Inter-Annotator Agreement & Quality Control:

To guarantee the objective quality of the Gold Standard, we will initially generate a pool of candidate instances significantly larger than the target 1,000. Each instance will be reviewed and corrected by **two independent human annotators**. We will verify consistency by calculating an Inter-Annotator Agreement (IAA) score using a metric analogous to the *Flattened Schema Scoring* defined in Section 5.

Only examples that achieve high agreement between annotators will be retained. Conflicting examples will be discarded or resolved by a third senior annotator. We commit to generating as many initial candidates as necessary to ensure the final dataset contains exactly 1,000 high-quality, validated examples.

Finally, we will strictly avoid any source text containing Personally Identifiable Information (PII). All data will be sourced from public domain or appropriately licensed repositories, with text fragments kept as short snippets consistent with Fair Use principles.

4.2 Data Distribution

The final dataset will consist of **1,000 high-quality** annotated examples. To specifically test generalization capabilities, the data will be split into a **Development Set of 200 examples** and a blinded **Test Set of 800 examples**.

To facilitate early engineering and robust pipeline development, we will adopt a two-stage release strategy:

- **Stage 1 (Starter Kit):** On **March 1, 2026**, we will release a "Starter Kit" containing the official evaluation scripts, Docker templates, and a small subset of **30 examples** from the Development Set. This allows participants to build their containers, debug connection logic, and validate their baselines a full month before the official start.
- **Stage 2 (Full Development):** On **April 1, 2026**, the remaining 170 examples of the Development Set will be released. This phase focuses on refining prompts, RAG strategies, and schema generalization.

Crucially, the split is designed to evaluate developer-aware generalization: specific domains and schemas present in the Test Set will be completely absent from the Development Set. This "hold-out" strategy forces systems to handle unseen vocabularies and structures without prior fine-tuning, simulating real-world deployment where an agent encounters novel tasks.

4.3 The "No-Training" Policy & Data Augmentation

To rigorously evaluate low-cost, highly transferable techniques that function effectively in low-resource domains, this challenge enforces a strict **No-Training Policy**. We purposefully **do not** release a large training dataset; instead, participants must rely on the small Development Set to calibrate their systems. Furthermore, in test time, we will provide API access to blind models, which means participants **cannot** tune their pipelines to specific models, or train LoRA adapters.

This design is intended to stress-test the participants' ability to leverage *inference-time techniques* (e.g., ReAct, Chain-of-Thought, retrieval) rather than relying on gradient updates.

However, while fine-tuning the model weights is prohibited, participants are permitted to use the Development Set to generate synthetic data or apply augmentation techniques to support their inference pipelines (e.g., creating synthetic examples for dynamic few-shot retrieval). In such cases, to maintain scientific transparency, any team utilizing data augmentation must report the methodology and publish the augmented datasets alongside their code.

4.4 Post-Challenge Availability

To foster open science and ensure the longevity of the benchmark, all resources will be made publicly available after the competition concludes. This includes the release of the complete dataset (Training, Development, and blinded Test sets) under a standard Creative Commons license. Furthermore, we will provide detailed annotation metadata, granting participants access to domain tags and complexity scores for each instance.

This will enable researchers to perform detailed ablation studies across domains and task modalities, allowing for a granular analysis of their models' strengths and weaknesses. Finally, the full evaluation suite and baseline implementations will be released, ensuring that GenSIE remains a transparent and reproducible standard benchmark for Spanish structured information extraction long after the IberLEF 2026 campaign.

5 Evaluation Metrics

Evaluating generative structured output requires a rigorous approach that assesses both **structural validity** and **semantic accuracy**. We define a **Flattened Schema Scoring** metric formalized as follows:

5.1 JSON Flattening

Let J be a JSON object. We define a flattening function $\Phi(J)$ that transforms the nested structure into a set of key-value pairs, where keys represent the full path to the value.

$$\Phi(J) = \{(k_1, v_1), (k_2, v_2), ..., (k_n, v_n)\}$$

Example: `{"event": {"city": "Madrid"}}` becomes `{ "event.city": "Madrid" }`.

Let $G = \Phi(J_{gold})$ be the set of pairs in the Gold Standard, and $S = \Phi(J_{sys})$ be the set of pairs in the System Output. The set of matching keys is defined as $K = \text{keys}(G) \cap \text{keys}(S)$.

5.2 Value Scoring Function

For a given path key $k \in K$, let g_k be the gold value and s_k be the system value. The score $Sim(g_k, s_k)$ is determined by the semantic type of the field defined in the schema:

Case A: Rigid Types (Numbers, Dates, Categorical Strings): Our schema design philosophy explicitly offloads strict semantic reasoning (e.g., verdicts, diagnoses, classifications) to **Rigid Types** such as Booleans and Enumerations. For these fields, precision is binary; there is no "partial credit" for a wrong classification.

$$Sim(g_k, s_k) = \mathbb{I}(g_k = s_k)$$

(Returns 1 if exact match, 0 otherwise).

Case B: Free Text (Descriptions, Summaries) Conversely, we reserve **Free Text** fields for content that is primarily extractive or descriptive (e.g., almost verbatim product descriptions, summaries, or snippets). In these specific cases, the goal is to capture the syntactical surface form rather than to perform complex semantic reasoning (which is handled by the Rigid Types). Therefore, a hybrid metric combining standard semantic embeddings with lexical overlap is sufficient and robust.

$$Sim(g_k, s_k) = \alpha \cdot \text{CosSim}(\mathbf{e}_{g_k}, \mathbf{e}_{s_k}) + (1 - \alpha) \cdot \text{Lexical}(g_k, s_k)$$

Where: - $\mathbf{e}_x$ is the vector representation from a standard multilingual sentence transformer (e.g., `paraphrase-multilingual-mpnet-base-v2`). - $\text{Lexical}(g_k, s_k)$ is a normalized BM25-based or token-overlap score. - α is a weighting hyperparameter (e.g., 0.7).

(Exact parameters and relevant metric details will be provided upon release of the development set.)

5.3 Aggregated Metrics

We calculate the total **True Positive Score (TPS)** by summing the similarities of shared keys:

$$TPS = \sum_{k \in K} Sim(G[k], S[k])$$

The final metrics are defined as:

$$Precision = \frac{TPS}{|S|} \quad , \quad Recall = \frac{TPS}{|G|}$$

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

5.4 Evaluation Tools

To facilitate development and ensure transparency, the organization will provide official evaluation scripts in Python alongside the release of the Development Set. These scripts will contain the exact parameters used for the official ranking (including the value of α and the specific sentence transformer model), allowing participants to locally test and validate their systems against the provided metrics.

6 Submission and Evaluation Environment

To ensure reproducibility and standardized hardware conditions, the evaluation will differ from typical text-submission tasks.

6.1 Submission Guidelines

Participants must submit a **GitHub repository** containing their full system.

- **License:** The repository must be open-sourced under a permissive FOSS license (e.g., **MIT, Apache 2.0**). Copyleft licenses (e.g., GPL) are **not** accepted to facilitate broader adoption.
- **Repository Privacy & Release Cycle:** To ensure fair play and independent development, submitted repositories must remain private during the competition phase. Teams will be required to grant read access to a designated member of the organizing committee for evaluation purposes. However, to uphold the principles of Open Science, all repositories must be made publicly available immediately after the release of the official results.
- **Enforcement & Cleanup:** Public availability of the code is a strict prerequisite for the acceptance of the final camera-ready version of the team's technical paper. The period between the results announcement and the camera-ready deadline serves as a grace period, allowing teams to refine their code, improve documentation, and polish their repositories before they become public artifacts.
- **Reproducibility:** The repository must contain all code, prompt templates, and auxiliary data (including any few-shot examples or synthetic data generated by the team) necessary to reproduce the results.
- **Documentation (README):** The repository must include a comprehensive `README.md` that details:
  - The technical setup of the pipeline.
  - Specific runtime requirements (e.g., mounting volumes, environment variable dependencies).
  - **Pipeline Selector:** If the team submits multiple pipelines, the README must clearly explicitly instruct how to run each specific pipeline (e.g., via different launch scripts or configuration flags).
- **Containerization & Isolation:** A `Dockerfile` must be included to build the submission image. Crucially, the resulting container must be self-sufficient, containing all necessary code, dependencies, and auxiliary data required for inference. The evaluation environment is completely isolated, *granting no access to the internet or the external filesystem.* Teams are fully responsible for ensuring their Dockerfiles build and run correctly under these constraints. To facilitate this process, the organizers will provide

a template Dockerfile (validated with the baseline systems) that demonstrates how to correctly configure the environment variables and handle the isolated execution context.
- **Multiple Pipelines:** To stimulate innovation and allow for the exploration of different techniques, each team is permitted to submit *up to three distinct pipelines*.
  - Teams can freely experiment (e.g., submitting one robust, complex pipeline and another experimental, highly efficient one) without fear of penalty.
  - The final **Team Ranking** will be based on the highest score obtained by any of their submitted pipelines (i.e., the best-performing configuration).

6.2 Inference Environment

The Docker containers will be executed in an isolated environment with no internet access. Participants are free to use any programming language or technology (e.g., vector databases for RAG), provided they run within the container.

Model Access: Since participants **cannot** download models at runtime, the system must connect to an external OpenAI-compatible inference server hosted by the organizers.

- The system will receive connection details via environment variables: `BASE_URL` and `MODEL`.
- **No Fine-Tuning:** The inference environment **does not** support loading external adapters (LoRA/PEFT) or soft prompts. All adaptations must be provided via the text prompt (context) or RAG mechanisms within the pipeline.
- This architecture ensures that all teams are evaluated using the exact same underlying LLMs, focusing the competition on the quality of the extraction pipeline (prompting strategies, RAG, schema parsing) rather than the raw power of a private model.

6.3 Technical Infrastructure & Feasibility

To guarantee a smooth and robust evaluation process for the “Hosted Inference” architecture, the organizers have secured the necessary computational resources and established clear operational boundaries.

- **Hardware Capacity:** The evaluation will be powered by a high-performance computing cluster equipped with 8x NVIDIA A100 (40GB) GPUs. This compute budget allows us to handle the high throughput required for evaluating multiple generative pipelines simultaneously.
- **Resource Quotas:** To ensure fair access and prevent denial-of-service scenarios, we will establish a reasonable maximum input token budget per submission (e.g., 32K tokens across all inference calls for each specific input example). Furthermore, to prevent stalled processes and infinite loops, a strict wall-clock timeout (e.g. 60 seconds) per test instance will be enforced. The exact parameters will be published in due time.

- **Qualification Phase:** To ensure the stability of the evaluation infrastructure and the scientific relevance of the results, all submissions must pass a Qualification Phase before entering the final evaluation. This phase involves running the submitted container on a subset of the Development Set to verify two conditions:
    1. **Technical Robustness:** The container must execute without hanging, crashing, or exceeding the timeout limits.
    2. **Baseline Improvement:** The system must demonstrate performance superior to the provided zero-shot baseline. Submissions that fail this sanity check will be rejected, protecting the evaluation cluster from inefficient or broken pipelines.
- **Baseline Robustness:** We recognize that connecting to a remote inference server can be technically challenging in a containerized environment. To mitigate this, the template Dockerfiles and baseline implementations provided by the organizers come pre-configured with robust connection logic. This includes built-in handling for retries, exponential backoff, and timeouts, ensuring that participants can focus on their extraction logic rather than debugging network protocols.

6.4 Rankings and Awards

We will evaluate all submitted systems across multiple models (including the 3 public baselines and additional "surprise" models to test agnosticism), producing two distinct leaderboards.

Main Leaderboard (Performance): The primary goal of the competition is to identify the most effective extraction systems regardless of computational cost. Since participants are restricted from fine-tuning or modifying the underlying models, this leaderboard explicitly serves as a benchmark for Inference-Time techniques. We strongly encourage participants to exploit the full reasoning capabilities of the LLMs using strategies such as Chain-of-Thought (CoT), ReAct loops, Ensembling / Self-Consistency, and multi-turn refinement.

- **Scoring:** Systems will be ranked based on their **Average F1 Score** across all evaluated models.
- **Team Ranking Strategy:** The final ranking will be determined by the submission (pipeline) that maximizes this Average F1 Score. We reward the "best generalist submission" per team, ensuring that the winning solution is not just optimized for a single LLM but demonstrates robust, model-agnostic performance.

Efficiency Leaderboard (Secondary): In parallel, we will maintain a secondary leaderboard to recognize and reward sustainable engineering and cost-effective solutions. For this metric, we will measure the Total Token Consumption for each system on the inference server. This is rigorously defined as the sum of all input and output tokens generated across all API calls required to solve a single instance, including any intermediate reasoning steps, self-corrections, or retries.

Teams that surpass the baseline F1 scores will be eligible for this ranking, which calculates a **Performance-to-Cost Ratio** (Average F1 divided by Total Token Count). This metric highlights systems that achieve high accuracy with minimal token usage—favoring efficient zero-shot prompting over token-heavy multi-step agents—and underscores the importance of economic viability in real-world deployments.

Additionally, we will release energy cost estimates (kWh) for all submissions, enabling authors to include sustainability metrics in their technical reports.

6.5 System Description Papers

Following the submission deadline on **May 8, 2026**, the official Gold Standard Test Set (inputs and annotations) will be immediately released to all participants.

This early release strategy allows teams to perform detailed error analysis and ablation studies on their local machines *while* the organizers conduct the official blinded evaluation. By the time the Official Leaderboards are released on **May 31**, participants will have had three weeks to analyze their model's behavior, ensuring that the System Description Papers (due **June 7**) go beyond simple metrics and offer deep insights into structural failures and reasoning gaps.

To support the community in producing high-quality publications, the organizing committee will conduct a collaborative review process. We will provide a round of constructive feedback on the submitted drafts, helping participants refine their methodology descriptions and strengthen their analysis prior to the camera-ready deadline.

7 Baselines

To demonstrate feasibility and provide a reference point for "Tiny", "Small" and "Medium" Language Models, the organizers will provide three open-source baselines. We explicitly target models accessible to research groups with limited compute (i.e., consumer-grade GPUs):

The baselines will be built using a standardized zero-shot prompting strategy. To ensure structural validity, we will employ grammar-constrained decoding (enforcing the output to strictly match the JSON schema). We will provide the full source code for this pipeline, including the prompt templates and constraint logic. Participants are encouraged to compare against this implementation or use it as a foundation for their own solutions.

Models: - **Tiny: Llama 3.2 3B Instruct** – Representing the state-of-the-art in "tiny" efficient models. - **Small: Salamandra 7b Instruct** – Developed by the Barcelona Supercomputing Center (BSC), this model represents the state-of-the-art in native Spanish language modeling, ensuring that our benchmarks reflect the specific linguistic nuances of the target language. - **Medium: Qwen 3 14b** – Representing the upper bound of consumer-hardware-friendly models with strong Spanish reasoning capabilities.

8 Organizing Committee

The organization is led by a consolidated consortium between the **Research Group on Artificial Intelligence and Data Science (GIA-UH)** at the University of Havana and the **Research Group in Natural Language Processing and Information Systems (GPLSI)** at the University of Alicante.

The committee comprises a balanced, multidisciplinary team of 8 members: five senior researchers and three PhD students. The composition also represents a strategic alliance between the fields of Computer Science and Linguistics, ensuring that the task is designed with both technical rigor in generative modeling and linguistic precision in corpus annotation.

Name	**Affiliation**	**Degree**	**Background**
Yudivian Almeida Cruz	University of Havana	PhD, Full-time Professor	Computer Science
Suilan Estévez Velarde	University of Havana	PhD, Full-time Professor	Computer Science
Alejandro Piad Morffis (*)	University of Havana	PhD, Full-time Professor	Computer Science
Isabel Espinosa Zaragoza	University of Alicante	PhD, Assistant Professor	Linguistics
María Miró Maestre	University of Alicante	PhD, Postdoctoral Researcher	Linguistics
Alba Pérez Montero	University of Alicante	PhD Student	Linguistics
Lucía Sevilla Requena	University of Alicante	PhD Student, Associate Professor	Linguistics
Ernesto Estevanell Valladares	University of Havana	PhD Student	Computer Science

(*) Corresponding author: (apiad@matcom.uh.cu; alepiad@gmail.com)

The organizers possess extensive experience in the management of competitive evaluations. Most notably, this core group successfully coordinated the eHealth-KD (Knowledge Discovery) shared tasks at TASS 2018 and IberLEF 2019 to 2021, managing the entire lifecycle from corpus annotation and guideline definition to the evaluation of participant systems. Beyond eHealth-KD, members of the committee have contributed to the organization of other relevant workshops and have published extensively on Human-in-the-Loop annotation methodologies, knowledge graph construction, and semantic evaluation.

9 Tentative Schedule

We propose the following schedule to ensure rigorous dataset construction and sufficient time for deep participant analysis. The timeline has been optimized to front-load engineering tasks and maximize the window for paper writing.

- **Dec 2025 – Jan 2026: Definition & Source Selection:** Finalization of extraction schemas and selection of diverse source texts (Wikipedia, News, Scientific Journals).
- **Jan 2026 – Feb 2026: Silver Data Generation & Curation:** Large-scale synthetic generation followed by strict human curation to create the Gold Standard. Priority will be given to finalizing the "Starter Kit" subset.
- **March 01, 2026: Starter Kit Release:** Release of the baseline code, Docker templates, and the 30-example Starter Kit. Participants begin engineering and environment setup.
- **April 01, 2026: Development Phase Start:** Release of the full Development Set (remaining 170 examples).
- **May 08, 2026: Submission Deadline:** Final deadline for participants to submit their Docker containers/repositories.
- **May 09, 2026: Test Set Release:** The Gold Standard Test Set is released to participants for local error analysis and paper writing.
- **May 09 – May 30, 2026: Evaluation Period:** Organizers run the submitted containers on the hosted infrastructure. This extended window includes buffer time for technical contingencies. *If results are available earlier, we will release them as soon as possible.*
- **May 31, 2026 (or earlier): Results Announcement:** Release of the official leaderboards.
- **June 07, 2026: Working Notes Deadline:** Submission of participant system description papers.
- **June 19, 2026: Notification & Feedback:** Acceptance notification and distribution of constructive feedback from the organizers.
- **July 01, 2026: Camera Ready Deadline:** Final deadline for the revised papers (including the overview paper). GitHub repositories must be public by this date.
- **September 22, 2026: Workshop Day:** Presentation of results at the IberLEF workshop (León, Spain).

10 Communication and Dissemination Strategy

To ensure broad participation and community engagement, the organization has designed a comprehensive dissemination plan leveraging our established networks and digital presence.

The organizing team manages a combined network of over 20,000 followers across key platforms including X (Twitter), LinkedIn, and Substack. We will execute a sustained communication campaign throughout the challenge lifecycle, featuring regular updates, baseline tutorials, and featured posts on the challenges of structured extraction. This constant stream of content is designed to build momentum, attract diverse teams from industry and academia, and keep participants engaged from the development phase through to the workshop.

We will leverage our strong institutional ties to maximize visibility within the Spanish-speaking research community. This includes direct promotion through the University of Havana (Cuba) and the University of Alicante (Spain), as well as the SEPLN (Spanish Society for Natural Language Processing) mailing lists. Additionally, we will activate our network of foreign collaborators across Europe and the Americas to ensure GenSIE attracts international attention, positioning it as a global benchmark for Spanish Information Extraction.