BUAP-NLP Lab at MentalRiskES 2026: Zero-Shot Prompting with Heterogeneous Large Language Model Selection for Early Detection of Mental Health Symptoms in Therapeutic Conversations

Alfredo Navarrete-Montes[1,*], Alejandro Amador-Lagunes[1] and Luis Yael Méndez-Sánchez[1]

[1]*Benemérita Universidad Autónoma de Puebla (BUAP), Puebla, México*

Abstract

This paper describes the participation of team BUAP-NLP Lab in the MentalRiskES 2026 shared task at IberLEF, focused on early detection of mental health symptoms from iterative therapeutic conversations in Spanish. We address Task 1 (prediction of psychometric questionnaire scores: GAD-7, PHQ-9, and CompACT-10 from the accumulated conversational history of each patient) and Task 2 (selection of the most clinically appropriate therapist response from three candidates). The proposed system implements a zero-shot client based on Large Language Models (LLMs) accessed via the OpenAI API, with no fine-tuning or additional training. Main contributions are: (i) a heterogeneous model selection strategy across the three official runs (`gpt-4o-mini`, `gpt-4.1-nano`, and `gpt-5-nano`) balancing cost, quality, and diversity; (ii) structured output enforcement via function calling, guaranteeing valid responses for both tasks; (iii) a local conversational history persistence mechanism enabling recovery from execution interruptions; and (iv) CodeCarbon integration to report CO_2 emissions per submission, aligned with the competition's sustainable NLP emphasis. Official results show a MAE_Combined of 1.039 on Task 1 (rank 6 overall; best R10 early-detection score among all participants: 0.889) and a Macro F1 of 0.211 on Task 2. We discuss empirical observations from execution logs, including progressive flattening of CompACT-10 predictions in late rounds and a model configuration bug that made the three runs effectively equivalent.

Keywords

MentalRiskES, IberLEF 2026, mental health, GAD-7, PHQ-9, CompACT-10, large language models, zero-shot prompting, function calling, CodeCarbon, sustainable NLP

1. Introduction

Early detection of mental health disorders is a problem of growing clinical relevance. Mental health professionals face patient volumes that make systematic administration of standardized questionnaires difficult over time, and timely intervention is known to significantly improve outcomes. Natural language processing offers the possibility of inferring clinical states from spontaneous patient language, complementing, rather than replacing, clinical judgment.

MentalRiskES, organized within the Iberian Languages Evaluation Forum (IberLEF), established a reproducible evaluation framework for mental health risk detection in Spanish across its 2023 and 2024 editions. The 2026 edition [3] introduces a more realistic and demanding scenario: instead of classification from isolated messages, systems must process iterative therapeutic conversations where clinically relevant information evolves across multiple turns. The task now also requires selecting the most appropriate therapist response, adding a component of reasoning about good clinical communication practices.

This paper describes the system developed by team BUAP-NLP Lab, from the Benemérita Universidad Autónoma de Puebla (BUAP), for participation in both tasks. Our approach is deliberately lightweight: commercial LLMs in zero-shot mode, guided by prompts with embedded clinical knowledge and output

IberLEF 2026, September 2026, León, Spain

*Corresponding author.

alfredonava794@gmail.com (A. Navarrete-Montes); alejandroamadorlag@hotmail.com (A. Amador-Lagunes); luis.mendezsanchez@correo.buap.mx (L. Y. Méndez-Sánchez)

0009-0007-1969-6326 (A. Navarrete-Montes); 0009-0009-2713-6817 (A. Amador-Lagunes); 0000-0003-2283-6987 (L. Y. Méndez-Sánchez)

constraints enforced via function calling. The motivation is threefold: the structured output space suits schema-based tools; no annotation or curation costs are incurred; and systematic exploration of cost-quality trade-offs becomes possible through heterogeneous model selection per run.

The rest of this paper is organized as follows. Section 2 situates the work within prior MentalRiskES editions. Section 3 describes both tasks, the questionnaires, the server protocol, and evaluation metrics. Section 4 details the proposed system. Section 5 reports the experimental setup. Section 6 presents official results. Section 7 discusses observations and limitations. Section 8 concludes.

2. Related Work

MentalRiskES was introduced at IberLEF 2023 [1] as the first shared evaluation in Spanish for early detection of mental health disorder risk, using approximately 175 annotated Telegram users. That edition attracted 37 registered teams, 18 with submitted results, and 16 papers. Dominant solutions were transformer-based classifiers (BETO, RoBERTuito, MarIA) in direct fine-tuning and embedding-based setups. The 2024 edition [2] extended tasks to three subtasks and introduced mandatory carbon emission reporting, anchoring the competition in the Green AI movement.

The 2026 edition [3] changes the problem fundamentally: input is now a therapeutic conversation delivered incrementally by rounds, and the output is a structured vector plus a categorical clinical decision. This shift reduces the utility of prior-edition datasets for fine-tuning and opens space for zero-shot LLM approaches. While commercial LLMs in zero-shot mode have shown mixed results for mental health tasks, their main advantage is flexibility and ease of deployment; our work adds emphasis on operational robustness and energy transparency.

3. Task Description

3.1. Iterative Server Protocol

The competition operates through a client-server protocol where the client must submit six requests per round before the server advances:

1. `GET /task1/getmessages/{token}`
2. `POST /task1/submit/{token}/0` (run 0)
3. `POST /task1/submit/{token}/1` (run 1)
4. `POST /task1/submit/{token}/2` (run 2)
5. `GET /task2/getmessages/{token}`
6. `POST /task2/submit/{token}/{0,1,2}` (three runs)

An empty JSON response to a GET signals competition completion.

3.2. Task 1: Psychometric Questionnaire Prediction

Given the accumulated message history of a patient up to the current round, the system must estimate that patient's responses to three standardized questionnaires:

- **GAD-7** [5]: 7 items, scale 0–3. Screening for generalized anxiety disorder.
- **PHQ-9** [6]: 9 items, scale 0–3. Screening for major depressive disorder.
- **CompACT-10** [7]: 10 items, scale 0–6. Evaluates acceptance, defusion, and committed action within the ACT framework.

Total output: 26 bounded integer values per patient per round.

3.3. Task 2: Therapist Response Selection

Given the conversational history, the last received message, and three candidate therapist responses (`option_1`, `option_2`, `option_3`), the system must select the most clinically appropriate option.

Options vary by round and patient.

3.4. Evaluation Metrics

For Task 1, the reported metrics are MZOE (Mean Zero-One Error: proportion of exact item predictions), MAE (mean absolute error per item), and Macro_MAE (macro-average across all items) per questionnaire, plus MAE_Combined aggregating all three. For early detection, MAE_Combined is computed at temporal windows R10, R30, and R60 (after 10, 30, and 60 patient messages respectively). For Task 2: Overall Accuracy, Macro F1, and Error Recovery. All systems also report CodeCarbon emission metrics.

4. Proposed System

4.1. General Architecture

The system is a self-contained Python 3.11 Jupyter Notebook with four functional layers:

Configuration: environment variables, execution flags (DRY_RUN, verbosity), robustness parameters (retries, backoff, timeouts), and dual logging to console and file.

Clinical definitions: official Spanish-language texts of all 26 questionnaire items with their headers and scale descriptions, stored as immutable module constants.

LLM inference: function calling schemas, prompt builders, OpenAI API calls with exponential retries, output validation and coercion, and fallback mechanisms.

Server communication: HTTP client encapsulating GET/POST with retries, local conversational history persistence per patient, and submission traceability.

The orchestrating function `get_post_data` executes the main loop: for each round it retrieves messages for both tasks, executes the three runs per task while measuring emissions with CodeCarbon, submits predictions in real mode or logs them in dry-run mode, and advances to the next round.

4.2. Task 1 Prompting Strategy

The Task 1 prompt is structured in five blocks: (i) declaration of the model's clinical role; (ii) explicit instruction to base inference solely on evidence present in the history, with conservative defaults (0–1 for GAD-7/PHQ-9; 3, the midpoint, for CompACT-10) for short or ambiguous histories; (iii) full numbered conversational history by round; (iv) internal clinical reasoning block with four guiding questions (predominant emotions, avoidance/rumination signals, concrete symptom evidence, temporal stability of patterns); and (v) enumerated questionnaire items with scales and the instruction to return the 26 answers via the `report_questionnaires` tool. The reasoning block acts as a lightweight chain-of-thought without requesting explicit verbalization, reducing output tokens while keeping responses schema-constrained. Figure 1 shows the actual prompt template used in the official runs.

4.3. Task 2 Prompting Strategy

The Task 2 prompt mirrors the Task 1 structure, adapted to a categorical decision: (i) patient conversational history; (ii) last received message in isolation; (iii) internal clinical reasoning (predominant emotion, communicative intent, immediate clinical need, relevant historical pattern); (iv) five explicit selection criteria derived from therapeutic best practices (prioritize empathic validation, prefer open-ended questions, avoid minimization, warm and clear language); (v) three candidate options; and (vi) instruction to invoke `choose_response` with `option_1`, `option_2`, or `option_3`. Figure 2 presents the Task 2 prompt template.

4.4. Structured Function Calling

Output validity is enforced via function calling with forced `tool_choice`, eliminating common LLM-pipeline errors: free-text responses, malformed JSON, out-of-range values, and missing keys. The Task 1

```
Eres un psicologo clinico experto.  Vas a leer el historial de mensajes de un
paciente
en una conversacion terapeutica y debes estimar como responderia el paciente a
tres
cuestionarios psicometricos estandarizados, basandote UNICAMENTE en el contenido
observado hasta ahora.

Si el historial es muy corto o no hay evidencia clara, responde con un valor
conservador (0 o 1 para GAD-7/PHQ-9; valor medio 3 para CompACT-10).

HISTORIAL DEL PACIENTE (rondas N):
Ronda 1: [msg_1] ... Ronda N: [msg_N]

Antes de completar los cuestionarios, analiza internamente:
- Que emociones predominan en el historial?
- Hay senales de evitacion, rumiacion o ideacion negativa?
- Que evidencia concreta hay para cada sintoma?
- El patron clinico es estable, mejora o empeora?

[Items GAD-7 / PHQ-9 / CompACT-10 con escalas completas]

Usa la herramienta 'reportar_cuestionarios' para devolver las 26 respuestas.
```

Figure 1: Task 1 prompt template (abbreviated). The five blocks are: (1) clinical role declaration, (2) conservative default policy for ambiguous histories, (3) numbered conversational history, (4) internal chain-of-thought reasoning block, (5) questionnaire items with function-calling instruction. Prompts are issued in Spanish, matching the language of the therapeutic conversations.

```
Eres un psicologo clinico experto. Vas a leer el ultimo mensaje de un paciente y
tres
posibles respuestas del terapeuta. Debes elegir la respuesta MAS APROPIADA.

HISTORIAL DEL PACIENTE (ultimas rondas):
Ronda 1: [msg_1] ... Ronda N: [msg_N]

ULTIMO MENSAJE DEL PACIENTE: [last_message]

Antes de elegir, analiza internamente:
- Que emocion predominante expresa el paciente?
- Esta buscando ser escuchado, orientado, o iniciando contacto?
- Que necesita ahora: validacion empatica, exploracion o informacion?
- Hay un patron clinico en el historial que guie la respuesta?

Criterios de eleccion:
- Prioriza validacion empatica sobre consejos directos
- Prefiere preguntas abiertas que inviten a explorar
- Evita minimizar los sentimientos del paciente
- Prefiere lenguaje calido y claro

OPCIONES: 1. [opt1] 2. [opt2] 3. [opt3]

Usa la herramienta 'elegir_respuesta' para devolver option_1, option_2 u
option_3.
```

Figure 2: Task 2 prompt template (abbreviated). Blocks: (1) history context, (2) last patient message in isolation, (3) internal reasoning, (4) five therapeutic selection criteria, (5) candidate options, (6) function-calling instruction.

schema declares three required array properties with `minItems`, `maxItems`, `minimum`, and `maximum` per questionnaire. The Task 2 schema declares a single string property constrained by enum to the three valid values. A post-validation layer additionally coerces Task 1 values to integer and clips them to the valid range.

4.5. Heterogeneous Model Selection Per Run

Rather than executing the same model three times, our system assigns a different model to each run, covering a cost-quality spectrum and providing real diversity. Table 1 shows the intended configuration.

Table 1
Intended model assignment per official run.

Run	Model	Profile
0	`gpt-4o-mini`	Best cost-quality balance
1	`gpt-4.1-nano`	Low-cost, lateral diversity
2	`gpt-5-nano`	Extended reasoning, higher latency

As discussed in Section 7, a configuration bug caused all three runs to fall back to `gpt-4o-mini` from round 7 onward.

4.6. Persistence and Robustness

Three types of data are persisted locally under `./data/`: accumulated message history per patient (enabling recovery after interruption), full submission payloads per run (for post-hoc auditing), and raw server GET payloads per round (for offline inspection). API and server calls use exponential backoff retry loops (base 2.0 s, up to three attempts). On persistent LLM failure, a deterministic fallback is invoked: all-zeros for Task 1 (conservative: no symptoms) and `option_1` for Task 2.

4.7. Emission Measurement with CodeCarbon

Each submission includes an emissions object with twelve required fields (`duration`, `emissions`, cpu/gpu/ram energy, cpu/gpu counts and models, ram size, country code). An `EmissionsTracker` is instantiated before each run and stopped upon completion. Measurement uses process-level tracking, so each run is independently accounted for.

5. Experimental Configuration

5.1. Models and Provider

All three models are accessed through the official OpenAI API via the `openai` Python SDK, with no additional provider or fine-tuning. Temperature is not explicitly set, introducing stochastic variability across runs to favor diversity.

5.2. Robustness Parameters

`MAX_API_RETRIES = 3`; `MAX_SERVER_RETRIES = 3`; `BACKOFF_BASE = 2.0` (wait times: 1, 2, 4 s); `HTTP_TIMEOUT = 60 s`; `MAX_ROUNDS_DRY_RUN = 2`.

5.3. Execution Infrastructure

The official run was performed on a laptop with an Intel Core i7-13620H (16 logical cores), 16 GB RAM, and an NVIDIA RTX 4060 Laptop GPU, running Windows 11 with Python 3.11 in a dedicated virtual environment. CodeCarbon uses Intel RAPL for CPU and RAM measurements; the reported country code is `MEX`. Execution logs confirm the official run began at 11:30 and completed at 14:58 on April 18, 2026, for a total wall-clock duration of approximately 3 hours and 28 minutes. A 13-minute interruption occurred between 12:27 and 12:40 (see Section 7.3), making the effective active execution approximately 3 hours and 15 minutes.

5.4. Execution Modes

In dry-run mode (`DRY_RUN=True`), LLM calls run at real cost but predictions are stored locally without server submission. This mode was used extensively to validate prediction quality before committing to official submissions. Real mode (`DRY_RUN=False`) executes all six POSTs per round.

6. Results

6.1. Task 1: Psychometric Questionnaire Prediction

Table 2 shows official Task 1 results. The three runs achieved near-identical performance due to the model configuration bug described in Section 7.

Table 2
Official Task 1 results (MAE per questionnaire and combined). Lower is better. Reference systems included for comparison.

System	$MAE_{GAD\text{-}7}$	$MAE_{PHQ\text{-}9}$	MAE_{cACT}	$MAE_{Comb.}$
BUAP Run 0	0.684	0.815	1.619	1.039
BUAP Run 1	0.703	0.848	1.612	1.054
BUAP Run 2	0.707	0.846	1.607	1.053
BASELINE Gemma (rank 4)	0.582	0.724	1.700	1.002
FBKillers run 2 (rank 1)	0.595	0.749	1.293	0.879

Run 0 achieved rank 6 overall (MAE_Combined = 1.039). GAD-7 showed the lowest error (MAE $\approx$ 0.68) while CompACT-10 showed the highest (MAE $\approx$ 1.62).

6.1.1. Early Detection Performance

Table 3 shows MAE_Combined at temporal windows. Run 1 achieved the **best R10 score among all participating teams** (0.889), outperforming FBKillers (0.906) and VerbaNex AI (0.940). All three BUAP runs tied for best R60 (0.671). Performance was weakest at R30.

Table 3
Early detection MAE_Combined at temporal windows R10, R30, R60. Bold indicates best result overall. Lower is better.

Run	R10	R30	R60
BUAP Run 0	0.942	1.420	**0.671**
BUAP Run 1	**0.889**	1.330	**0.671**
BUAP Run 2	0.949	1.410	**0.671**
FBKillers (best overall)	0.906	1.260	0.657
VerbaNex AI (rank 2)	0.940	1.210	0.800

6.2. Task 2: Therapist Response Selection

Table 4 shows official Task 2 results. All three runs placed in the bottom quarter of the 37-entry ranking (positions 30, 34, and 37 respectively).

Figure 3 shows the confusion matrices for the three runs. The dominant pattern is a near-uniform distribution of errors: the diagonal cells (correct predictions) are consistently the smallest or second-smallest in their row, indicating that the model fails to discriminate among the three response candidates at a level above random chance. When the correct answer was Option 1, errors are distributed almost equally between Options 2 and 3. When the correct answer was Option 2, the most frequent error is predicting Option 1 (over 50% of cases across all runs). The same bias toward Option 1 appears

Table 4
Official Task 2 results. Higher is better.

Run	Overall Acc.	Macro F1	Error Recovery
Run 0	0.201	0.198	0.190
Run 1	0.215	0.211	0.190
Run 2	0.194	0.191	0.176

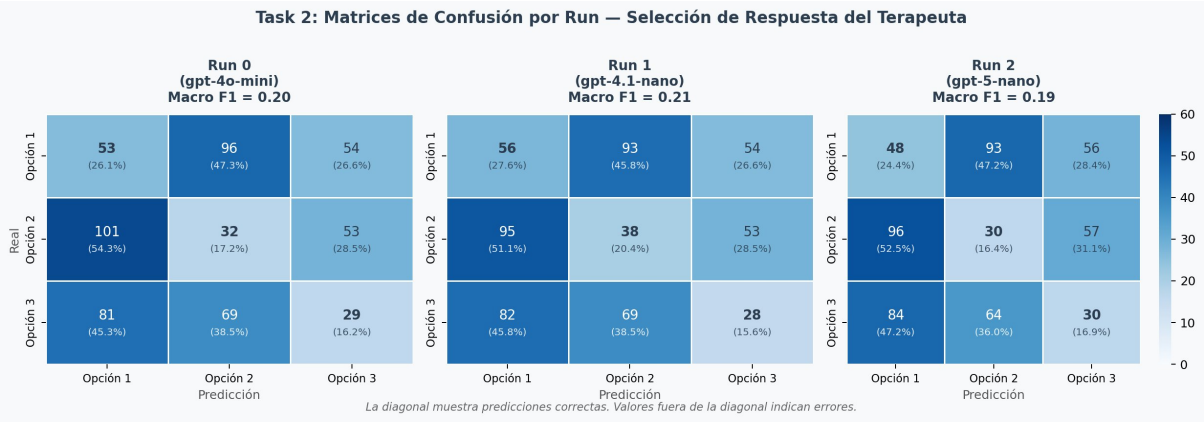

Figure 3: Confusion matrices for Task 2 across all three runs. Rows represent the true therapist response; columns represent the predicted response. Bold values on the diagonal indicate correct predictions.

when Option 3 is the correct answer. This systematic tendency to over-predict Option 1 is consistent across runs despite their near-identical effective configuration. Overall, the accuracy per class hovers around 16–28%, close to the random baseline of 33%, which confirms that zero-shot prompting without task-specific supervision is insufficient for the nuanced pragmatic reasoning required by Task 2. These results provide direct empirical support for the qualitative observations in Section 7.4.

6.3. Sustainability Report

Table 5 shows aggregated CodeCarbon data from the full official execution (≈81 rounds). The total footprint of ≈22 g CO_2e is comparable to a brief email exchange. BUAP-NLP Lab was the team with the **lowest total energy consumption** among all participants (≈2.95×10^{-5} kWh for Task 2 submissions).

Table 5
Energy and CO_2 emissions per run (CodeCarbon).

Run	Energy (Wh)	CO_2 (g)	Time (min)
Run 0 (`gpt-4o-mini`)	9.9	5.0	≈28
Run 1 (`gpt-4.1-nano`)	7.4	3.8	≈22
Run 2 (`gpt-5-nano`)	26.0	13.2	≈51
Total	43.3	22.0	≈101

7. Discussion

7.1. `gpt-5-nano`: Latency vs. Reasoning Quality

Run 2 consumed ≈2.6× more energy than Run 0 despite processing the same patients and rounds. Internal timestamps show `gpt-5-nano` takes 20–30 s per prediction and generates thousands of internal tokens before completing the tool call, consistent with extended-reasoning behavior in the model-5

family. This implies a trade-off: deeper reasoning comes at the cost of higher latency, higher energy consumption, and greater timeout risk.

7.2. Progressive Flattening of CompACT-10

A notable observation from execution logs is the progressive flattening of CompACT-10 predictions in advanced rounds. In the final processed round (patient S12, round 82), the predicted CompACT-10 vector was uniform at value 3 across all ten items, matching the conservative midpoint default instructed in the prompt for ambiguous evidence.

As conversational history grows, the model tends toward a risk-minimizing response, likely due to dilution of concrete clinical signals in a long context. This is especially pronounced for CompACT-10, whose constructs (acceptance and values processes) are more abstract and less directly inferable from conversational text than GAD-7 or PHQ-9 symptoms. Future work should explore few-shot examples specific to CompACT-10, history trimming to the last N turns, and prompt re-weighting to emphasize differentiation from the conservative default.

7.3. Round Counter Bug and Model Configuration Loss

Two distinct bugs affected the official execution. First, a round counter discrepancy: Loop 1 (≈11:34–12:27) completed six rounds and was interrupted mid-round 7, most likely by a `gpt-5-nano` timeout on Task 2 run 2. Loop 2 (≈12:40–14:58) restarted and recovered all patient histories from disk, but the round counter reset to zero on each invocation of `get_post_data`, reporting 75 rounds at completion while the server had advanced to round 82. This does not affect submission correctness (all POSTs returned HTTP 200/201) but limits post-execution observability.

Second, a model configuration loss: the per-run model dictionary was not persisted to disk and reset upon restart, causing all three runs to fall back to `gpt-4o-mini` from round 7 onward. This explains the near-identical performance across runs in Table 2. Both bugs will be addressed in future versions: reading the server's `round` field as the single source of truth, and persisting the model configuration alongside the conversational history.

7.4. Task 2 Underperformance

Task 2 requires nuanced pragmatic and clinical reasoning: distinguishing subtle differences in phrasing, empathic tone, and therapeutic alignment across three options that may appear superficially similar. Zero-shot prompting without task-specific few-shot examples is insufficient for this precision. The model configuration bug may have also disproportionately affected Task 2, since `gpt-5-nano`'s extended reasoning might have provided more careful pragmatic evaluation than `gpt-4o-mini`.

7.5. Heterogeneous Model Selection: Benefits and Limitations

The heterogeneous assignment strategy provides a key operational property: if one model experiences degradation or systematic timeouts, the other two runs remain valid submissions. This was relevant in practice, as `gpt-5-nano`'s slowness would have compromised all submissions under a uniform configuration. However, heterogeneity complicates interpretability: attributing a round's performance to a specific model requires per-run analysis.

8. Conclusions and Future Work

This paper described the participation of team BUAP-NLP Lab in MentalRiskES 2026. The chosen approach, zero-shot prompting with commercial LLMs, structured function calling, and heterogeneous model selection per run, prioritized operational simplicity, robustness against failures, and transparency in energy consumption. The system completed the official competition with a total footprint of ≈22 g CO_2e and recovered from an intermediate interruption thanks to disk-persisted history.

Official results show competitive Task 1 performance (MAE_Combined 1.039, rank 6; best R10 early-detection score among all participants: 0.889) and below-expectation Task 2 results (Macro F1 0.211, ranks 30–37/37). A key lesson is that Task 1 (structured regression with explicit symptom anchors in language) is a better fit for zero-shot LLMs than Task 2 (nuanced pragmatic reasoning over near-similar response candidates).

Future work directions: (i) few-shot examples specific to CompACT-10 to mitigate progressive flattening; (ii) adaptive history trimming to the last N turns to preserve clinical signal in late rounds; (iii) fixing the round counter and model configuration persistence bugs; (iv) exploring open-weights Spanish models (BETO, MarIA, fine-tuned Llama-3) for more controlled ablations; and (v) per-questionnaire emission breakdowns to inform future design decisions. Beyond these specific points, this work illustrates that a commercial LLM zero-shot system can serve as a non-trivial and operationally solid baseline for structured clinical NLP tasks in Spanish when function calling and robustness are treated as first-class requirements.

Declaration on Generative AI

In accordance with the CEUR-WS policy on the use of generative AI tools in publications (effective January 2025), the authors declare the following. During manuscript drafting, Claude by Anthropic was used as a writing assistance tool. Its specific contributions, following the GenAI Usage Taxonomy, are: (i) drafting and structural support for organizing and expanding section content; (ii) paraphrasing and style revision to improve clarity and conciseness; and (iii) grammar and coherence editing of individual passages. All content was reviewed, edited, and validated by the authors, who assume final responsibility for its accuracy and integrity. No AI model appears as an author of this paper.

Acknowledgments

The authors thank Luis Yael Méndez Sánchez for academic guidance throughout the project, and the Benemérita Universidad Autónoma de Puebla for institutional support. We also thank the organizers of MentalRiskES 2026 and IberLEF for organizing the shared evaluation and maintaining the evaluation infrastructure accessible throughout the competition.

References

[1] A. M. Mármol-Romero, A. Moreno-Muñoz, F. M. Plaza-del-Arco, M. D. Molina-González, M. T. Martín-Valdivia, L. A. Ureña-López, and A. Montejo-Ráez. Overview of MentalRiskES at IberLEF 2023: Early Detection of Mental Disorders Risk in Spanish. *Procesamiento del Lenguaje Natural*, 71:329–350, 2023.

[2] A. M. Mármol-Romero, A. Moreno-Muñoz, F. M. Plaza-del-Arco, M. D. Molina-González, M. T. Martín-Valdivia, L. A. Ureña-López, and A. Montejo-Ráez. Overview of MentalRiskES at IberLEF 2024: Early Detection of Mental Disorders Risk in Spanish. *Procesamiento del Lenguaje Natural*, 73, 2024.

[3] A. M. Mármol-Romero, F. Suárez-Maroto, R. García-Viedma, M. Hernández-López, F. M. Plaza-del-Arco, M. D. Molina-González, L. A. Ureña-López, and A. Montejo-Ráez. Overview of MentalRiskES at IberLEF 2026: Early Detection of Mental Disorders Risk in Spanish. *Procesamiento del Lenguaje Natural*, 77, 2026.

[4] Adrià Bonet-Jover, José Antonio González-Barba, and Luis Chiruzzo. Overview of IberLEF 2026: Natural Language Processing Challenges for Spanish and other Iberian Languages. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2026), co-located with the 42nd Conference of the Spanish Society for Natural Language Processing (SEPLN 2026)*, CEUR-WS.org, 2026.

[5] R. L. Spitzer, K. Kroenke, J. B. W. Williams, and B. Löwe. A brief measure for assessing generalized anxiety disorder: The GAD-7. *Archives of Internal Medicine*, 166(10):1092–1097, 2006.

[6] K. Kroenke, R. L. Spitzer, and J. B. W. Williams. The PHQ-9: Validity of a brief depression severity measure. *Journal of General Internal Medicine*, 16(9):606–613, 2001.

[7] A. W. Francis, D. L. Dawson, and N. Golijani-Moghaddam. The development and validation of the Comprehensive Assessment of Acceptance and Commitment Therapy Processes (CompACT). *Journal of Contextual Behavioral Science*, 5(3):134–145, 2016.

[8] V. Schmidt, K. Goyal, A. Joshi, B. Feld, L. Conell, N. Laskaris, et al. CodeCarbon: Estimate and track carbon emissions from machine learning computing, 2021. https://codecarbon.io/

[9] OpenAI. GPT-4o and GPT-4o mini: API documentation, 2024. https://platform.openai.com/docs/models