VerbaNex AI Lab at PoliticHeadlinES-IberLEF 2026: Generative and Embedding-Based Approaches for Multimodal Headline Ranking in Spanish Political News

Carlos Agamez[1,*], Edwin Puertas[1], Jairo Serrano[1] and Juan Carlos Martinez-Santos[1]

[1]*Escuela de Transformación Digital, Universidad Tecnológica de Bolívar, Cartagena, Colombia*

Abstract

This paper describes the system submitted by team VerbaNex AI Lab to the PoliticHeadlinES shared task at IberLEF 2026, which focuses on multimodal headline ranking for Spanish political news. Given a news article and ten candidate headlines (one correct, nine distractors), participants must rank the candidates from most to least relevant. We explore three complementary approaches: (i) an embedding-based ranker using semantic similarity with the `embeddinggemma` model and multiple distance metrics; (ii) a hybrid text-image fusion combining `embeddinggemma` with CLIP visual embeddings at varying interpolation weights; and (iii) a generative ranker built on `gemini-3.1-flash-lite-preview` that directly reasons over the article text and candidate headlines using structured output. Our best system, the generative ranker, achieves an Acc@1 of 0.945 and PA-nDCG of 0.9377 on Task 1 (text-only) and 0.930 / 0.9227 on Task 2 (multimodal) on the development set, demonstrating that instruction-following language models substantially outperform embedding-based baselines on this task.

Keywords

headline ranking, multimodal NLP, generative ranking, Spanish political news, IberLEF 2026

1. Introduction

Political news headlines play a central role in how citizens discover and interpret information in digital media environments. Automatically selecting or ranking the most appropriate headline for a given article is therefore a relevant challenge at the intersection of Natural Language Processing (NLP), Information Retrieval (IR), and multimodal learning.

PoliticHeadlinES [1] is a shared task hosted at IberLEF 2026 [2] that formalizes this problem for Spanish political news. Given a news article and ten candidate headlines, systems must produce a ranking that places the original, correct headline at the top. The dataset originates from PoliSUM-2025 [3].

A key difficulty of this task is that the candidate headlines are deliberately chosen to be thematically similar, making them hard to distinguish through simple lexical or semantic similarity alone. Correctly identifying the original headline requires understanding subtle factual and editorial differences between candidates — a form of reasoning that goes beyond vector-space retrieval. This observation motivated our progressive exploration from retrieval-based methods toward prompt-engineered generative models that exploit the multimodal capabilities of large language models.

In this paper, we describe the system submitted by team VerbaNex AI Lab (Universidad Tecnológica de Bolívar, Colombia). Our participation in this challenge continues our research lab's ongoing efforts to leverage NLP for analyzing and improving processes in the news media ecosystem, building upon our previous work in detecting persuasion and framing in online news [4] as well as developing automated fake news detection systems [5]. For the present task, we report experiments conducted across two competition phases, tracing an evolution from TF-IDF and dense embedding rankers in the development

IberLEF 2026, September 2026, León, Spain

*Corresponding author.

cagamez@utb.edu.co (C. Agamez); epuerta@utb.edu.co (E. Puertas); jserrano@utb.edu.co (J. Serrano); jcmartinezs@utb.edu.co (J. C. Martinez-Santos)

0009-0004-7849-9734 (C. Agamez); 0000-0002-0758-1851 (E. Puertas); 0000-0001-8165-7343 (J. Serrano); 0000-0003-2755-0718 (J. C. Martinez-Santos)

phase to a fully generative approach using the Google Gemini API in the evaluation phase, where prompt engineering and native multimodal reasoning yielded our best results.

The rest of the paper is structured as follows. Section 2 describes the task and dataset. Section 3 presents our system architecture and the evolution of our approach across both phases. Section 4 reports experimental results. Section 5 discusses the findings. Section 6 concludes the paper.

2. Task and Dataset Description

PoliticHeadlinES [1] is a shared task at IberLEF 2026 that ranks candidate headlines for Spanish political news articles. For each instance, a system receives the full body of a news article, along with a set of 10 candidate headlines, exactly one of which is the original headline published with the article; the remaining 9 are thematically related distractors chosen to maximize difficulty.

The objective is to produce a complete ranking of the ten candidates, ordered from most to least relevant.

2.1. Subtasks

The shared task defines two independent subtasks:

Task 1 — Text-only ranking. Systems must rank the candidate headlines using exclusively the textual content of the article. This subtask evaluates the ability of models to capture semantic relevance and coherence from language alone.

Task 2 — Multimodal ranking (text + image). Systems may additionally exploit an image associated with the article. It is the primary subtask of the challenge: the official leaderboard ranking is determined by Task 2 results.

2.2. Dataset

The dataset is derived from PoliSUM-2025 [3], a corpus of Spanish political news. A unique MD5 hash identifies each article, and its ten candidate headlines are labeled `t1`–`t10`. Table 1 summarises the dataset statistics.

Table 1
PoliticHeadlinES dataset statistics by competition phase.

Phase	Split	Articles	Candidates/article
Development	Train	100	10
Development	Dev	2,825	10
Evaluation	Train	16,030	10
Evaluation	Test	4,912	10

All articles across both phases have an associated image, making the full dataset suitable for multimodal experimentation. The development phase provided a smaller corpus used for initial exploration and system prototyping, while the evaluation phase released the full-scale dataset used for the official submission. The test-set ground-truth labels are held private and used for official evaluation on CodaLab.

2.3. Evaluation Metric

The official evaluation metric is **PA-nDCG** (Position-Aware normalized Discounted Cumulative Gain), computed with $\alpha = 0.9$ and $k = 10$. This metric rewards systems that place the correct headline at the top of the ranking, while penalizing inversions in proportion to their distance from the ideal position. We additionally report **Acc@1** (accuracy at rank 1, i.e. the fraction of instances where the correct headline is ranked first) for the embedding-based experiments.

2.4. Official Baselines

The shared task organizers provided two official baselines implemented in a public Google Colab notebook to serve as starting points for participants.

Task 1 baseline — TF-IDF. For the text-only subtask, the baseline ranks candidate headlines by cosine similarity with the article body using a TF-IDF representation (unigrams and bigrams, up to 50,000 features, fitted on the training corpus).

Task 2 baseline — TF-IDF + CLIP. For the multimodal subtask, the baseline extends the text signal with visual similarity computed by a pre-trained CLIP model (`openai/clip-vit-base-patch32`) [6]. Both signals are min-max normalized per row and linearly combined with fixed weights ($w_{\text{text}} = 0.85$, $w_{\text{img}} = 0.15$). When an image is unavailable, the system falls back to the TF-IDF text-only ranking.

Table 2 reports the official baseline scores evaluated on the development phase dev set (2,825 articles).

Table 2
Official baseline scores on the development phase dev set.

System	Task 1 PA-nDCG	Task 2 PA-nDCG
TF-IDF + CLIP (official baseline)	0.836	0.816

These baselines establish the lower-bound reference for our experiments; our systems build on and substantially improve upon this starting point.

3. System Description

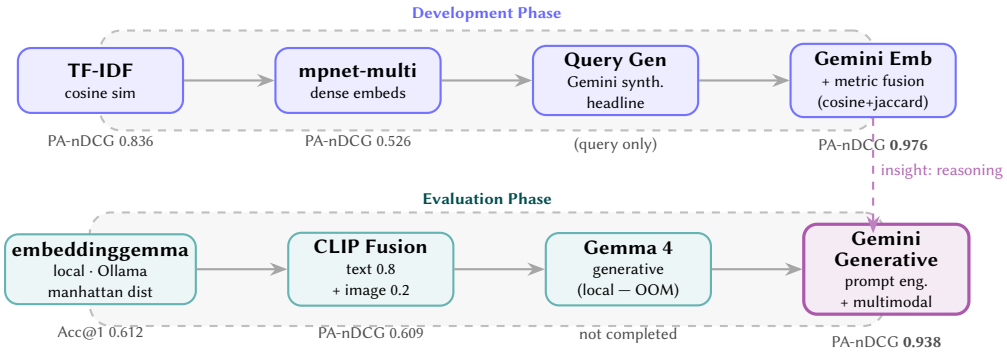

Figure 1: Evolution of system approaches across the two competition phases. Each node shows the method and its development-set score. The dashed arrow indicates a shift from retrieval-based to reasoning-based ranking.

Our approach evolved across two competition phases. In the development phase, we explored retrieval-based methods to increase expressiveness, starting with sparse representations and progressing to dense multimodal embeddings. In the evaluation phase, guided by the insight that headline discrimination requires reasoning rather than similarity measurement, we shifted to a fully generative paradigm using prompt engineering and the native multimodal capabilities of the Google Gemini API.

3.1. Development Phase: From Sparse Retrieval to Semantic Embeddings

The development phase provided a small corpus (100 training articles, 2,825 development articles) that we used for rapid experimentation with different representation strategies.

Sparse baselines. Following the official baseline, we first implemented a TF-IDF ranker with cosine similarity (unigrams and bigrams, up to 50,000 features). We also tested the Spanish-specialized encoder `sentence-transformers/paraphrase-multilingual-mpnet-base-v2` [7], a multilingual dense sentence encoder, as an alternative based on Transformer sentence embeddings [8]. Neither

approach produced competitive results, achieving PA-nDCG around 0.526 on the development set, revealing the limitations of purely lexical or generic semantic similarity for this task.

Query generation with Gemini. To better bridge the semantic gap between the article body and the candidate headlines, we introduced a query generation step: for each article, `gemini-2.0-flash` was prompted — in the role of a Spanish political newspaper editor — to produce an ideal headline summarising the article. This synthetic headline was then used as the retrieval query instead of the full article body, placing the query and the candidates in a more comparable semantic space — a strategy conceptually related to retrieval-augmented generation [9].

Gemini dense embeddings and metric fusion. Replacing the query encoder with `gemini-embedding-2-preview` resulted in a significant performance improvement. To maximize retrieval quality, we ran a systematic grid search over linear combinations of cosine, Jaccard, and Manhattan distances with weights summing to 1.0 (step size 0.1, approximately 66 combinations). The best configuration — cosine 0.8 + Jaccard 0.2 — achieved PA-nDCG = 0.9764 on the development set, placing VerbaNex AI Lab second on the development-phase leaderboard. Task 2 was not separately implemented in this phase; the text-only ranking was submitted for both subtasks.

Key limitation identified. Despite strong results, embedding-based ranking fundamentally compares vector distances. Given that the candidate headlines are designed to be thematically similar, fine-grained discrimination between them requires understanding subtle factual and editorial differences — a reasoning capability that vector similarity alone cannot reliably provide.

3.2. Evaluation Phase: Generative Ranking with Prompt Engineering

In the evaluation phase, the full dataset was released (16,030 training and 4,912 test articles, all with associated images). Guided by the limitation identified during development, we explored three complementary strategies.

Local embedding ranker (embeddinggemma). As a fast, API-free baseline for the evaluation phase, we used `embeddinggemma` [10] (Google/Gemma 3, 622 MB, served via Ollama) to encode article bodies and candidate headlines. Rankings were produced by Manhattan distance, which yielded the best results among four metrics tested (Acc@1 = 0.612 on a 500-sample evaluation split). For Task 2, we used `gemini-embedding-2-preview` in multimodal mode, encoding article text together with the associated image (Acc@1 = 0.848).

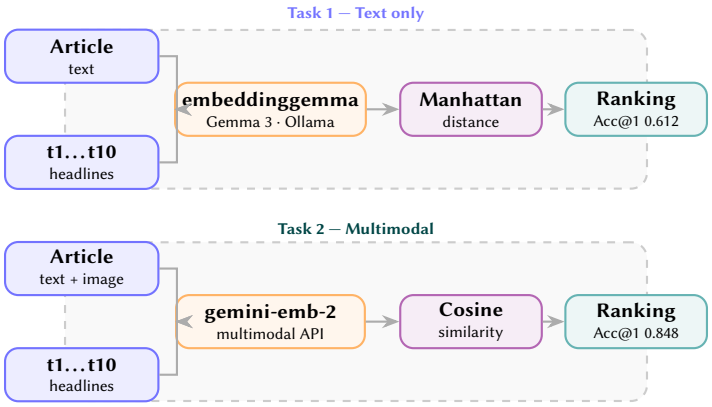

Figure 2: Embedding-based ranking pipelines. Task 1 uses `embeddinggemma` with Manhattan distance (text only). Task 2 uses `gemini-embedding-2 preview` encoding article text and image jointly with cosine similarity.

Hybrid CLIP fusion. To explore lightweight multimodal fusion without API costs, we combined `embeddinggemma` text embeddings with visual embeddings from CLIP [6] (`openai/clip-vit-base-patch32`). Both signals were min-max normalized per row and combined with learnable weights. The best configuration (80% text, 20% image) achieved PA-nDCG = 0.609 and Acc@1 = 0.614, a marginal improvement over the text-only variant but well below the Gemini multimodal embeddings.

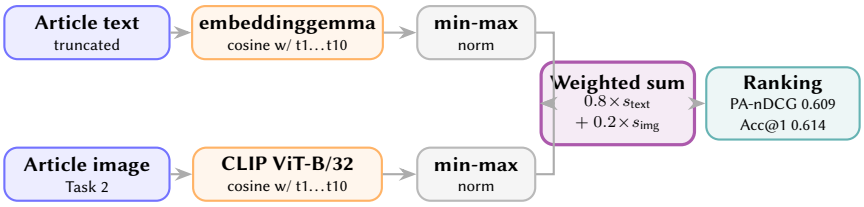

Figure 3: Hybrid CLIP fusion pipeline (Task 2). Text and image similarity scores are computed independently with `embeddinggemma` and CLIP, respectively, min-max normalized per row, then combined with fixed weights (80% text, 20% image).

Local generative ranker (Gemma 4). We attempted to run a fully generative ranker using `gemma4:e2b` [11] (7.2 GB, served via Ollama with RAM offloading). The model was prompted to reason directly over the article and the ten candidate headlines and return a structured ranking. However, this experiment could not be completed due to hardware memory constraints on our local GPU (NVIDIA RTX 3050, 6 GB VRAM).

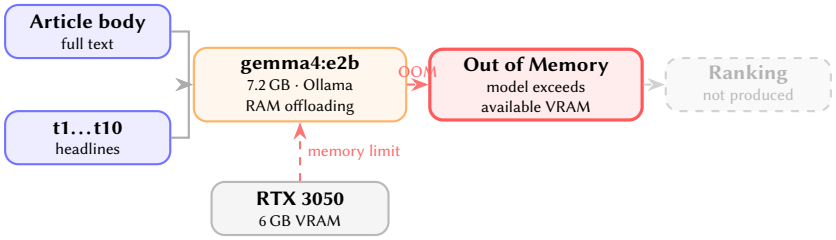

Figure 4: Attempted local generative ranker with `gemma4:e2b`. The model (7.2 GB) exceeded the local GPU's VRAM (NVIDIA RTX 3050, 6 GB), preventing completion of this experiment.

3.3. Final System: Gemini Generative Ranker

The final submitted system reframes headline ranking as a *direct reasoning task* rather than a retrieval problem. Given a news article and its 10 candidate headlines, the model is asked to determine which headline is the original one—a task it can approach by reading, comparing, and reasoning, exactly as a human editor would.

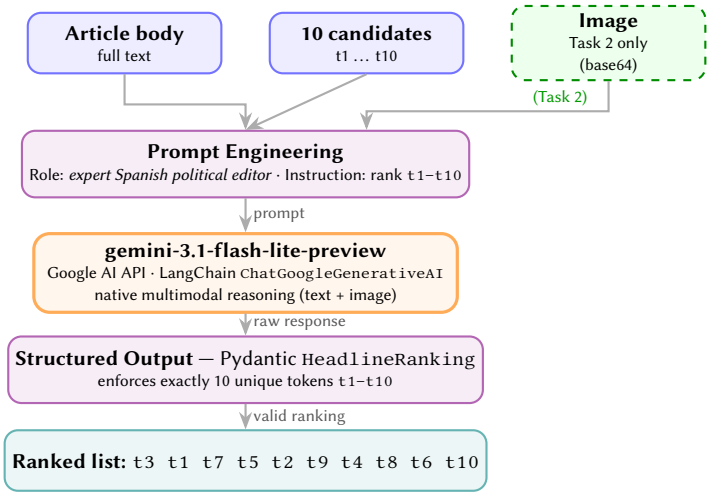

Figure 5: Pipeline of the final generative ranking system. The image input (dashed) is only available in Task 2 and is passed inline to the model, enabling native multimodal reasoning without manual embedding fusion.

Prompt engineering. We used `gemini-3.1-flash-lite-preview` via the Google AI API via LangChain's `ChatGoogleGenerativeAI` interface. Each call consists of two messages: a *system prompt* that establishes the model's role and task, and a *user message* that provides the article body and the ten candidate headlines.

The system prompt used in the submitted system is as follows:

```
Eres un editor jefe especializado en periodismo pol tico en espa ol.
Se te proporcionan el cuerpo de una noticia y 10 titulares
candidatos (t1..t10). Exactamente UNO es el titular original.
Los otros 9 son distractores tem ticos.

Razona en este orden:
1. Identifica el hecho central, los protagonistas y el tono del art culo.
2. Eval a cada titular por: (a) exactitud factual     solo menciona
   hechos presentes en el texto; (b) especificidad     nombres,
   cifras y eventos concretos; (c) estilo editorial     sin
   Sensacionalismo ni generalizacin excesiva.
3. [Task 2] Usa la imagen para confirmar protagonistas o contexto
   Que ayude a descartar titulares imprecisos.
4. Descarta a los titulares con informacin ausente o exagerada.

Devuelve el ranking completo t1..t10 de m s a menos apropiado.
```

The user message follows a fixed template that concatenates the article body (truncated to 8,000 characters for Task 1 and 6,000 for Task 2) with a numbered block of candidate headlines:

```
Articulo:
{article_body}

Titulares candidatos:
t1: {title_1}
t2: {title_2}
...
t10: {title_10}

Devuelve el ranking completo de los 10 titulares.
```

For Task 2 the image is prepended to the user message as a base64-encoded inline block, and the closing instruction is changed to *"Analiza la imagen adjunta junto con el artículo y devuelve el ranking completo."* To illustrate, a real instance from the training set looks as follows (article body abbreviated):

```
Articulo:
El en simo golpe asestado por el Covid a la econom a espa ola,
ahora sacudida por el s bito retraimiento de un consumo privado
que suele tocar techo en las fiestas navidenas [...]

Titulares candidatos:
t1: Las investiduras de todos los candidatos a la presidencia...
t2: El Gobierno asume la gesti n de la crisis en Valencia...
t3: Editorial ABC: El a o de los ajustes          <- correct
t4: Bel n Hoyo dimite como miembro del Comit ...
t5: Editorial ABC: Ni para los secretos de Estado
t6: Ayuso: Pedro S nchez lo hace todo al r e v  s
t7: Viaje a Madrid: El PP pagar  el viaje...
t8: La vivienda y la ley andaluza sobre Do ana...
t9: De "Mr. Marshall" a "Mr. No": c mo  ser ...
t10: Un centenar de condenas por agresi n sexual...

Devuelve el ranking completo de los 10 titulares.

Expected output: t3 t2 t5 t1 t4 t8 t10 t9 t6 t7
```

Structured output. To guarantee well-formed predictions, we used LangChain's structured output with a Pydantic schema (`HeadlineRanking`) that enforces exactly ten unique tokens (`t1`–`t10`) in a ranked list. It eliminates the need for post-processing or fallback heuristics.

Multimodal integration for Task 2. For the multimodal subtask, the associated article image was passed directly to the model alongside the article text and candidates, encoded as a base64 inline image. Rather than fusing separate text and image embeddings with manual weights — as in the CLIP-based approach — the model integrates both modalities natively within its attention mechanism, allowing it to leverage visual context (e.g. recognisable politicians, locations, or events) to further discriminate between candidates.

4. Experiments and Results

Throughout this section we distinguish three types of scores: (i) *development-phase scores*, obtained on the 2,825-article dev set during the development phase; (ii) *internal validation scores*, estimated on held-out samples from the evaluation-phase training split (500 articles for Exp. 1–2; 200 for Exp. 3); and (iii) *official scores*, returned by CodaBench on the full 4,912-article test set. All results in Table 3 correspond to categories (ii) and (iii).

All evaluation-phase experiments were conducted on the full evaluation dataset (16,030 training and 4,912 test articles). Since the test-set ground-truth labels are held private and released only through CodaBench, system performance was estimated on held-out samples from the training split: 500 articles for the embedding-based systems and 200 articles for the generative ranker. The primary metric is PA-nDCG ($\alpha = 0.9$, $k = 10$); we additionally report Acc@1 where applicable. Note that Experiment 1 was evaluated before PA-nDCG was integrated into our pipeline and therefore reports nDCG@10 in its place. Final predictions for all 4,912 test articles were submitted to CodaBench for official scoring.

Table 3 summarises the results of the three evaluation-phase systems described in Section 3, reporting both the internal validation scores (estimated on held-out training-split samples) and the official PA-nDCG scores returned by CodaBench on the 4,912-article test set.

Table 3
Performance of evaluation-phase systems. Internal scores were estimated on held-out training-split samples (500 articles for Exp. 1–2; 200 for Exp. 3). Official scores (metric_1_3 and metric_2_3) are PA-nDCG values returned by CodaBench on the full 4,912-article test set; the mean is their average. † nDCG@10 reported internally (PA-nDCG not computed for this system). **Bold** marks the best result per column.

System	Task 1 PA-nDCG		Task 2 PA-nDCG		Official Mean
	Internal	Official	Internal	Official	
Exp. 1: `embeddinggemma` + `gemini-emb-2`	0.827†	0.831	0.942†	0.578	0.705
Exp. 2: `embeddinggemma` + CLIP fusion	0.595	0.569	0.609	0.572	0.571
Exp. 3: Gemini generative (submitted)	**0.938**	**0.922**	**0.923**	**0.926**	**0.924**

The results reveal a clear hierarchy across paradigms and expose an important generalization failure in the embedding-based approaches. Exp. 1 achieves a Task 1 official PA-nDCG of 0.831, consistent with its internal estimate of 0.827. However, its Task 2 official score collapses to 0.578 despite an internal nDCG@10 of 0.942 — a drop of more than 36 points. This gap suggests that the `gemini-embedding-2-preview` multimodal encoder, while effective on the small training sample, does not generalize to the full test set, likely due to distributional shift between the validation sample and the 4,912-article test set. Exp. 2 produces more consistent results (Task 1: 0.595 vs. 0.569; Task 2: 0.609 vs. 0.572), but remains the lowest-performing system overall, confirming that local CLIP fusion cannot match API-backed multimodal representations. The generative ranker (Exp. 3) is the only system that generalizes reliably across both tasks, achieving official PA-nDCG scores of 0.922 (Task 1) and 0.926 (Task 2), both within 0.002 of the internal validation estimates. This stability, combined with the highest

absolute performance, placed VerbaNex AI Lab **3rd out of 20 participating teams** on the CodaBench leaderboard, confirming that instruction-following generative models are both more accurate and more robust than embedding-based rankers for this task.

Figure 6 compares internal validation and official CodaBench PA-nDCG scores for each system across both tasks, making The generalization gap is visible per experiment.

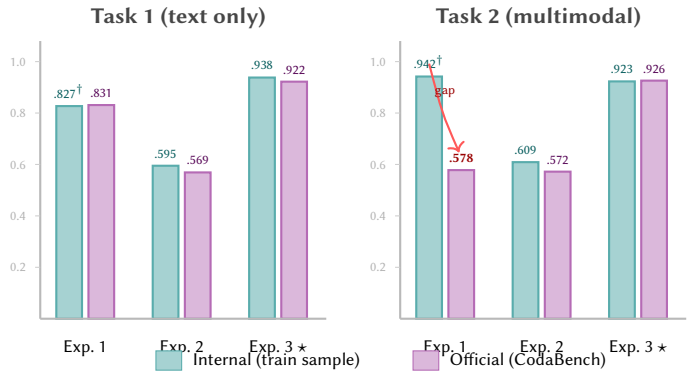

Figure 6: Internal validation vs. official CodaBench PA-nDCG for each system, split by task. Teal bars show scores estimated on held-out training samples; violet bars show the official test-set results. The red arrow highlights the severe generalization gap in Exp. 1 Task 2 (0.942 internal vs. 0.578 official), caused by the `gemini-embedding-2-preview` multimodal encoder not transferring to the full test set. Exp. 3 (⋆, submitted system) generalizes consistently across both tasks. † nDCG@10 reported internally for Exp. 1.

5. Discussion

- **Reasoning vs. retrieval.** The most consistent finding across all experiments is that headline ranking is fundamentally a reasoning task rather than a retrieval task. The candidate headlines in PoliticHeadlinES are deliberately designed to be thematically similar, meaning that vector-space distance between article and headline embeddings cannot reliably discriminate between them. The generative ranker (Exp. 3) succeeds precisely because it reads and compares the candidates — identifying factual nuances, specificity of names and figures, and editorial tone — rather than measuring proximity in a shared embedding space. This observation aligns with recent findings in the information retrieval literature suggesting that large language models acting as rerankers consistently outperform dense retrieval systems on tasks requiring fine-grained textual understanding [9].

- **The generalization failure of multimodal embeddings.** A striking result is the collapse of Exp. 1 on Task 2: the `gemini-embedding-2-preview` multimodal encoder achieved an internal nDCG@10 of 0.942 on a 500-article training sample yet produced an official PA-nDCG of only 0.578 on the full 4,912-article test set, a drop of more than 36 points. This gap is unlikely to reflect a bug in the pipeline, since the same encoder achieved consistent Task 1 scores (0.827 internal vs. 0.831 official). A plausible explanation is that the multimodal encoding of images is sensitive to the distribution of visual content in the test set: the small validation sample may have over-represented article types where images provide a discriminative signal. In contrast, the full test set contains a higher proportion of articles with generic or uninformative images, making the multimodal score a source of noise rather than signal. This result underscores the risk of evaluating multimodal systems on small samples and highlights the importance of testing on diverse, large-scale data before concluding multimodal utility.

- **Cost-accuracy tradeoff and the potential of Exp. 1.** An important practical dimension of this work is the cost of each approach. The generative ranker (Exp. 3) requires one API call per article to `gemini-3.1-flash-lite-preview`, which, while efficient for a lightweight model, scales linearly with dataset size and incurs ongoing inference costs. The total API cost for `gemini-3.1-flash-lite-preview` across the full 4,912-article test set was approximately $6.12 USD: text input tokens accounted for $3.61, image input tokens (Task 2) for $1.45, and

text output tokens for $1.06. This confirms that a fully generative multimodal approach remains economically viable at research scale, even when processing nearly 5,000 articles in a single submission. By contrast, Exp. 1 Task 1 — using a local `embeddinggemma` model running via Ollama — requires no API access and would have placed the team **8th out of 20** on the leaderboard based on the text-only task alone. It suggests that with targeted improvements to address the generalization gap in Task 2, an embedding-based system could offer a competitive, cost-effective alternative. Specifically, two directions seem promising: (i) replacing the multimodal `gemini-embedding-2-preview` encoder with a more robust multimodal representation evaluated on a larger validation set, or (ii) applying the query generation step from the development phase — using a generative model to synthesize an ideal headline as the retrieval query — which could narrow the semantic gap without full generative inference on every article.

- **Native multimodal reasoning vs. late fusion.** Comparing Exp. 2 (manual CLIP fusion) with the multimodal path of Exp. 3 (native image+text reasoning by `gemini-3.1-flash-lite-preview`) reveals a substantial quality difference in favor of native reasoning. Manual late fusion assigns a fixed weight (0.2) to the image signal regardless of whether the image is informative for a given article. At the same time, the generative model can adaptively leverage visual context only when it provides additional discriminative evidence, such as recognizing a specific politician, location, or event depicted in the photograph. This flexibility likely accounts for much of the gap between Exp. 2 and Exp. 3 on Task 2.

6. Conclusions

We presented the system submitted by VerbaNex AI Lab to the PoliticHeadlinES shared task at IberLEF 2026, which addresses multimodal headline ranking for Spanish political news. Our experiments trace a progression from sparse TF-IDF retrieval and dense embedding-based rankers to a fully generative approach, and yield three main conclusions:

- **Headline discrimination requires reasoning, not retrieval.** Instruction-following language models that read and compare candidates directly outperform all embedding-based approaches by a wide margin, achieving a PA-nDCG of 0.922 on Task 1 and 0.926 on Task 2 and placing VerbaNex AI Lab 3rd out of 20 teams on the CodaBench leaderboard.

- **Multimodal embeddings do not reliably generalize.** The `gemini-embedding-2-preview` encoder showed a severe performance drop from internal validation to the official test set on Task 2, highlighting the danger of over-relying on small validation samples when evaluating multimodal systems. Native multimodal reasoning within a generative model, by contrast, generalized consistently across both tasks.

- **There is a viable path to a cheaper competitive system.** The local embedding ranker (Exp. 1) placed 8th on Task 1 using only a local model with no API costs. Future work should focus on resolving the Task 2 generalization failure — for instance, through more robust multimodal representations or query generation — to produce a system that approaches the accuracy of the generative ranker at a fraction of the inference cost. It is particularly relevant for large-scale deployment scenarios where per-article API calls are not economically feasible.

Code Availability

The source code of the submitted system is publicly available at https://github.com/VerbaNexAI/IberLEF2026/tree/main/PoliticHeadlinES.

Acknowledgments

We dedicate this work to the master's degree scholarship program in Engineering at the Universidad Tecnológica de Bolívar (UTB) in Cartagena, Colombia.

We want to express our gratitude to the team at the VerbaNex AI Lab [1] for their dedication, collaboration, and ongoing support of our research endeavors.

Declaration on Generative AI

During the preparation of this work, the authors used `gemini-3.1-flash-lite-preview` (Google) as a core component of the generative ranking system described in Section 3.3. Additionally, Claude (Anthropic) was used to assist in writing and editing the manuscript. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] J. Gómez-Navalón, T. Bernal-Beltrán, R. Pan, F. García-Sánchez, J. A. García-Díaz, R. Valencia-García, Overview of PoliticHeadlinES at IberLEF 2026: Multimodal Headline Ranking in Spanish Political News, Procesamiento del Lenguaje Natural 77 (2026).

[2] A. Bonet-Jover, J. Á. González-Barba, L. 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.

[3] R. Pan, T. Bernal-Beltrán, J. Gómez-Navalón, J. A. García-Díaz, R. Valencia-García, Spanish PoliSUM-2025: Evaluating Stylistic and Editorial Fidelity in Abstractive Summarization of Political News, Procesamiento del Lenguaje Natural 76 (2026) 177–189.

[4] J. Cuadrado, E. Martinez, A. Morillo, D. Peña, K. Sossa, J. C. Martinez-Santos, E. Puertas, UTB-NLP at SemEval-2023 task 3: Weirdness, lexical features for detecting categorical framings, and persuasion in online news, in: A. K. Ojha, A. S. Doğruöz, G. Da San Martino, H. Tayyar Madabushi, R. Kumar, E. Sartori (Eds.), Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023), Association for Computational Linguistics, Toronto, Canada, 2023. URL: https://aclanthology.org/2023.semeval-1.214/. doi:10.18653/v1/2023.semeval-1.214.

[5] E. Puertas, J. Vásquez, J. C. Martinez-Santos, Realcheck: A web application for fake news detection using natural language processing, in: 2023 IEEE Colombian Caribbean Conference (C3), 2023, pp. 1–5. doi:10.1109/C358072.2023.10436244.

[6] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, I. Sutskever, Learning Transferable Visual Models From Natural Language Supervision, in: Proceedings of the 38th International Conference on Machine Learning (ICML), 2021, pp. 8748–8763.

[7] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2019. URL: http://arxiv.org/abs/1908.10084.

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, volume 30, 2017, pp. 5998–6008.

[1] https://github.com/VerbaNexAI

[9] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, D. Kiela, Retrieval-augmented generation for knowledge-intensive NLP tasks, in: Advances in Neural Information Processing Systems, volume 33, 2020, pp. 9459–9474.

[10] Google DeepMind Gemma Team, Introducing EmbeddingGemma: The best-in-class open model for on-device embeddings, https://developers.googleblog.com/en/introducing-embeddinggemma/, 2025. Released September 2025. Google Developers Blog.

[11] Google DeepMind Gemma Team, Gemma 4: Our most capable open models to date, https://deepmind.google/models/gemma/gemma-4/, 2026. Released April 2, 2026. Apache 2.0 License.