HEART-UJA-UA at PoliticHeadlinES-IberLEF 2026: Generative and Encoder-Based Approaches for Spanish Political Headline Ranking

Adrián Moreno-Muñoz[1,†], Miquel Canal-Esteve[2,*,†], María-Teresa Martín-Valdivia[1], Yoan Gutiérrez[2], Luis-Alfonso Ureña-López[1], Fernando Llopis[2] and Eugenio Martínez-Cámara[1]

[1]CEATIC, University of Jaén, Campus Las Lagunillas s/n, 23071, Jaén, Spain
[2]University of Alicante, Carretera de San Vicente del Raspeig, s/n San Vicente del Raspeig, Alicante, España

Abstract

This paper presents our participation in PoliticHeadlinES 2026, a shared task on ranking candidate headlines for Spanish political news articles. We compare encoder-based and generative approaches, including direct semantic similarity, a lightweight pairwise neural ranker built on frozen multilingual embeddings, and a retrieval-augmented few-shot prompting pipeline. Our final results show that the generative approach achieves the best overall performance, indicating that in-context ranking with retrieved examples is particularly effective for this task. We also explore multimodal late fusion with CLIP-based image embeddings and different strategies for representing long articles, finding that visual information provides limited benefits and that truncating the article to the first 512 tokens remains a strong baseline for encoder models. Overall, the HEART-UJA-UA team ranked sixth out of 20 participants in the official leaderboard, obtaining PA-nDCG scores of 0.885 in Task 1 and 0.896 in Task 2.

Keywords

Headline ranking, Spanish political news, Generative models, Retrieval-augmented prompting, Learning to rank

1. Introduction

Political news plays a central role in modern societies, shaping how citizens access information, interpret public events, and form opinions. In digital media environments, headlines are particularly influential, since they are often the first, and sometimes the only, element users read before deciding whether to click, share, or engage with a news article. Selecting an appropriate headline is therefore not only a summarization problem, but also a relevance, faithfulness, and framing problem: a headline should be concise and informative while accurately reflecting the article content and avoiding misleading interpretations [1, 2, 3].

The PoliticHeadlinES shared task [4] focuses on ranking candidate headlines for Spanish political news articles. Given a news article and ten candidate headlines, systems must produce a ranking from the most to the least appropriate headline. Unlike headline generation or standard classification, this task requires fine-grained relevance estimation, making it closer to information retrieval and learning-to-rank problems [5]. The task is based on the Spanish PoliSUM-2025 dataset [6]. The shared task is divided into two evaluation settings. In Task 1, systems must rank the candidate headlines using only the article text, whereas in Task 2 they may additionally exploit the image associated with the article.

This paper describes our participation in PoliticHeadlinES. We explored encoder-based ranking methods for both the text-only and multimodal settings. First, we evaluated direct semantic similarity between article and title embeddings. Second, we trained a lightweight pairwise neural ranker on top

IberLEF 2026, September 2026, León, Spain
*Corresponding author.
†These authors contributed equally.
ammunoz@ujaen.es (A. Moreno-Muñoz); mikel.canal@ua.es (M. Canal-Esteve)
0009-0000-8809-8804 (A. Moreno-Muñoz); 0009-0006-8022-5534 (M. Canal-Esteve)
© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2026-09-29

CEUR-WS.org/Vol-4273/PoliticHeadlinES2026_paper3.pdf

of frozen multilingual encoders, using article and title embeddings together with their element-wise difference and product. For the multimodal subtask, we incorporated article images through late fusion with CLIP-based visual embeddings. We also compared different strategies for representing long articles, including truncation, chunking, and weighted pooling.

In addition, we evaluated a generative ranking approach based on few-shot in-context learning. In this setting, instruction language models receive the article, the ten candidate headlines, and a small set of retrieved training examples with their gold rankings. This allows the model to compare candidates according to task-oriented criteria such as fidelity, specificity, and coherence, providing a complementary alternative to encoder-based scoring.

Our results show that a neural ranker trained on top of `intfloat/multilingual-e5-large` clearly outperforms direct semantic similarity. In contrast, adding visual information did not improve the official score, suggesting that the textual signal was more informative than the image representation in this setting. Finally, the simplest strategy, encoding only the first 512 tokens of the article, performed better than more complex chunk-based alternatives. Among the evaluated generative models, `gemma-4-31B-it` obtained the best validation results, while `Qwen3.6-27B` was selected as a competitive alternative.

2. Related Work

2.1. Encoder-based Approach

The task addressed in this work can be framed as a ranking problem, where a system must order a fixed set of candidate titles according to their relevance to a given news article. In our setting, the main challenge is to learn an effective relevance function between an article and a set of candidate titles, making the problem closely related to neural information retrieval, semantic textual similarity, and learning-to-rank.

Early neural approaches to sentence-level semantic matching relied on dense vector representations to compare textual units in a shared embedding space. Sentence-BERT introduced a siamese and triplet-network architecture on top of BERT to obtain semantically meaningful sentence embeddings that can be efficiently compared using cosine similarity [7]. This line of work is particularly relevant to our first baseline, where articles and titles are encoded independently and ranked according to semantic similarity. More recently, dense retrieval models based on pretrained language models have become a dominant approach in information retrieval, as they allow queries and documents to be represented as dense vectors and compared efficiently in an embedding space [8]. In this context, the E5 family of models has shown strong multilingual embedding performance and provides open-source encoders specifically designed for query–passage matching scenarios [9]. This motivates the use of `intfloat/multilingual-e5-large` as one of the main textual encoders in our experiments.

However, direct cosine similarity may be insufficient when the target task requires a specific notion of relevance. Learning-to-rank methods address this limitation by training models to optimize the ordering of candidate items rather than treating each candidate independently. RankNet introduced a pairwise neural learning-to-rank framework in which the model learns preferences between pairs of documents [10]. Subsequent work systematized learning-to-rank approaches for information retrieval, distinguishing between pointwise, pairwise, and listwise formulations [11]. Our neural ranking model follows this pairwise perspective: given a gold ordering of titles for each article, the model is trained to assign higher scores to better-ranked titles than to lower-ranked ones. This allows the system to learn task-specific interactions between article and title embeddings beyond raw semantic similarity.

Since the task also provides images associated with the articles, our work is also connected to multimodal representation learning. CLIP demonstrated that visual representations can be learned from natural language supervision by training on large-scale image–text pairs, enabling strong zero-shot transfer across visual tasks [12]. This makes CLIP a natural choice for extracting image embeddings in multimodal news ranking scenarios. More broadly, multimodal machine learning studies how information from different modalities can be represented, aligned, and fused [13]. In our experiments,

we adopt a late fusion strategy, combining frozen text and image embeddings at the representation level. This design is simple, computationally efficient, and compatible with independently pretrained encoders.

Overall, previous work supports the main design choices explored in this paper: dense textual embeddings provide an efficient basis for semantic ranking, pairwise learning-to-rank models can adapt generic embeddings to a task-specific relevance function, and multimodal late fusion offers a straightforward mechanism for incorporating image information. Our contribution is to empirically compare these alternatives in the specific setting of candidate title ranking, showing that a frozen multilingual E5 encoder combined with a lightweight pairwise neural ranker outperforms direct semantic similarity, while image-based late fusion provides limited benefits in the evaluated configuration.

2.2. Few-Shot Learning and In-Context Learning

Large language models have shown strong few-shot learning abilities, allowing them to perform new tasks from a small number of demonstrations included directly in the prompt, without updating model parameters [14]. This property is useful for headline ranking, where the model can be shown examples of articles, candidate headlines, and expected rankings before solving a new instance.

However, in-context learning is sensitive to the demonstrations provided. [15] showed that selecting examples semantically close to the test instance can outperform random example selection. This motivates retrieval-based few-shot prompting, where similar training examples are retrieved and included in the prompt to provide task-relevant guidance. For PoliticHeadlinES, this is especially appropriate because headline quality depends on subtle contextual factors such as faithfulness, specificity, framing, and omission of salient details.

2.3. Prompt Engineering and Structured Reasoning

Prompt design is important when applying generative models to ranking tasks. Since the model must compare ten candidate headlines simultaneously, an underspecified prompt may lead to unstable outputs, incomplete rankings, repeated identifiers, or decisions based mainly on superficial lexical overlap. Previous surveys have shown that prompt engineering is a central mechanism for adapting large language models to downstream tasks without modifying their parameters [16]. A structured prompt can reduce these problems by defining the expected output format and the criteria that should guide the ranking decision.

In headline ranking, criteria such as fidelity, specificity, and coherence are particularly relevant. Fidelity requires that the headline accurately reflect the article without introducing unsupported claims; specificity favors headlines that capture salient details; and coherence refers to whether the headline provides a consistent interpretation of the article's main event. Prior work on chain-of-thought prompting has shown that intermediate reasoning patterns can improve the ability of large language models to solve complex tasks [17]. Although the proposed system does not require long explanations in the output, resolved demonstrations can help establish the expected comparison behavior.

2.4. Retrieval-Augmented Generation for Ranking

The proposed generative approach is also related to retrieval-augmented generation. Retrieval-augmented methods combine a generative model with a retrieval component, allowing the model to condition its output on relevant retrieved information rather than relying only on its parametric knowledge [18]. In this setting, the retrieved items are not external factual passages, but labeled training examples that illustrate how similar headline-ranking decisions were resolved.

This differs from traditional retrieval-augmented question answering, but follows the same general principle: retrieval provides task-specific context, while generation performs the final reasoning step. The retrieved examples act as demonstrations that adapt the model to the ranking task at inference time, which is useful when no task-specific fine-tuning is performed. When several retrieval signals

are available, rank aggregation methods such as Reciprocal Rank Fusion can also be used to combine rankings from different similarity criteria [19].

2.5. Hybrid Approach: Retrieval and Generation

The proposed approach follows a hybrid design that combines retrieval of informative examples with generative reasoning over the article and its candidate headlines. This differs from encoder-only ranking, since the generative model does not simply score fixed vector representations, but compares candidates in context according to qualitative criteria such as faithfulness, specificity, coherence, and possible framing effects.

Recent work has shown that large language models can be used as zero-shot or prompt-based rankers, although their performance may depend on prompt formulation, candidate order, and position bias [20, 21]. This is relevant for headline ranking, where the model must compare several plausible candidates and return a complete ordered list. In this way, the generative approach complements encoder-based ranking methods, especially in scenarios where explicit fine-tuning is limited or where ranking decisions depend on nuanced natural-language criteria.

3. Methodology

In this section, we describe the dataset, model architectures, and evaluation setup used in our experiments on candidate headline ranking for Spanish political news. We consider both encoder-based and generative approaches, including multimodal variants that incorporate article images, and we examine how different representation and inference strategies affect performance under the official shared-task setting. We also detail the retrieval, prompting, and post-processing components used in the generative pipeline, together with the evaluation metric adopted in PoliticHeadlinES 2026.

3.1. Dataset

The PoliticHeadlinES dataset consists of Spanish political news articles, each associated with a fixed set of ten candidate headlines. For each article, systems must rank the candidate headlines from the most to the least appropriate one. During the development phase, the organizers provided a small dataset with 100 training examples and 50 test examples, which was used for preliminary experiments and model comparison. For the official evaluation phase, the training set contained 16,030 examples, while the official test set contained 4,912 articles.

Each training instance includes the article body, the associated image identifier, the ten candidate headlines, and the gold ranking used for supervision. In contrast, the test instances provide the article body, image identifier, and candidate headlines, but not the gold ranking.

3.2. Encoder-based Ranking Approaches

We explored two encoder-based approaches for ranking candidate titles given an input article. The first approach relies on direct semantic similarity: both the article and the candidate titles are encoded into a shared embedding space, and titles are ranked according to their cosine similarity with the article representation. The second approach uses the same frozen encoder representations, but trains a lightweight neural ranking model to score each article–title pair.

In both cases, the article body was encoded using the first 512 tokens, without overlapping chunks. This limit was selected because 512 tokens correspond to the maximum context window supported by some of the evaluated encoders, such as `intfloat/multilingual-e5-large`. Candidate titles were encoded independently using a maximum length of 69 tokens, which corresponds to the length of the longest title in the dataset when tokenized with the E5 tokenizer, and is also sufficient to cover all titles across the three evaluated models.

For the neural approach, we used a pairwise learning-to-rank formulation. Given an article and its gold ranking of candidate titles, the model was trained to assign higher scores to better-ranked titles than to lower-ranked ones. For each article–title pair, the ranking model receives the article embedding, the candidate title embedding, their element-wise absolute difference, and their element-wise product. These features allow the model to capture both semantic similarity and interaction patterns between the two representations. The ranking head is implemented as a multilayer perceptron trained with a margin ranking loss, while the encoder itself remains frozen.

Table 1 compares the direct similarity approach against the trained neural ranker on the development set, using 100 training examples and 50 validation examples.

Model		
textbfSemantic Similarity	**Neural Ranker**	
`intfloat/multilingual-e5-large`	0.920	**0.968**
`google-bert/bert-base-multilingual-uncased`	0.580	0.934
`SINAI/ALIA-MrBERT-es-legal-embeddings`	0.880	0.962

Table 1
Comparison of encoder models using direct semantic similarity and a trained neural ranking model on the development set for Task 1.

The results show that training a neural ranking model consistently improves over direct semantic similarity. The best overall configuration was obtained with `intfloat/multilingual-e5-large` combined with the neural ranker. For this reason, future experiments will be conducted with `intfloat/multilingual-e5-large`.

It is worth noting that `SINAI/ALIA-MrBERT-es-legal-embeddings` is an internally trained model that is not publicly available at the time of writing.

3.2.1. Multimodal Late Fusion

We also evaluated whether incorporating visual information from the article image could improve ranking performance. To this end, we adopted a late fusion strategy at the embedding level. The article body and candidate titles were encoded using the frozen text encoder `intfloat/multilingual-e5-large`. In parallel, the associated article image was encoded using a frozen visual encoder based on `openai/clip-vit-large-patch14`. Both text and image embeddings were L2-normalized during precomputation before being combined in the fusion step.

The final article representation was computed as a weighted linear combination of the textual and visual embeddings:

$$\mathbf{h}_{article} = w_t \mathbf{h}_{text} + w_i \mathbf{h}_{image},$$

where w_t and w_i denote the contribution of the textual and visual modalities, respectively. We evaluated three fusion configurations: $(w_t = 1, w_i = 0)$, $(w_t = 0.9, w_i = 0.1)$, $(w_t = 0.7, w_i = 0.3)$, and $(w_t = 0.5, w_i = 0.5)$.

3.2.2. Text Truncation and Pooling Strategies

Finally, we compared several strategies for representing long article bodies. Besides the baseline strategy of using only the first 512 tokens, we explored chunk-based representations with and without overlap. When multiple chunks were produced, their embeddings were combined either using standard mean pooling or weighted mean pooling. In the latter case, higher weights were assigned to earlier chunks, giving more importance to the beginning of the article, which typically contains the most informative content. Concretely, we used an exponentially decaying weighting scheme with a decay factor of 0.85, where the weight of the i-th chunk is proportional to 0.85^i, followed by normalization to ensure a convex combination.

All experiments in this section were conducted using the best-performing multimodal configuration identified in our preliminary experiments, with fusion weights $w_t = 0.9$ and $w_i = 0.1$, where the contribution of the image modality is minimal.

3.3. Generative Ranking Approach

In addition to the encoder-based ranking models, we evaluated a generative approach based on few-shot in-context learning. Instead of training a task-specific ranking head, the system prompts a large language model to directly order the ten candidate headlines for each article. The model receives the article body, the candidate headlines, and a small set of retrieved examples from the training data. The objective is to exploit the model's ability to compare candidates according to qualitative criteria such as fidelity, specificity, and coherence.

The generative system was implemented as a retrieval-based few-shot pipeline. For each input article, the system first retrieves similar training instances. These examples are then inserted into the prompt together with their gold rankings, allowing the model to observe the expected input-output format before ranking the target instance. Finally, the model output is post-processed to ensure that it is a valid complete ranking containing each identifier from `t1` to `t10` exactly once.

3.3.1. Dynamic Few-Shot Example Selection

For each target article, the system selects a fixed number of similar examples from the training set. Each training example contains the article body, the ten candidate headlines, and the corresponding gold ranking. In the final configuration, we used three retrieved examples per target instance.

We considered similarity-based retrieval in order to avoid using the same demonstrations for all inputs. This makes the few-shot context dependent on the article being ranked. The retrieval module supports lexical similarity based on token overlap, semantic similarity based on text embeddings, and a hybrid ranking strategy. In the submitted configuration, semantic similarity was used. Article embeddings were computed for the training examples and cached, so that at inference time the system only needed to embed the target article and retrieve the most similar training instances.

This dynamic selection is intended to provide the model with examples that are closer to the target article in topic and structure. As a result, the demonstrations can guide the model toward the type of distinctions required in the task, such as identifying unsupported claims, overly generic headlines, incorrect entities, or headlines that omit the central event.

3.3.2. Prompt Construction

The complete prompt templates, few-shot format, inference configuration, and limitations of the generative approach are reported in Appendix A.

The prompt is composed of two parts: a system instruction and a user message. The system instruction defines the role of the model as an expert editor in Spanish political news and establishes the ranking criteria. The model is instructed to prioritise fidelity to the article, specificity of the selected facts, semantic coherence, and the penalisation of misleading or unsupported headlines.

The user message contains the article body and the ten candidate headlines, labelled from `t1` to `t10`. The retrieved examples are inserted before the target instance. Each example includes the article, its ten candidate headlines, and the correct ranking. The target case is then presented using the same format, so that the model must produce only the ordered sequence of headline identifiers.

The output format is strictly constrained. The model is instructed to return a single line containing the ten identifiers, separated by spaces, without explanations or additional text. This constraint is important because the official evaluation requires a complete ranking for each article.

3.3.3. Text-only and Multimodal Inference

We evaluated the generative approach in both task settings. For Task 1, only the article text and the candidate headlines are provided to the model. For Task 2, the system can additionally include the article image. In this case, the image is loaded from the image directory and sent together with the textual prompt.

The multimodal prompt follows the same general structure as the text-only prompt, but it additionally asks the model to consider visual coherence. The model is instructed to internally analyse the image, identify relevant persons, places, actions, or symbols, and integrate this information with the article content before producing the final ranking. Nevertheless, the final answer remains constrained to the same format: a single ordered sequence from `t1` to `t10`.

All experiments with the generative model were run with deterministic decoding, using temperature 0.0 and top-p 1.0. Thinking output was disabled, and the prompt explicitly required the model to return only the final ranking.

Several instruction generative models were evaluated under the same prompting and post-processing pipeline. Table 2 shows the validation scores obtained for both tasks. Based on these validation results, `gemma-4-31B-it` was selected as the strongest generative configuration, while `Qwen3.6-27B` was retained as a competitive alternative from the Qwen family.

Model	Task 1	Task 2
`gpt-oss-120b`	0.895	–
`qwen3.5-122b-a10b-fp8`	0.916	–
`qwen3.5-35b-a3b`	0.916	0.796
`qwen3.6-35b-a3b`	0.935	0.935
`gemma-4-e4b-it`	0.836	0.915
`gemma-4-31B-it`	**0.955**	**0.955**
`Qwen3.6-27B`	0.935	0.935

Table 2
Comparison of the evaluated generative models. "–" indicates that the model did not support the corresponding setting or that it was not used for that task.

3.3.4. Output Validation and Normalisation

Since generative models may occasionally produce additional text, repeated identifiers, missing titles, or malformed outputs, a post-processing step was applied to every model response. The system extracts valid ranking tokens matching the pattern `t1` to `t10`, removes duplicates while preserving order, and appends any missing identifiers at the end. This guarantees that every prediction contains exactly ten unique headline identifiers.

If the model fails to return a usable ranking, the system falls back to a lexical ranking based on token overlap between the article and each candidate headline. This fallback ensures full coverage of the evaluation set. The final predictions are saved incrementally to a CSV file and later compressed into the required submission format.

3.4. Evaluation

For each article, the system must output a complete ranking of the ten candidate titles, represented as an ordered sequence of identifiers from `t1` to `t10`. Task 1 evaluates the ranking produced using textual information only, whereas Task 2 evaluates the ranking obtained when both textual and visual information may be used.

The official metric is a position-aware normalized Discounted Cumulative Gain, denoted as PA-nDCG@k, with $k = 10$ and $\alpha = 0.9$. Unlike standard nDCG, this metric imposes a hard constraint on the first ranked title: if the top-ranked predicted title does not match the first title in the gold ranking, the score for that example is set to zero. If the top prediction is correct, the score is computed as a

weighted combination of this top-1 match and an auxiliary nDCG score over the remaining ranked titles:

$$\text{PA-nDCG@}k = \begin{cases} 0, & \text{if } \hat{y}_1 \neq y_1, \\ \alpha + (1-\alpha) \cdot \text{nDCG}_{\text{aux}}@k, & \text{if } \hat{y}_1 = y_1. \end{cases}$$

Here, $\hat{y}_1$ and y_1 denote the first predicted and gold-ranked titles, respectively. The auxiliary nDCG is computed after removing the correctly predicted top title from both rankings. Therefore, the metric strongly prioritizes selecting the best headline in the first position, while still rewarding the relative ordering of the remaining candidate titles.

4. Results

This section presents the main empirical findings for both the encoder-based and generative approaches evaluated in this work. We first analyse the encoder results, including the comparison between direct similarity and neural ranking, the effect of multimodal fusion, and different strategies for handling long articles. We then report the official results of the generative systems and examine how model choice and retrieval embeddings influence final performance across the two shared-task settings.

4.1. Encoder-based Approach

4.1.1. Impact of Multimodal Fusion

Table 3 reports the Task 1 and Task 2 results obtained with different text–image fusion weights.

Fusion Weights	Task 1	Task 2
$w_t = 1,\ w_i = 0$	0.695	0.695
$w_t = 0.9,\ w_i = 0.1$	0.711	**0.709**
$w_t = 0.7,\ w_i = 0.3$	**0.713**	0.706
$w_t = 0.5,\ w_i = 0.5$	0.695	0.666

Table 3
Impact of different text–image fusion weights on Task 1 and Task 2 performance.

Although the validation scores are very similar across configurations, the official results reveal a clearer effect of the fusion weights. In particular, assigning a small but non-zero weight to the image modality yields the best performance, whereas both removing the image signal entirely and increasing its contribution lead to worse official scores. This suggests that visual information can provide a limited but useful complementary signal when carefully weighted, while excessive reliance on the image representation may introduce noise. Therefore, the final system prioritizes the textual modality while retaining a small visual contribution.

4.1.2. Effect of Truncation and Pooling

Table 4 summarizes the results obtained with different truncation and pooling strategies for Task 1 and Task 2.

The results show that the best-performing strategy was the simplest one: encoding only the first 512 tokens of the article. Chunk-based approaches did not improve the official score, even when they achieved similar validation performance. However, when chunking was used, weighted mean pooling consistently outperformed standard mean pooling, suggesting that weighting chunks can partially mitigate the loss introduced by splitting the article into multiple segments.

Overall, these experiments indicate that the most effective configuration is a frozen `intfloat/multilingual-e5-large` encoder combined with a neural pairwise ranking head, using only the first 512 tokens of the article body and minimal image contribution.

Configuration	Task 1	Task 2
First 512 tokens	**0.711**	**0.709**
Overlap, token-based truncation, mean pooling	0.672	0.668
Overlap, token-based truncation, weighted mean pooling	0.693	0.692
No overlap, sentence-based truncation, mean pooling	0.671	0.665
No overlap, sentence-based truncation, weighted mean pooling	0.693	0.692

Table 4
Comparison of truncation and pooling strategies for Task 1 and Task 2.

4.2. Generative Approach

4.2.1. Generative Model Comparison

Table 5 compares the official results obtained with the submitted generative systems. We evaluated two instruction generative models, `gemma-4-31B-it` and `Qwen3.6-27B`, combined with two retrieval embedding configurations. The first configuration used `Qwen3-Embedding-0.6B` for both the textual and visual settings. The second configuration used `Qwen3-Embedding-8B` for textual retrieval and `Qwen3-VL-Embedding-8B` for the visual setting.

Model	Embedding Configuration	Task 1	Task 2
`gemma-4-31B-it`	`Qwen3-Emb-0.6B`	0.889	0.888
`Qwen3.6-27B`	`Qwen3-Emb-0.6B`	**0.897**	0.887
`gemma-4-31B-it`	`Qwen3-Emb-8B + Qwen3-VL-Emb-8B`	0.891	**0.895**
`Qwen3.6-27B`	`Qwen3-Emb-8B + Qwen3-VL-Emb-8B`	0.896	0.885

Table 5
Official results of the submitted generative systems. `Qwen3-Emb-0.6B` denotes the use of `Qwen3-Embedding-0.6B` in both the textual and visual settings. `Qwen3-Emb-8B + Qwen3-VL-Emb-8B` denotes the use of `Qwen3-Embedding-8B` for text and `Qwen3-VL-Embedding-8B` for visual information.

The results show that the best Task 1 score was obtained with `Qwen3.6-27B` using `Qwen3-Embedding-0.6B`, reaching 0.897. However, since the organizers indicated Task 2 as the main task, the final selected configuration was the one with the highest multimodal score. This was achieved by `gemma-4-31B-it` combined with `Qwen3-Embedding-8B` for the textual component and `Qwen3-VL-Embedding-8B` for the visual component, obtaining 0.895 on Task 2.

4.2.2. Impact of Retrieval Embeddings

The comparison also shows that the impact of the retrieval embedding configuration depends on the generative model. For `gemma-4-31B-it`, replacing `Qwen3-Embedding-0.6B` with the larger textual embedding model and the dedicated vision-language embedding model improved Task 2 performance from 0.888 to 0.895. This suggests that, for this model, stronger retrieval representations helped select more informative few-shot examples in the multimodal setting.

In contrast, `Qwen3.6-27B` achieved its best Task 1 score with `Qwen3-Embedding-0.6B`, while the larger embedding configuration slightly reduced its Task 2 score from 0.887 to 0.885. Therefore, the larger embedding models did not provide a uniform improvement across all generative models. The gains were most evident for `gemma-4-31B-it`, particularly in the task prioritized by the organizers.

4.2.3. Final Generative Configuration

Overall, the submitted generative systems achieved competitive results across both tasks, with small differences between configurations. The best text-only result was obtained by `Qwen3.6-27B`, whereas the best multimodal result was obtained by `gemma-4-31B-it`. Since Task 2 was considered the main task, we selected `gemma-4-31B-it` with `Qwen3-Embedding-8B` and `Qwen3-VL-Embedding-8B` as the final generative system.

4.3. Comparison with Other Participants

In the official leaderboard, the best HEART-UJA-UA submission obtained PA-nDCG scores of 0.885 in Task 1 and 0.896 in Task 2, ranking sixth among 21 submissions from 20 participating teams. These results place our system in the upper part of the leaderboard and show that retrieval-augmented few-shot prompting is an effective alternative for headline ranking without requiring the training of a task-specific ranking head. Instead, the approach relies on retrieved demonstrations, structured prompts, and output normalisation.

The relatively strong performance in Task 2 is especially relevant, since this was considered the main task by the organisers. At the same time, the gap with the top-ranked systems suggests that further improvements could be achieved through stronger multimodal modelling, more robust calibration of generative rankings, or task-specific supervised training.

Task 2 Rank	Team	Task 1 Rank	Task 1	Task 2 Rank	Task 2
1	Theron	1	0.954	1	0.954
2	LKE-BUAP	2	0.928	2	0.928
3	VerbaNex AI Lab	3	0.922	3	0.926
4	acho_gpt	4	0.903	4	0.903
5	GACOP	5	0.897	5	0.897
6	**HEART-UJA-UA**	**6**	**0.895**	**6**	**0.891**
7	franrodrigo	7	0.884	7	0.884
8	I2C-UHU	8	0.826	8	0.826
9	francordel	10	0.807	9	0.813
10	LYS-UDC	9	0.824	10	0.808
11	ITO	11	0.799	11	0.796
12	UTP	12	0.739	12	0.770
13	UC-UCO-Plenitas	13	0.705	13	0.711
14	LACELL	14	0.660	14	0.659
15	enrique_lr	15	0.656	15	0.656
16	UIE	16	0.645	16	0.645
17	I2C-UHU-Perseus	17	0.623	17	0.623
18	PUCE-UTP	18	0.581	18	0.555
19	baseline	19	0.517	19	0.513
20	MIMIC	20	0.348	20	0.348

Table 6
Official PoliticHeadlinES leaderboard comparison. Scores are reported on a 0–1 scale and the table is ordered by the official Task 2 ranking.

5. Conclusions

This paper presented our participation in the PoliticHeadlinES shared task on headline ranking for Spanish political news. We explored encoder-based approaches that frame the task as a ranking problem, comparing direct semantic similarity against a lightweight neural ranker trained on top of frozen multilingual embeddings.

Our encoder-based show that adapting frozen sentence embeddings with a pairwise neural ranking head clearly improves over direct cosine similarity. The best configuration was obtained with `intfloat/multilingual-e5-large`, using the first 512 tokens of the article and a maximum title length of 69 tokens. We also evaluated multimodal late fusion with CLIP-based image embeddings, but the contribution of visual information was limited: increasing the image weight degraded the official score, suggesting that the textual signal was the most informative component in this setting.

We compared several strategies for representing long articles. The simplest approach, based on truncating the article to the first 512 tokens, outperformed more complex chunking and pooling methods. These findings suggest that, for this task, strong multilingual text encoders combined with a lightweight

learning-to-rank model offer an effective and computationally efficient solution. Future work could explore more task-specific multimodal alignment strategies and cross-encoder architectures capable of modeling article–headline interactions more deeply.

In the generative setting, we evaluated several instruction models combined with different retrieval embedding configurations. The best Task 1 result was obtained with `Qwen3.6-27B` using `Qwen3-Embedding-0.6B`. However, since the organizers identified Task 2 as the main task, our final selected generative configuration was `gemma-4-31B-it` combined with `Qwen3-Embedding-8B` for textual retrieval and `Qwen3-VL-Embedding-8B` for visual retrieval. This system achieved the best multimodal score among our submitted generative runs.

Overall, our findings suggest that strong multilingual text representations remain highly effective for Spanish political headline ranking, especially when combined with a task specific ranking head. At the same time, retrieval augmented few-shot prompting provides a competitive alternative that does not require training a dedicated ranking model and can explicitly compare candidates according to qualitative criteria such as fidelity, specificity, and coherence. Future work could explore more task-specific multimodal alignment strategies, cross-encoder architectures capable of modeling article headline interactions more deeply, and more robust prompting or calibration techniques to reduce the sensitivity of generative rankers to candidate order and retrieved examples. The code developed for this work is publicly available at https://github.com/gplsi/PoliticHeadlinES-HEART-UJA_UA.

6. Limitations

Limitations of the embedding-based approach:

1. The embedding-based approach relies on fixed vector representations of the article and the candidate headlines. As a result, subtle interactions between specific parts of the article and individual headline candidates may be lost, especially when relevance depends on fine-grained details, entity mismatches, or nuanced framing differences.
2. Encoder models are also constrained by their context window. In our experiments, representing the article with only the first 512 tokens proved effective, but this strategy may omit relevant information appearing later in the document. More complex chunking and pooling strategies can mitigate this issue, although in our setting they did not consistently translate into better ranking performance.
3. The multimodal embedding setting remains challenging. Although Task 2 allows the use of the associated image, the visual signal is not always informative for selecting the most appropriate headline, and simple late-fusion strategies may be insufficient to capture deeper cross-modal interactions.

Limitations of the generative approach:

1. The method is sensitive to prompt formulation. Small changes in the instructions, output constraints, or ordering of candidates may affect the final ranking. Although the prompt explicitly states that the order of presentation should not be interpreted as relevance, generative models may still exhibit position bias when comparing multiple candidates.
2. The retrieval based few-shot strategy depends on the quality of the selected examples. If the retrieved demonstrations are only superficially similar to the target article, they may provide weak or even misleading guidance.
3. The post-processing step guarantees valid output format but cannot guarantee semantic correctness. When the model returns a poor ranking, normalization can only repair missing or repeated identifiers, not the underlying ranking decision.

Declaration on Generative AI

Generative AI tools were used solely to support the revision of grammar, wording, and overall language quality of the manuscript. They were not used to generate scientific content, conduct the experiments, analyze the results, or draw conclusions.

7. Acknowledgements

This work is mainly funded by Project HEART-NLP-UJA (PID2024-156263OB-C21), funded by MICIU/AEI/10.13039/501100011033 and by ERDF/EU. This work has also been partially supported by the *Ministerio para la Transformación Digital y de la Función Pública* and *Plan de Recuperación, Transformación y Resiliencia* - Funded by EU – NextGenerationEU within the framework of the project *Desarrollo Modelos ALIA*; Project VERITAS-H (AIA2025-163322-C64), funded by MICIU/AEI/10.13039/501100011033 and by ERDF/EU; Project CONSENSO (PID2021-122263OB-C21) funded by MCIN/AEI/10.13039/501100011033 and by the European Union NextGenerationEU/PRTR, Project ROMANET (CERV-2024-CHAR-LITI-101215052), funded by the European Union under the Citizens, Equality, Rights and Values programme; Project GALENO-IA (DGP_PIDI_2024_00852) funded by *Junta de Andalucía*.

References

[1] E. Ifantidou, Newspaper headlines, relevance and emotive effects, Journal of Pragmatics 218 (2023) 17–30. doi:10.1016/j.pragma.2023.09.013.

[2] B. Araújo, H. Prior, Framing political populism: The role of media in framing the election of jair bolsonaro, Journalism Practice 15 (2021) 226–242. doi:10.1080/17512786.2019.1709881.

[3] L. Guo, C.-C. Su, H.-T. Chen, Do news frames really have some influence in the real world? a computational analysis of cumulative framing effects on emotions and opinions about immigration, The International Journal of Press/Politics 30 (2025) 211–231. doi:10.1177/19401612231198271.

[4] J. Gómez-Navalón, T. Bernal-Beltrán, R. Pan, G.-S. Francisco, 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).

[5] J. Lee, G. Bernier-Colborne, T. Maharaj, S. Vajjala, Methods, applications, and directions of learning-to-rank in NLP research, in: Findings of the Association for Computational Linguistics: NAACL 2024, Association for Computational Linguistics, 2024, pp. 1900–1917. doi:10.18653/v1/2024.findings-naacl.148.

[6] 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.

[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 and the 9th International Joint Conference on Natural Language Processing, Association for Computational Linguistics, 2019, pp. 3982–3992.

[8] W. X. Zhao, J. Liu, R. Ren, J.-R. Wen, Dense text retrieval based on pretrained language models: A survey, arXiv preprint arXiv:2211.14876 (2022).

[9] L. Wang, N. Yang, X. Huang, L. Yang, R. Majumder, F. Wei, Multilingual E5 text embeddings: A technical report, arXiv preprint arXiv:2402.05672 (2024).

[10] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, G. Hullender, Learning to rank using gradient descent, in: Proceedings of the 22nd International Conference on Machine Learning, 2005, pp. 89–96.

[11] T.-Y. Liu, Learning to rank for information retrieval, Foundations and Trends in Information Retrieval 3 (2009) 225–331.

[12] 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, PMLR, 2021, pp. 8748–8763.

[13] T. Baltrušaitis, C. Ahuja, L.-P. Morency, Multimodal machine learning: A survey and taxonomy, IEEE Transactions on Pattern Analysis and Machine Intelligence 41 (2019) 423–443.

[14] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners, in: Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20, Curran Associates Inc., Red Hook, NY, USA, 2020.

[15] J. Liu, D. Shen, Y. Zhang, B. Dolan, L. Carin, W. Chen, What makes good in-context examples for GPT-3?, in: E. Agirre, M. Apidianaki, I. Vulić (Eds.), Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures, Association for Computational Linguistics, Dublin, Ireland and Online, 2022, pp. 100–114. URL: https://aclanthology.org/2022.deelio-1.10/. doi:10.18653/v1/2022.deelio-1.10.

[16] P. Sahoo, A. K. Singh, S. Saha, V. Jain, S. Mondal, A. Chadha, A systematic survey of prompt engineering in large language models: Techniques and applications, 2025. URL: https://arxiv.org/abs/2402.07927. arXiv:2402.07927.

[17] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. H. Chi, Q. V. Le, D. Zhou, Chain-of-thought prompting elicits reasoning in large language models, in: Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22, Curran Associates Inc., Red Hook, NY, USA, 2022.

[18] 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: Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20, Curran Associates Inc., Red Hook, NY, USA, 2020.

[19] G. V. Cormack, C. L. A. Clarke, S. Buettcher, Reciprocal rank fusion outperforms condorcet and individual rank learning methods, in: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '09, Association for Computing Machinery, New York, NY, USA, 2009, p. 758–759. URL: https://doi.org/10.1145/1571941.1572114. doi:10.1145/1571941.1572114.

[20] Y. Hou, J. Zhang, Z. Lin, H. Lu, R. Xie, J. McAuley, W. X. Zhao, Large language models are zero-shot rankers for recommender systems, 2024. URL: https://arxiv.org/abs/2305.08845. arXiv:2305.08845.

[21] S. Sun, S. Zhuang, S. Wang, G. Zuccon, An investigation ofnbsp;prompt variations fornbsp;zero-shot llm-based rankers, in: Advances in Information Retrieval: 47th European Conference on Information Retrieval, ECIR 2025, Lucca, Italy, April 6–10, 2025, Proceedings, Part II, Springer-Verlag, Berlin, Heidelberg, 2025, p. 185–201. URL: https://doi.org/10.1007/978-3-031-88711-6_12. doi:10.1007/978-3-031-88711-6_12.

A. Generative Prompt Templates and Configuration

This appendix reports the prompt templates and inference configuration used in the generative ranking approach. The goal is to make the generative experiments as reproducible as possible. The prompts were designed to enforce a constrained ranking format and to guide the model toward headline selection based on fidelity, specificity, semantic coherence, and, in the multimodal setting, visual coherence.

A.1. Task 1: Text-only Prompt

For Task 1, the model receives only the article text and the ten candidate headlines. The prompt instructs the model to rank the candidates from the most to the least appropriate headline.

A.1.1. System Prompt

```
Eres un editor experto en política española y en verificación de titulares.

Tu tarea es ordenar 10 titulares candidatos del más relevante al menos
relevante para un artículo periodístico.

Criterios de ranking, en este orden:
1. Fidelidad: el titular debe reflejar con precisión lo que dice el artículo.
2. Especificidad: prioriza el titular que recoge mejor los hechos, actores,
   acciones y contexto concretos.
3. Coherencia semántica: el titular debe encajar con el tema central, no solo
   con palabras sueltas.
4. Penalización de errores: baja en el ranking titulares que:
   - Nombren a una persona, partido o institución diferente a la del artículo.
   - Atribuyan una acción que el artículo no menciona explícitamente.
   - Usen lenguaje valorativo ("escándalo", "fracaso", "polémica") sin
     respaldo textual directo.
   - Capturen solo el tema general pero omitan el hecho concreto relatado.
5. Coherencia visual (solo si se proporciona imagen): si la imagen muestra a
   una persona, lugar o acción concreta, prioriza titulares que coincidan con
   ese contenido visual frente a los que lo ignoran o contradicen.

Reglas de salida:
- Devuelve solo una secuencia de 10 tokens separados por espacios.
- Usa cada token t1, t2, ..., t10 exactamente una vez.
- El primer token debe ser el titular más relevante y el último el menos
  relevante.
- No devuelvas explicaciones, signos de puntuación, texto adicional ni saltos
  de línea.

Ejemplo de salida válida:
t3 t1 t9 t4 t2 t10 t8 t7 t6 t5
```

A.1.2. User Prompt Template

```
### ARTÍCULO ###
{article}

### TITULARES CANDIDATOS ###
IMPORTANTE: El orden de aparición de los titulares NO indica su relevancia.
Evalúa cada uno de forma independiente.

t1: {title_1}
t2: {title_2}
t3: {title_3}
t4: {title_4}
t5: {title_5}
```

```
t6: {title_6}
t7: {title_7}
t8: {title_8}
t9: {title_9}
t10: {title_10}

Ordena los titulares del más relevante al menos relevante.

Recuerda:
- Prioriza el titular más fiel y específico al contenido real del artículo.
- Penaliza titulares incompletos, engañosos, sensacionalistas o que atribuyan
  hechos no mencionados.
- El orden de aparición arriba no implica relevancia: evalúa cada titular de
  forma independiente.
- Devuelve solo los tokens t1..t10 en orden, una sola vez cada uno, en una
  única línea.
```

A.2. Task 2: Multimodal Prompt

For Task 2, the model receives the article text, the ten candidate headlines, and the associated image. The prompt explicitly asks the model to reason internally over both textual and visual evidence before producing the final ranking. However, the final answer is still constrained to a single sequence of headline identifiers.

A.2.1. System Prompt

```
Eres un editor experto en política española y en verificación de titulares.

Tu tarea es ordenar 10 titulares candidatos del más relevante al menos
relevante para un artículo periodístico, usando tanto el texto del artículo
como la imagen asociada.

Proceso de razonamiento obligatorio (hazlo internamente antes de responder):
PASO 1 — Analiza la imagen: identifica personas, lugares, acciones, símbolos
o elementos visuales relevantes.
PASO 2 — Comprende el artículo: identifica el hecho principal, los actores
clave, la acción concreta y el contexto.
PASO 3 — Cruza texto e imagen: determina qué elementos visuales confirman,
complementan o contradicen el texto.
PASO 4 — Evalúa cada titular: para cada t1..t10, determina si es fiel al texto
y coherente con la imagen.
PASO 5 — Detecta distractores: identifica titulares con entidades incorrectas,
hechos no mencionados, lenguaje valorativo sin respaldo o demasiado genéricos.
PASO 6 — Construye el ranking: ordena de mayor a menor relevancia integrando
señales textuales y visuales.
PASO 7 — Emite la respuesta: escribe ÚNICAMENTE la secuencia final de tokens.

Criterios de evaluación (aplica en cada paso):
1. Fidelidad: el titular refleja con precisión lo que dice el artículo.
2. Especificidad: recoge los hechos, actores, acciones y contexto concretos.
3. Coherencia semántica: encaja con el tema central, no solo con palabras
   sueltas.
```

```
4. Penalización de errores: baja en el ranking titulares que:
   - Nombren a una persona, partido o institución diferente a la del artículo.
   - Atribuyan una acción que el artículo no menciona explícitamente.
   - Usen lenguaje valorativo ("escándalo", "fracaso", "polémica") sin
     respaldo textual directo.
   - Capturen solo el tema general pero omitan el hecho concreto relatado.
5. Coherencia visual: prioriza titulares coherentes con el contenido visual;
   penaliza los que lo contradicen o ignoran cuando la imagen es informativa.

Reglas de salida ESTRICTAS:
- Devuelve SOLO una secuencia de 10 tokens separados por espacios.
- Usa cada token t1, t2, ..., t10 exactamente una vez.
- El primer token es el más relevante, el último el menos relevante.
- No incluyas el razonamiento, explicaciones, signos de puntuación ni saltos
  de línea en la respuesta final.

Ejemplo de salida válida:
t3 t1 t9 t4 t2 t10 t8 t7 t6 t5
```

A.2.2. User Prompt Template

```
### ARTÍCULO ###
{article}

### IMAGEN ###
[La imagen asociada al artículo se adjunta a continuación]

### TITULARES CANDIDATOS ###
IMPORTANTE: El orden de aparición NO indica relevancia. Evalúa cada titular
de forma independiente.

t1: {title_1}
t2: {title_2}
t3: {title_3}
t4: {title_4}
t5: {title_5}
t6: {title_6}
t7: {title_7}
t8: {title_8}
t9: {title_9}
t10: {title_10}

Sigue el proceso de 7 pasos internamente:
1. ¿Qué muestra la imagen? ¿Personas, lugares, acciones, símbolos reconocibles?
2. ¿Cuál es el hecho principal del artículo? ¿Quiénes son los actores? ¿Qué
   acción concreta ocurre?
3. ¿Qué elementos visuales confirman o complementan el texto? ¿Hay
   contradicciones?
4. ¿Qué titulares reflejan ese hecho con precisión, especificidad y coherencia
   visual?
5. ¿Qué titulares contienen errores, exageraciones, omisiones o son demasiado
   genéricos?
```

```
6. Ordena los 10 titulares de mayor a menor relevancia integrando ambas señales.
7. Escribe SOLO la secuencia de tokens t1..t10 en una única línea, sin nada más.
```

A.3. Few-shot Example Format

When retrieval-based few-shot prompting was enabled, the selected examples were inserted before the target instance. Each example followed the same structure: article, candidate headlines, and gold ranking. The target case was then appended after the retrieved demonstrations.

```
A continuación tienes ejemplos resueltos del mismo formato. Aprende el patrón
de salida y responde solo con tokens t1..t10.

EJEMPLO 1
ARTICULO:
{example_article_1}

TITULARES:
t1: {example_1_title_1}
t2: {example_1_title_2}
...
t10: {example_1_title_10}

SALIDA_CORRECTA:
{example_1_gold_ranking}

EJEMPLO 2
ARTICULO:
{example_article_2}

TITULARES:
t1: {example_2_title_1}
t2: {example_2_title_2}
...
t10: {example_2_title_10}

SALIDA_CORRECTA:
{example_2_gold_ranking}

EJEMPLO 3
ARTICULO:
{example_article_3}

TITULARES:
t1: {example_3_title_1}
t2: {example_3_title_2}
...
t10: {example_3_title_10}

SALIDA_CORRECTA:
{example_3_gold_ranking}

CASO_OBJETIVO
```

```
{target_prompt}
```

A.4. Generative Inference Configuration

Table 7 summarizes the main inference and retrieval parameters used in the generative approach.

Parameter	Value
Decoding temperature	0.0
Top-p	1.0
Maximum output tokens	4096
Thinking mode	Disabled
Number of retrieved examples	3
Minimum similarity threshold	0.2

Table 7
Main inference and retrieval parameters used in the generative ranking pipeline.