Overview of the CLEF 2026 JOKER Task 1: Humor-Aware Information Retrieval in English and Hinglish

Poojan Vachharajani[1], Arjun Mukherjee[2], Jasvindar Singh[2], Krishna Tewari[2], Sukomal Pal[2], Jaap Kamps[3] and Liana Ermakova[4]

[1]Netaji Subhas University of Technology, New Delhi, India
[2]Indian Institute of Technology (BHU), Varanasi, India
[3]University of Amsterdam, Amsterdam, The Netherlands
[4]University of Brest, HCTI, Brest, France

Abstract

This paper presents the details of Task 1 of the JOKER-2026 Track, a humour-aware information retrieval task where the goal is to retrieve short humorous texts from a document collection for a given query. The intended use case is retrieving jokes on a specific topic, which may benefit humanities research, second-language learning, and the writing or translation of comedic texts. In this edition, we provide two settings: English and Hinglish. For both tracks, retrieved documents must satisfy two criteria: they must be relevant to the query and must be instances of wordplay, humour, or punning. The English setting includes 10 training queries with relevance judgments and 972 test queries, while the Hinglish setting includes 10 training queries with relevance judgments and 485 test queries. Runs are submitted in a standard TREC-style format and evaluated using standard information retrieval metrics, with MAP used for ranking. Overall, 18 teams submitted 767 runs for this task, including 642 runs for English and 125 runs for Hinglish.

Keywords

Humare-aware Information Retrieval, Computational Humour, Wordplay, Puns, Humour

1. Introduction

This paper provides an overview of Task 1 (*Humour-aware Information Retrieval*: Retrieve short humorous texts for a query in English and Hinglish) of the JOKER-2026 Track[1] [1], held as part of the Conference and Labs of the Evaluation Forum (CLEF 2026)[2]. For details on other JOKER-2026 tasks, we refer the reader to their respective overview papers [2, 3, 4]. The JOKER track series [5, 6, 7, 8, 1] aims to bring together researchers from information retrieval, natural language processing, computational linguistics, translation studies, and the humanities to advance the automatic analysis, retrieval, translation, and generation of humour and wordplay. Since its introduction as a humour-aware information retrieval task in 2024 [9], Task 1 has focused on retrieving short humorous texts from a document collection in response to user queries.

Standard information retrieval architectures usually optimize semantic relevance or topical matching. They are therefore not directly designed for non-literal, ambiguous, or meta-linguistic language. Humour-aware retrieval introduces an additional layer of difficulty because a retrieved document must not only match the query topic, but must also exhibit humour, wordplay, or punning. Thus, in Task 1, a document is considered relevant only if it satisfies a dual-relevance condition: it must be topically related to the input query and must be an explicit instance of structural wordplay, humour, or punning.

CLEF 2026 Working Notes, 21 – 24 September 2026, Jena, Germany
pjmathematician@gmail.com (P. Vachharajani); arjunmukherjee.rs.cse23@itbhu.ac.in (A. Mukherjee); jasvindarsingh.rs.min24@itbhu.ac.in (J. Singh); krishnatewari.rs.cse24@itbhu.ac.in (K. Tewari); spal.cse@itbhu.ac.in (S. Pal); kamps@uva.nl (J. Kamps); liana.ermakova@univ-brest.fr (L. Ermakova)
https://www.joker-project.com (L. Ermakova)
0009-0008-4214-5378 (P. Vachharajani); 0009-0007-4322-3537 (A. Mukherjee); 0009-0008-4461-2152 (J. Singh); 0009-0005-6599-9956 (K. Tewari); 0000-0001-8743-9830 (S. Pal); 0000-0002-6614-0087 (J. Kamps); 0000-0002-7598-7474 (L. Ermakova)

[1]https://www.joker-project.com
[2]https://clef2026.clef-initiative.eu/

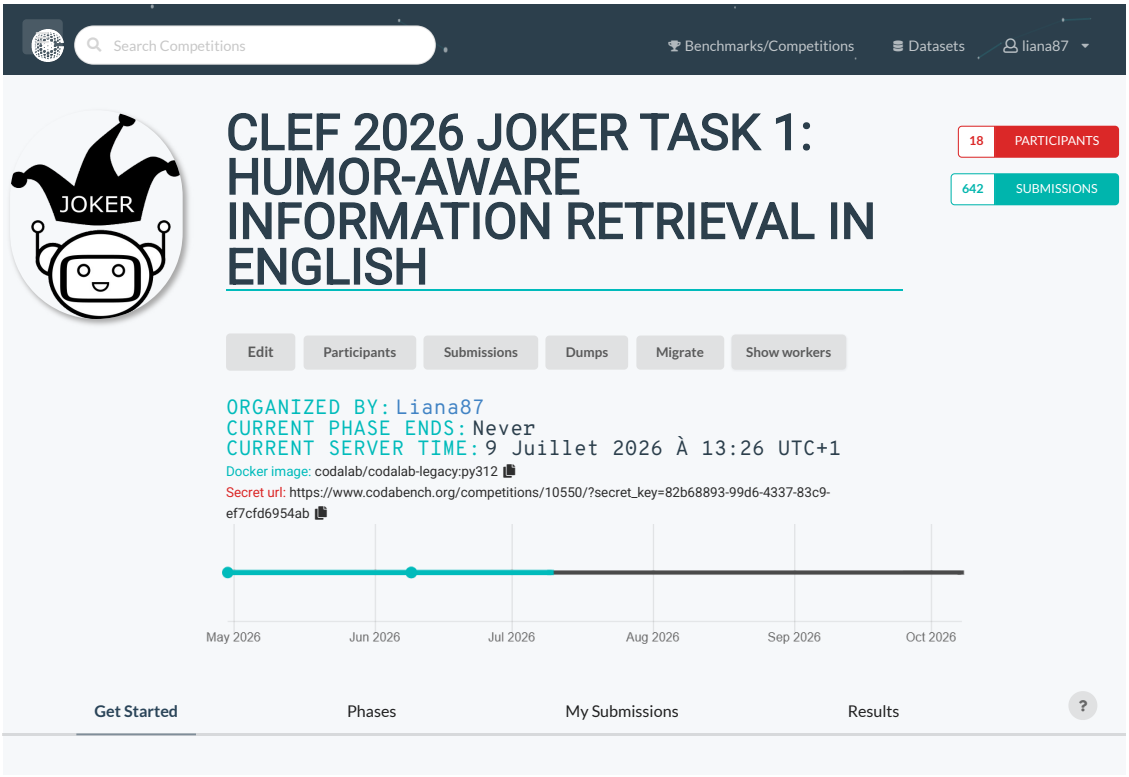

Figure 1: CodaBench for Task 1

For example, a query such as "math" requires retrieving jokes about mathematics, while a query such as "doctor" requires retrieving humorous texts in which the medical topic is central to the joke.

The practical motivation of this task lies at the intersection of information retrieval, computational humour, and digital humanities. Humour-aware retrieval can support humour scholars who need searchable collections of jokes and wordplay, second-language learners who use humorous examples to study idiomatic and culturally situated language, translators who need to adapt jokes across cultures, and professional writers or comedians who search for topic-specific humorous material [9, 10, 1]. The task therefore evaluates not only whether systems can retrieve documents about a topic, but also whether they can recognize humour-specific linguistic mechanisms that are often missed by general-purpose search systems.

For JOKER-2026, Task 1 is organized through separate CodaBench benchmarks for English[3] and Hinglish[4]. The inclusion of Hinglish broadens the task beyond the previously evaluated language settings and introduces a multilingual and code-mixed retrieval scenario, where lexical variation, script choices, and cultural humour references may affect retrieval performance.

In the official submission phase, Task 1 received 767 runs from 18 teams. The English subtask attracted 16 teams and 642 runs, while the Hinglish subtask attracted 11 teams and 125 runs, as shown in Table 1. These submission statistics indicate strong participation in the 2026 edition and a substantial expansion of the task compared with the 2025 edition [10].

[3] https://www.codabench.org/competitions/10550/
[4] https://www.codabench.org/competitions/10551/

Table 1
CLEF 2026 JOKER Task 1 official submission statistics

JOKER	Task 1	
	English	Hinglish
Teams	16	11
Runs	642	125
Teams	18	
Runs	767	

2. Dataset Details and Statistics

The evaluation framework for the 2026 edition scales historical iterations of the JOKER corpus by cleaning structural distribution shifts, eliminating duplicates, and expanding multi-dialect collections. The task was evaluated across two distinct linguistic tracks:

- **English Track:** Consists of a target corpus of 96,603 short documents (heterogeneous joke templates, one-liners, and narrative distractors) evaluated across 971 active test queries containing 4,572 hidden true relevance judgments.
- **Hinglish Track:** Introduces a specialized code-mixed evaluation set containing 32,652 documents and 485 test queries. This track introduces non-standard transliterations, informal Romanized spellings, and mixed-language expressions embedding English lexical items within Hindi grammatical frameworks.

For both language tracks, the training data was intentionally kept small (10 queries each) to test the system's generalizability and to lower-bound data efficiency, yielding 25 positive qrels for English and 60 for Hinglish.

2.1. Dataset Construction and Provenance

To scale and adapt the JOKER corpus, a systematic generation and scraping pipeline was established using advanced Large Language Models (LLMs) and search-grounding tools. Rather than collecting arbitrary web-scraped documents, the data preparation pipeline enforced explicit structural and semantic criteria across both target languages.

2.1.1. Positive Humor and Wordplay Generation

The extraction and synthesis of positive humorous texts relied on two complementary approaches: search-grounded LLM harvesting and targeted synthetic comedy generation.

1. **Search-Grounded Scraping (Gemini):** For specialized or colloquial words (e.g., Hinglish terms like *Chindi*), Google's `gemini-2.5-flash` model was queried with Google Search grounding active. This allowed the model to retrieve real-world jokes matching the target concepts while validating their references. For the Hinglish track, the scraping prompt enforced structural diversity by requiring a parallel triad for each query word: one joke in English, one joke in Hindi (transliterated into the Latin/English script), and one joke in Hinglish (code-mixed English/Hindi in the Latin script), accompanied by reference URLs and English semantic explanations.
2. **Controlled Humor Synthesis (Claude):** To expand the English and Hinglish templates with structured, high-quality puns, we employed Anthropic's `claude-sonnet-4-20250514-v1:0` via Amazon Bedrock. Under a structured system prompt, groups of thematic query terms were processed to produce two distinct, original jokes centered around each word. The model returned JSON blocks enclosing the target word, the generated jokes, and detailed explanations of the active linguistic mechanisms.

2.1.2. Negative Distractor Generation

A key challenge in evaluating dual-relevance retrieval is isolating systems that simply match query keywords from systems that truly understand humor. To create strong semantic distractors, we systematically generated factual (non-humorous) sentences containing the target keywords.

- **English Factual Distractors:** For each query keyword, we generated 20 factual sentences, statements, or conversational snippets using the `gemini-3.1-flash-lite-preview` model.
- **Hinglish Factual Distractors:** For code-mixed terms, we generated 60 sentences per word (20 English, 20 Romanized Hindi, 20 Hinglish) using `claude-sonnet-4-20250514-v1:0` to represent non-humorous contexts. These distractors cover a wide range of factual topics, encyclopedic descriptions, and neutral dialogues. They serve as hard negatives that systems must filter out to satisfy the dual-relevance condition.

2.2. Formats

This section outlines the format of the data used in the CLEF 2026 JOKER Task 1. The task contains four main files: the document corpus, the query file, the relevance judgments, and the submitted run file. Each file uses a simple field-based structure to make the task compatible with standard information retrieval evaluation tools.

2.2.1. Corpus Format

The corpus is provided as a JSON list of documents. Each document contains two fields: `id` and `contents`. The `id` field gives the document identifier. The `contents` field contains the full textual content of the document.

English Track examples

```
[
    {
        "docid": 0,
        "text": "Why did the musician get an A+ in alphabet class? Because
        ↪  they really knew their A!"
    },
    {
        "docid": 1,
        "text": "What did the article say when it won first place? 'I'm a
        ↪  winner!'"
    }
]
```

Hinglish Track examples

```
[
    {
        "docid": "21295",
        "text": "She handled the sharam situation very gracefully."
    },
    {
        "docid": "16670",
        "contents": "My neighbor is such a jaanwar when he's angry - it's
        ↪  like watching Animal Planet live!"
    }
]
```

2.2.2. Query Format

Queries are also provided in JSON format. Each query contains two fields: `qid` and `query`. The `qid` field gives the query identifier. The `query` field contains the query term or expression. Each query represents the information need for which relevant humorous documents must be retrieved.

English Track examples

```
[
    {"qid": "0", "query": "a"},
    {"qid": "1", "query": "ability"},
    {"qid": "2", "query": "abrupt"},
    {"qid": "3", "query": "accord"}
]
```

Hinglish Track examples

```
[
    {"qid": 115, "query": "Dal"},
    {"qid": 413, "query": "Service"},
    {"qid": 87, "query": "Chhat"},
    {"qid": 39, "query": "Bazaar"}
]
```

2.2.3. Qrels Format

The relevance judgments, or qrels, are as well provided in JSON format. Each judgment contains three fields: `qid`, `docid`, and `qrel`. The `qid` field identifies the query. The `docid` field identifies the relevant document. The `qrel` field gives the relevance value. A value of `1` indicates that the document is relevant to the query. For both English and Hinglish track the qrel format examples remain similar.

```
[
    {"qid": 0, "docid": 0, "qrel": 1},
    {"qid": 0, "docid": 1, "qrel": 1},
    {"qid": 87, "docid": 752, "qrel": 1}
]
```

Only the qrels for the training topics are released. These judgments can be used for offline training, validation, and system development. The qrels for the test topics are not released publicly and are stored in CodaBench for official evaluation.

2.2.4. Run Submission Format

System predictions were required to be submitted as a JSON file following a standard TREC-style run format. Each run entry contains the fields `run_id`, `manual`, `qid`, `docid`, `rank`, and `score`.

- `run_id` field identifies the submitted system.
- `manual` field should be set to `0` for automatic system predictions.
- `qid` field identifies the query.
- `docid` field identifies the retrieved document.
- `rank` field gives the position of the document in the ranked list.
- `score` field gives the retrieval score assigned by the system.

A sample run prediction file is as follows:

```
[
    {
        "run_id": "UAms_Task1_Anserini",
        "manual": 0,
        "qid": 1,
        "docid": "41958",
        "rank": 0,
        "score": 3.8852999210357666
    },
    {
        "run_id": "UAms_Task1_Anserini",
        "manual": 0,
        "qid": 1,
        "docid": "44782",
        "rank": 1,
        "score": 3.7513999938964844
    }
]
```


2.3. Training and Test Data

The task provides two query sets. The first is a smaller training set for which qrels are released. This set supports offline experimentation and model development. The second is a larger test set used for official evaluation on the CodaBench leaderboard. Participants submit their run files for the test queries to CodaBench, where they are evaluated against hidden qrels.

Participants may include results for the training topics in their submission files. They may also add extra fields to the submitted JSON file if needed. However, official evaluation is performed over all topics in the test query file. The submitted runs are evaluated using the standard `trec_eval` measures and implementation.

3. Evaluation

System outputs are evaluated against the hidden humor-aware relevance annotations using standard information retrieval measures across multiple ranking cutoffs. All relevant documents are both relevant to the topic and humorous.

The primary ranking metric is **Mean Average Precision (MAP)**, which penalizes models that fail to track both topical and humorous alignment concurrently. To evaluate top-rank precision density, which is critical for interactive conversational AI systems, we explicitly track normalized Discounted Cumulative Gain at early and deep ranks (NDCG@5, NDCG@10, NDCG@100), Mean Reciprocal Rank (MRR), and Precision at Rank 10 (P@10).

4. Participants' Approaches

A total of 18 teams actively engaged with the Task 1 framework, exhibiting a clear split between neural Large Language Model (LLM) distillation strategies and highly optimized, handcrafted feature ensembles.

arampageos [11] explored artifact-aware and hybrid structural patterns within the dataset. They constructed complex query and document feature representations spanning word- and character-level TF-IDF, compact string overlaps, and surface joke templates to feed to tree-based ensembles (Random Forests, Extra Trees, Gradient Boosting). Notably, after observing a strong corpus-order sequential regularity in the English benchmark, they applied a specialized block-ranking strategy that protected

the top ranks before sequentially outputting matching source segments. For Hinglish, they utilized a weighted Reciprocal Rank Fusion (RRF) over learning-to-rank configurations.

vanguard_team [12] presented the IROH pipeline, a sophisticated three-stage architecture driven by a LoRA-adapted ensemble of Large Language Models as the judge. They employed Gemma-4 (31B) to distill structured query-aware rationales and generated four types of humor-specific hard negatives (literal rewrites, defused punchlines, wrong-topic jokes, and near-miss puns). Candidates generated by a first-stage hybrid sparse-dense retriever (BM25 + BGE Embeddings) were filtered via a GTE-Reranker-ModernBERT cross-encoder before being evaluated by a soft weighted vote from Qwen2.5-7B and Gemma-4-31B judges.

astral_fate [13] implemented a two-stage retrieval pipeline using tree-based classifiers optimized with LightGBM and XGBoost. They engineered a 75-dimensional feature matrix focusing heavily on explicit joke-format heuristics and pun morphology, mapping formulaic template openers (*"Why did...", "What do you call..."*), punctuation structures, and character-level trigram Jaccard coefficients. Their work demonstrated that shallow morphology signals are remarkably descriptive in low-supervision settings.

arjun108 [14] built a sparse-focused architecture combining standard Okapi BM25 with query-expansion strategies. They developed a rule-based humor multiplier and a phonetic wordplay bonus that re-scored retrieved documents based on morphological pun patterns, comparing explicit corpus extensions against WordNet synonym lookups.

xinrui_qiu [15] engineered RIVER, a modular two-stage pipeline separating semantic topicality from wordplay filtering. They utilized a language-independent dense encoder (`paraphrase-multilingual-mpnet-base-v2`) to calculate initial query, document cosine similarities, followed by language-specific cross-encoder fine-tuning using a `roberta-base` binary classifier optimized with balanced negative downsampling.

ezrasmink [16] analyzed post-hoc filtering mechanisms using custom-trained neural pun detectors based on RoBERTa and `mDeBERTa-v3-base`. They evaluated combinations of sparse BM25 index metrics, BGE-M3 multi-vector combinations, and continuous pun probabilities blended with a WordNet lexical-ambiguity polysemy score to reward semantic depth.

5. Official Experimental Results

Table 2 and Table 3 provide the full multi-metric evaluations across the English and Hinglish tracks, sorted by official leaderboard rank.

The experimental runs yield critical insights into the constraints of computational humor retrieval. First, standard lexical indexing performs significantly worse in isolation because keyword matching surfaces massive pools of literal or encyclopedic prose distractors that lack wordplay.

The introduction of hand-crafted joke-format filters (*astral_fate*) or generative judge distillation architectures (*vanguard_team*) drives massive absolute gains, boosting MAP beyond 0.60. The split-frequency analysis highlights that while dense embeddings capture wide semantic concepts to support baseline recall, structural indicators, such as capitalization anomalies, punctuation markers, dialogue quotations, and document brevity, act as the primary knobs for separating true humorous context from informational text.

A major finding highlighted by *arampageos* is the manifestation of sequential block patterns in the hidden English distribution, where a block-first heuristic yielded an upper-bound MAP of 0.7582. However, cross-lingual analysis demonstrates that this structural artifact does not transfer across domains: the Hinglish track proved highly resilient to block-offset adjustments, thereby necessitating reliance on transferable feature-driven machine learning models. On Hinglish, the tree-based ensemble combining Extra Trees and Gradient Boosting achieved the leading transferable performance, with a MAP of 0.6648, indicating that modeling non-linear surface variations remains paramount in code-mixed spaces.

Table 2
Official evaluation results for Task 1 (English Track): top five runs per team wherever available. Baseline metrics are bolded.

#	Owner	MAP	NDCG			MRR	Prec	Run ID
			5	10	100		10	
1	arampageos	0.76	0.85	0.82	0.85	0.99	0.30	submission_english_blockn LeakHybrid_06072_blockfirst_seq
2	arampageos	0.76	0.85	0.82	0.85	0.99	0.30	submission_english_block LeakHybrid_06072_blockfirst_seqextra
3	arampageos	0.66	0.77	0.74	0.78	0.86	0.30	submission_english_block LeakHybrid_06072_p2_seq
4	arampageos	0.63	0.70	0.72	0.76	0.85	0.30	submission_english_block LeakHybrid_06072_p5_seq
5	arampageos	0.62	0.70	0.69	0.75	0.85	0.28	submission_english_block LeakHybrid_06072_p10_seq
6	vanguard_team	0.63	0.69	0.71	0.77	0.83	0.31	ce30j70_t30_p85_qw60_go30_gn10
7	astral_fate	0.61	0.70	0.69	0.75	0.85	0.28	lgb_cls
8	astral_fate	0.61	0.69	0.69	0.74	0.84	0.28	_BASELINE_2026only
9	astral_fate	0.61	0.69	0.68	0.74	0.84	0.27	ENSEMBLE_best5
10	astral_fate	0.61	0.69	0.68	0.74	0.84	0.27	ENSEMBLE_no_xgb
11	astral_fate	0.60	0.69	0.68	0.74	0.84	0.27	ENSEMBLE_all7
12	arjun108	0.39	0.47	0.50	0.54	0.68	0.23	S1_1
13	arjun108	0.35	0.42	0.45	0.49	0.59	0.21	Submission_1_2
14	madhangi	0.32	0.38	0.40	0.50	0.54	0.18	BM25
15	madhangi	0.10	0.15	0.14	0.18	0.27	0.06	Baseline
16	madhangi	0.10	0.15	0.14	0.18	0.27	0.06	Baseline2
17	madhangi	0.00	0.00	0.00	0.00	0.00	0.00	Humour_Rerank
18	xinrui_qiu	0.31	0.41	0.41	0.44	0.60	0.17	MPNetRoBERTa
19	ezrasmink	0.30	0.39	0.40	0.43	0.55	0.18	no reranker + WordNet 0.3
20	ezrasmink	0.30	0.38	0.40	0.44	0.56	0.17	hybrid, no reranker, no WordNet
21	ezrasmink	0.28	0.34	0.36	0.45	0.52	0.16	Pool 500: Hybrid RRF + humour-polysemy, pool500
22	ezrasmink	0.22	0.26	0.32	0.37	0.38	0.17	pool5000
23	prasann	0.27	0.26	0.34	0.46	0.39	0.20	comtrastive learn with instruct
-	baseline	0.11	0.10	0.11	0.27	0.21	0.05	BM25 Anserini Index
-	baseline	0.10	0.09	0.10	0.27	0.19	0.05	BM25 + RM3 Anserini Index

6. Analysis and Evaluation of Dual Relevance

Humor-aware information retrieval requires systems to evaluate candidates on both topical relevance and the presence of humor or wordplay. This section analyzes the performance of participants' systems under this dual-relevance framework, evaluating how models balanced these distinct criteria.

6.1. Topical Relevance vs. Humor Detection

Standard information retrieval architectures, such as dense encoders and cross-encoder rerankers, are trained to prioritize semantic and topical matching. In a humor-aware retrieval setting, this focus can be counterproductive. As the University of Amsterdam (*ezrasmink*) reported [16], applying a modern cross-encoder (`bge-reranker-v2-m3`) to rescore candidates consistently lowered the Mean Average Precision (MAP). The cross-encoder reordered documents based on plain topical similarity, pushing factual, literal distractors to the top and displacing jokes.

To address this tension, several participants separated semantic matching from humor filtering into distinct modules:

Table 3
Official evaluation results for Task 1 (Hinglish Track): top five runs per team wherever available. Baseline metrics are bolded.

#	Owner	MAP	NDCG			MRR	Prec	Run ID
			5	10	100		10	
1	arampageos	0.66	0.72	0.75	0.80	0.88	0.45	submission_hinglish_tree_ens_et_gb
2	arampageos	0.66	0.72	0.75	0.80	0.88	0.45	submission_hinglish_rankbias 6648_mixA_p5
3	arampageos	0.66	0.72	0.75	0.80	0.88	0.45	submission_hinglish_exact Rescue_06648_p5_r1_check5
4	arampageos	0.66	0.72	0.75	0.80	0.88	0.45	submission_hinglish_exact Rescue_06648_p10_r2_check5
5	arampageos	0.66	0.72	0.75	0.80	0.88	0.45	submission_hinglish_best 6648_boost_safeA_k100
6	xinrui_qiu	0.39	0.54	0.52	0.54	0.84	0.27	RIVER_task_1_MPNetRoBERTa
7	arjun108	0.34	0.45	0.46	0.51	0.76	0.26	S1_1
8	arjun108	0.31	0.42	0.43	0.48	0.71	0.24	S1_2
9	madhangi	0.22	0.29	0.29	0.45	0.55	0.16	BM25
10	madhangi	0.08	0.10	0.10	0.26	0.28	0.05	baseline
-	baseline	0.09	0.18	0.16	0.16	0.53	0.05	Baseline

- **High-Recall First Stage:** *xinrui_qiu* (*RIVER*) used the multilingual `paraphrase-multilingual-mpnet-base-v2` to produce initial candidate sets, applying a permissive similarity threshold of 0.1 to avoid losing relevant jokes early in the pipeline [15].
- **Specialized Humor Filtering:** *xinrui_qiu* then used a fine-tuned `roberta-base` cross-encoder as a dedicated binary classifier to identify wordplay. By combining the retrieval and humor scores with a fixed 0.3/0.7 weighted fusion, they restricted the final ranking to documents identified as humorous.

6.2. The Role of Literal Query Matching

The structure of JOKER-2026 Task 1 English introduced specific design constraints. An analysis of the test set by *astral_fate* revealed that the target query word appeared literally in 94% of the manually annotated relevant documents [13]. This property made lexical keyword match a reliable proxy for topical relevance.

With topicality largely addressed by literal matching, the main retrieval challenge shifted to distinguishing jokes from encyclopedic, definitional, or prose passages that also contained the query term. *astral_fate* addressed this by introducing hand-crafted joke-format and pun-morphology features:

- **Template Openers:** Checking for formulaic structures like *"Why did...", "What do you call..."*.
- **Punctuation and Length Signals:** Counting question marks, exclamation points, and monitoring document length.

Adding these joke-format features produced the single largest performance increase in their ablation studies, raising the test MAP by 0.33 (from 0.2635 to 0.5940), as shown in Table 2. This suggests that when topicality is addressed by exact query matching, shallow structural heuristics can be highly effective at isolating humorous forms.

6.3. Cross-Year Data Augmentation and Negative Transfer

A common response to the small number of training queries (10 per track) was to augment the training data with qrels from previous JOKER editions. However, this strategy frequently resulted in negative transfer.

As documented by *astral_fate*, augmenting the JOKER-2026 English training set with 660 positive labels from JOKER-2025 degraded test performance by approximately 0.08 MAP [13]. This degradation stems from a difference in query distributions:

- **JOKER-2025 English Task 1:** Features conceptual queries (e.g., "colors") where relevant documents use conceptually related terms ("blue", "red", "hue") rather than the query word itself.
- **JOKER-2026 Task 1:** Relies on literal, word-level queries where exact-word matching is highly dominant.

Training on the combined dataset led models to assign high weights to dense semantic embeddings (to capture conceptual matches) at the expense of exact lexical match signals. This learned weight distribution was appropriate for the 2025 data but degraded performance on the 2026 test set, showing that naive data pooling can be counterproductive when underlying retrieval paradigms change.

6.4. LLM Rationales and Hard Negatives

The team *vanguard_team* (*VANGUARD*) addressed the dual-relevance task using a three-stage pipeline (*IROH*) driven by rationale distillation and structured data augmentation [12]. To help their models identify humorous structures, they used `Gemma-4` (31B) to generate a one-sentence explanation (rationale) of the specific linguistic mechanism (e.g., homophony, double entendre, or malapropism) for each training pair.

To improve the model's discriminative ability, they generated four types of query-aware hard negatives:

1. **Literal Rewrites:** Factual statements matching the query topic with all humor removed.
2. **Defused Jokes:** Retaining the setup but replacing the pun word with a literal alternative.
3. **Wrong-Topic Jokes:** Genuinely humorous texts unrelated to the query.
4. **Near-Miss Puns:** Forced or incomplete wordplay attempts related to the query.

Their results showed that a distilled, ensemble judge consisting of `Qwen2.5-7B` and `Gemma-4-31B` achieved 0.6347 MAP, which was the highest score among non-artifact-aware systems. However, they observed that training with these structured hard negatives actually degraded performance on the official leaderboard. While the hard negatives improved validation scores on their local splits, they caused the models to overfit to synthetic humor boundaries, reducing generalizability on the official test set.

6.5. Benchmark Robustness and Artifacts

A key finding in the English track was a dataset artifact identified by *arampageos* (*DUTH*) [11]. They observed that a significant portion of relevant documents in the English test set occurred in sequential blocks within the corpus. By applying a block-ranking strategy that preserved the top ranks and outputted matching source segments sequentially, they achieved a MAP of 0.7582, outperforming all other runs.

However, cross-lingual analysis showed that this block structure was specific to the English dataset. When applied to the Hinglish track, the same block-ranking configurations produced low performance. On Hinglish, their best result (0.6648 MAP) was obtained using a supervised ensemble of tree-based learning-to-rank models (Extra Trees, Gradient Boosting, and Histogram-based Gradient Boosting). This indicates that the Hinglish dataset was resilient to the sequential artifact, requiring models to rely on features like character-level TF-IDF and morphological signals to handle transliterated and code-mixed humor. This distinction highlights the importance of multi-dataset and multi-lingual validation to ensure that ranking improvements reflect generalizable humor understanding rather than dataset-specific orderings.

7. Discussion and Conclusions

In this paper, we presented an overview and discussion of Task 1 of the JOKER-2026 Track on humour-aware information retrieval in English and Hinglish. The task required systems to retrieve short, humorous texts in response to a given query. A document was considered relevant only when it satisfied two conditions: it had to be topically related to the query and it had to be an instance of humour, wordplay, or punning. This dual-relevance requirement makes the task more challenging than standard information retrieval, where systems usually optimize only topical or semantic relevance.

The 2026 edition expanded the task setting in two important ways. First, the English track was scaled to a larger evaluation collection, consisting of 96,603 short documents, 971 active test queries, and 4,572 hidden relevance judgments. Second, Hinglish was introduced as a new code-mixed setting, with 32,652 documents and 485 test queries. This setting introduced additional retrieval challenges, including Romanized Hindi, informal spellings, mixed English-Hindi expressions, and culturally grounded humour. For both language tracks, the training data was intentionally small, with only 10 training queries per language. This design encouraged systems to rely on generalizable retrieval strategies rather than task-specific supervised training alone.

The official submission phase showed strong participation. Overall, 18 teams submitted 767 runs for Task 1. The English track received 642 runs from 16 teams, while the Hinglish track received 125 runs from 11 teams. The submitted systems covered a wide range of methods, including BM25-based retrieval, query expansion, dense multilingual retrieval, cross-encoder reranking, handcrafted humour features, tree-based ensembles, neural pun detectors, and LLM-based humour judges.

The results show that standard lexical retrieval remains a weak baseline for humour-aware search. The BM25 baseline achieved a MAP of 0.11 for English and 0.09 for Hinglish. These scores indicate that keyword matching can retrieve topically related documents, but it is not sufficient for identifying humorous or wordplay-based relevance. In contrast, the best submitted systems achieved much stronger results, with the top English run reaching a MAP of 0.76 and the top Hinglish run reaching a MAP of 0.66.

For English, the best result was achieved by *arampageos*, with a MAP of 0.76, NDCG@5 of 0.85, NDCG@10 of 0.82, NDCG@100 of 0.85, MRR of 0.99, and P@10 of 0.30. This run substantially outperformed the BM25 baseline and all other submitted systems. However, the run also relied on an observed sequential block pattern in the English data. This suggests that part of the gain may be associated with structural regularities in the benchmark rather than purely transferable humour-understanding ability. Therefore, this result should be interpreted carefully.

The next strongest English systems were submitted by *vanguard_team* and *astral_fate*, with MAP scores of 0.63 and 0.61, respectively. These systems are especially informative because they used more generalizable strategies. *vanguard_team* combined sparse and dense retrieval, cross-encoder reranking, and LLM-based humour judging. *astral_fate* used handcrafted pun morphology, joke-format features, and tree-based learning. Their strong performance suggests that humour-aware retrieval benefits from combining topical retrieval with explicit humour-sensitive filtering.

The Hinglish results show a different pattern. The best Hinglish run was again submitted by *arampageos*, achieving a MAP of 0.66, NDCG@5 of 0.72, NDCG@10 of 0.75, NDCG@100 of 0.80, MRR of 0.88, and P@10 of 0.50. Unlike the English setting, the Hinglish track was less affected by sequential block-based strategies. The best performance instead came from tree-based ensemble modelling using transferable ranking and surface-level features. This suggests that feature robustness is particularly important in code-mixed humour retrieval, where spelling variation, transliteration, and informal language are common.

Across both tracks, the strongest systems shared a common pattern. They did not treat the task as a single-step retrieval problem. Instead, they combined initial retrieval with humour-oriented reranking or filtering. These filters included joke templates, pun morphology, phonetic wordplay indicators, dense semantic similarity, cross-encoder scores, neural pun detection, and LLM-based relevance judgments. This confirms the core motivation of the task: humour-aware retrieval requires systems to model both what the text is about and how the humorous effect is produced.

The comparison between English and Hinglish also highlights the importance of multilingual and code-mixed evaluation. English systems benefited from lexical, structural, and humour-specific cues. Hinglish systems had to handle informal Romanization, language mixing, and culturally situated humour. Dense multilingual encoders helped capture semantic similarity across languages, but the results show that semantic matching alone is still insufficient. Wordplay, punchline structure, and non-literal meaning remain difficult to identify reliably.

Overall, the 2026 results show clear progress over simple retrieval baselines. At the same time, they confirm that humour-aware retrieval remains an open challenge. General-purpose retrieval models are still largely humour-agnostic. Stronger systems need to combine retrieval, reranking, and explicit humour modelling. The English block-pattern observation also shows that future editions should further control for dataset artifacts and hidden structural regularities. This would help ensure that leaderboard gains reflect genuine humour-aware retrieval ability rather than corpus-specific shortcuts.

Future work should focus on separating topical relevance from humour relevance during evaluation. Additional analysis by joke type, pun mechanism, language variety, and query difficulty would also be useful. For Hinglish and other code-mixed settings, future datasets should further examine transliteration variation, cultural references, and mixed-language humour. These directions can help the JOKER task continue to support reusable benchmarks for computational humour, multilingual information retrieval, and digital humanities research.

For more information about the JOKER lab this year, please refer to the overview paper [1], and the Working Notes papers for Task 2: Pun Translation [2], Task 3: Onomastic Wordplay Translation [3], and Task 4: Humour Generation [4]. Visit the JOKER website at https://joker-project.com for any other information related to the track.

Acknowledgments

We thank the track and task organizers for their amazing service and effort in making realistic benchmarks available for analyzing and processing humorous text.

This project is funded by the National Research Agency (ANR-19-GURE-0001) under the program "*Investissements d'avenir*" integrated into France 2030.

Jaap Kamps is partly funded by the Netherlands Organization for Scientific Research (NWO NWA #1518.22.105), the University of Amsterdam (AI4FinTech program), and ICAI (AI for Open Government Lab). Views expressed in this paper are not necessarily shared or endorsed by those funding the research.

Declaration on Generative AI

During the preparation of this work, the authors used *ChatGPT* and *Grammarly* in order to: **Grammar and spelling check** and **Paraphrase and reword**. After using these tools/services, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] L. Ermakova, et al., Overview of the CLEF 2026 JOKER track: Humor detection, search, and translation, in: M. Hagen, M. Potthast, B. Stein, P. Schaer, E. Zangerle, S. MacAvaney, J. M. Struß, E. Sánchez Salido, A. Barrón Cedeño, A. García Seco de Herrera (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Seventeenth International Conference of the CLEF Association (CLEF 2026), Lecture Notes in Computer Science, Springer, 2026.

[2] L. Ermakova, Y. Naud, L. Binet, B. Dumas, J. Kamps, Overview of the CLEF 2026 JOKER Task 2: Pun Translation from English to French, in: [17], 2026.

[3] L. Ermakova, Y. Naud, L. Binet, J. Kamps, Overview of the CLEF 2026 JOKER Task 3: Onomastic Wordplay Translation from English to French, in: [17], 2026.
[4] I. Kuzmin, A.-G. Bosser, J. Kamps, L. Ermakova, Overview of the CLEF 2026 JOKER Task 4: Humor Generation, in: [17], 2026.
[5] L. Ermakova, T. Miller, F. Regattin, A.-G. Bosser, C. Borg, Élise Mathurin, G. L. Corre, S. Araújo, R. Hannachi, J. Boccou, A. Digue, A. Damoy, B. Jeanjean, Overview of JOKER@CLEF 2022: Automatic wordplay and humour translation workshop, in: A. Barrón-Cedeño, G. D. S. Martino, M. D. Esposti, F. Sebastiani, C. Macdonald, G. Pasi, A. Hanbury, M. Potthast, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction: Proceedings of the Thirteenth International Conference of the CLEF Association (CLEF 2022), volume 13390 of *Lecture Notes in Computer Science*, Springer, Cham, 2022, pp. 447–469. doi:10.1007/978-3-031-13643-6_27.
[6] L. Ermakova, T. Miller, A.-G. Bosser, V. M. Palma Preciado, G. Sidorov, A. Jatowt, Overview of JOKER – CLEF-2023 track on automatic wordplay analysis, in: A. Arampatzis, E. Kanoulas, T. Tsikrika, S. Vrochidis, A. Giachanou, D. Li, M. Aliannejadi, M. Vlachos, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction: Proceedings of the Fourteenth International Conference of the CLEF Association (CLEF 2023), volume 14163 of *Lecture Notes in Computer Science*, Springer, Cham, 2023, pp. 397–415. doi:10.1007/978-3-031-42448-9_26.
[7] L. Ermakova, A.-G. Bosser, T. Miller, V. M. Palma Preciado, G. Sidorov, A. Jatowt, Overview of the CLEF 2024 JOKER track: Automatic humour analysis, in: L. Goeuriot, P. Mulhem, G. Quénot, D. Schwab, L. Soulier, G. M. D. Nunzio, P. Galuščáková, A. G. S. de Herrera, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction: Proceedings of the Fifteenth International Conference of the CLEF Association (CLEF 2024), volume 14959 of *Lecture Notes in Computer Science*, Springer, Cham, 2024, pp. 165–182. doi:10.1007/978-3-031-71908-0_8.
[8] L. Ermakova, A.-G. Bosser, T. Miller, R. Campos, Overview of JOKER: Humour in the machine, in: J. C. de Albornoz, J. Gonzalo, L. Plaza, A. G. S. de Herrera, J. Mothe, F. Piroi, P. Rosso, D. Spina, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Sixteenth International Conference of the CLEF Association (CLEF 2025), Lecture Notes in Computer Science, Springer, 2025.
[9] L. Ermakova, A.-G. Bosser, T. Miller, A. Jatowt, Overview of the CLEF 2024 JOKER task 1: Humour-aware information retrieval, in: G. Faggioli, N. Ferro, P. Galuščáková, A. G. Seco de Herrera (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), volume 3740 of *CEUR Workshop Proceedings*, 2024, pp. 1775–1785.
[10] L. Ermakova, R. Campos, A.-G. Bosser, T. Miller, Overview of the CLEF 2025 JOKER task 1: Humour-aware information retrieval, in: Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2025), volume 4038 of *CEUR Workshop Proceedings*, CEUR-WS.org, Madrid, Spain, 2025, pp. 2744–2760. URL: https://ceur-ws.org/Vol-4038/paper_218.pdf.
[11] G. Arampatzis, A. Arampatzis, DUTH at CLEF JOKER 2026: Conservative Hybridisation for Humor-Aware Retrieval, Pun Translation, Onomastic Wordplay Translation, and Multilingual Humour Generation, in: [17], 2026.
[12] A.-M. L. Mocanu, S. Mocanu, C.-O. Truică, E.-S. Apostol, IROH: Insightful Ranking Of Humor using Multi-Stage Hybrid Retrieval with Rationale-Distilled LLM Judges for JOKER 2026 Track Task 1 English, in: [17], 2026.
[13] F. E. Eldin, Cairo University at the CLEF 2026 JOKER Track Heuristic Pun-Morphology Features and Tree Ensembles for Humor-Aware Information Retrieval, in: [17], 2026.
[14] A. Mukherjee, K. Tewari, J. Singh, S. Pal, From Words to Wit: Humor-Aware Information Retrieval in English and Hinglish, in: [17], 2026.
[15] X. Qiu, H. Chen, J. Zhai, CLEF 2026 JOKER Track: Relevance-Aware Multilingual Retrieval for Humorous Wordplay, in: [17], 2026.
[16] J. van Bakel, R. Flier, E. Smink, J. Bakker, J. Kamps, University of Amsterdam at the CLEF 2026 JOKER Track, in: [17], 2026.

[17] E. Sánchez Salido, A. Barrón Cedeño, A. García Seco de Herrera, S. MacAvaney, J. M. Struß (Eds.), Working Notes of CLEF 2026: Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2026.