University of Amsterdam at the CLEF 2026 JOKER Track

Jesse van Bakel, Robin Flier, Ezra Smink, Jan Bakker and Jaap Kamps

University of Amsterdam, Amsterdam, The Netherlands

Abstract

This paper reports on the University of Amsterdam's participation in the CLEF 2026 JOKER track. Our overall goal is to investigate the non-literal use of language, such as in humor and wordplay, that remains challenging for current information retrieval and natural language processing technologies. Our specific focus is on a post hoc approach that uses wordplay or humor detection to filter out humorous translations, outputs, or search results. Our main findings are the following. First, we successfully developed effective humor detection classifiers for both English and French. Second, for humor-aware information retrieval, we could increase retrieval effectiveness by filtering for humorous content in search results ranked solely on topical relevance. Our strongest retrieval results were obtained by combining sparse and dense retrieval via reciprocal-rank fusion and reweighting the ranking with a neural pun detector. On the other hand, the cross-encoder re-ranking lowered effectiveness by prioritizing topical relevance over the humor signal. Third, for wordplay translation, we generate multiple translation candidates and select the one with the highest pun score as determined by the detector. This approach performs well, with limited gain on the automatic evaluation measures, but qualitative analysis confirms that this is an encouraging strategy. Fourth, for humor generation, we produce multiple outputs for a given query and select the one based on ambiguity and distinctiveness scores. We find that the largest general model performs best for English, whereas language-specific models perform best for Spanish and French. Overall, most gains are not realized by using a larger model, but instead by how the humor signals are combined.

Keywords

Information Storage and Retrieval, Natural Language Processing, Wordplay Translation, Humor Retrieval, Funny Names Translation

1. Introduction

The CLEF 2026 JOKER track investigates possible solutions to the challenges of automated analysis and processing of humor. The JOKER track series aims to advance the development of methods for interpreting, generating, and translating wordplay by bringing together computer scientists, linguists, and translators. The CLEF 2026 JOKER Track builds upon the findings from CLEF 2025 JOKER Track [1], in particular on humor-aware search [2], and wordplay translation [3]. We conduct an extensive analysis of the track: Task 1 on *Humor-aware Information Retrieval*; Task 2 on *Wordplay Translation*; Task 3 on *Onomastic Wordplay Translation*; and Task 4 on *Humor Generation*. For details on the exact track setup in 2026, we refer to the Track Overview paper CLEF 2026 LNCS proceedings [4], as well as the detailed task overviews in the CEUR proceedings [5, 6, 7, 8].

For each task, we investigated several models for retrieval, translation, or text generation. This was done by leveraging existing topic relevance ranking systems and translation models. This was combined with several post hoc approaches to detect humorous text. For the retrieval task, this means that humorous texts should get a higher ranking than non-humorous texts. For the translation and generation tasks, the most humorous text should be selected as the final output. Post-processing search results and text outputs was done using different rule-based scores and neural classifiers as filters for humorous text.

The rest of this paper is structured as follows. Next, in Section 2 we discuss our experimental setup and the specific runs submitted. Section 3 discusses the results of our runs. We end in Section 4 by discussing our results and outlining the lessons learned.

CLEF 2026 Working Notes, 21–24 September 2026, Jena, Germany

j.bakker@uva.nl (J. Bakker); kamps@uva.nl (J. Kamps)

0009-0002-9085-8491 (J. Bakker); 0000-0002-6614-0087 (J. Kamps)

© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

2. Experimental Setup

In this section, we will detail our approach for the three CLEF 2026 JOKER track tasks.

For details of the exact task setup and results, we refer the reader to the detailed overviews of the track in [4] and the overviews of each task in [5, 6, 7, 8].

The basic ingredients of the track are:

Corpus For Task 1, there is a large corpus of 96,603 documents (usually a single sentence each) for the retrieval task in English. The corpus for the retrieval task in Hinglish contains 32,652 documents.

Train Data For Task 1, there are 10 English train queries with relevance judgments (ranging from 2 to 7 relevant per query). There are also 10 Hinglish train queries with relevance judgments (6 relevant per query).

For Task 2, there are 1,405 English wordplays, totaling 5,838 professional human translations into French.

For Task 3, there are 353 English onomastic wordplays, or informally "funny" names, with professional human translations in French.

For Task 4, there are 3,344 training examples in English, 392 in Spanish, and 1,599 in French.

Test Data For Task 1, there are 972 English test queries. These do not contain the 10 train queries. There are also 485 Hinglish test queries, which do not include any of the train queries.

For Task 2, there are 4,061 English wordplays to be translated into French.

For Task 3, there are 2,333 English onomastic wordplays, with French reference translations made by professional translators.

For Task 4, there are 150 test instances in English, 151 in Spanish, and 156 in French.

We created runs for all four tasks of the 2026 track, which we will discuss in order.

2.1. Task 1: Humor-aware Information Retrieval

This task asks to retrieve short humorous texts for a query. We submitted 17 runs in total, shown in Table 1.

Baseline Rankers We first submitted two baseline runs focusing on regular information retrieval effectiveness. These are vanilla baseline runs on an Anserini index, using either BM25 or BM25+RM3 with default settings [9].[1] We submitted two runs for both the English and the Hinglish data. To understand the effectiveness of standard retrieval systems optimized for topical relevance, our submissions used default, non-optimized settings.

BGE-M3 BGE-M3 is a text embedding model developed by the Beijing Academy of Artificial Intelligence (BAAI), capable of doing sparse, dense, and multi-vector retrieval. It supports over 100 languages and can handle both short and long texts.[2] With dense embeddings, the retrieval system is able to capture a larger part of a document's semantic meaning. The first three runs with BGE-M3 combined the BM25 score with BGE-M3's dense retrieval. This was done using a 40/60 ratio.

In addition to this linear combination, we also fused the BM25 and BGE-M3 rankings with Reciprocal Rank Fusion (RRF, $k = 60$) [10]. This combines results by their rank position rather than their raw scores and thereby avoids having to normalize the very different score scales of a lexical and a dense retriever.

[1] https://github.com/castorini/pyserini
[2] https://huggingface.co/BAAI/bge-m3

Table 1
CLEF 2026 JOKER Track Submissions

	Run	Description
1	UAms_Task1en_bm25	BM25 baseline (Anserini, stemming)
1	UAms_Task1en_bm25_filter40	BM25 + Filter on RoBERTa pun classifier (drops scores <40%)
1	UAms_Task1en_bm25_f1	BM25 + F1-score with RoBERTa Pun classifier score
1	UAms_Task1en_rm3	RM3 baseline (Anserini, stemming)
1	UAms_Task1en_rm3_filter40	RM3 + Filter on RoBERTa pun classifier (drops scores <40%)
1	UAms_Task1en_rm3_f1	RM3 + F1-score with RoBERTa Pun classifier score
1	UAms_Task1en_BGE_M3_Hybrid_00	BM25 + BGE-M3 (40/60) baseline
1	UAms_Task1en_BGE_M3_Hybrid_40	BM25 + BGE-M3 (40/60) + Filter on RoBERTa classifier
1	UAms_Task1en_BGE_M3_Hybrid_f1	BM25 + BGE-M3 (40/60) + F1-score with RoBERTa Pun classifier score
1	UAms_task1en_hybrid_norerank	BM25 + BGE-M3 (RRF), mDeBERTa-v3 pun detector
1	UAms_task1en_5000pool	BM25 + BGE-M3 (RRF) + BGE cross-encoder re-ranking
1	UAms_task1en_hybrid_norerank_wordnet0.3	BM25 + BGE-M3 (RRF) + mDeBERTa-v3 with WordNet polysemy
1	UAms_task1en_twostage_hw0.6_pool500	BM25 + BGE-M3 (RRF) for recall, mDeBERTa-v3 with WordNet ranking
1	UAms_Task1hi_bm25	BM25 baseline (Anserini, stemming)
1	UAms_Task1hi_rm3	RM3 baseline (Anserini, stemming)
1	UAms_Task1hi_BGE_M3_Hybrid_00	BGE-M3 baseline
1	UAms_Task1hi_BGE_M3_Hybrid_f1	BGE-M3 + F1-score with RoBERTa Pun classifier score
2	UAms_task2_MarianMT_XLMRoBERTa	MarianMT + XLM-RoBERTa multilingual classifier
2	UAms_task2_MarianMT_CamemBERT	MarianMT + CamemBERT French-specific classifier
2	UAms_task2_marian_c0.0	MarianMT + mDeBERTa-v3 pun detector
2	UAms_task2_marian_c.25	MarianMT + content preservation
2	UAms_task2_nllb_c0.25	NLLB-1.3B + mDeBERTa-v3 pun detector and content preservation
2	UAms_task2_mbart_c.25	mBART + mDeBERTa-v3 pun detector and content preservation
3	Uams_task3_literal	MarianMT bare name (literal)
3	UAms_task3_context	MarianMT + description context
3	UAms_task3_copy_extract	Rule-based name extraction
3	UAms_task3_extended_context	Extraction + literal hybrid
3	UAms_task3_two_option	Extraction + context hybrid
4	UAms_Task4_T5ft	Fine-tuned T5
4	UAms_Task4_T5ptft	Pretrained and fine-tuned T5
4	UAms_Task4_Gemma2_A	Gemma 2 version A
4	UAms_Task4_Gemma2_B	Gemma 2 version B

RoBERTa We trained a RoBERTa model to distinguish puns from non-puns. This was then combined with the baseline and BGE-M3 rankings. For all runs ending in '`40`', all documents with a score above 40% were kept. For all runs ending in '`f1`', the F1-score of the normalized relevance score and normalized pun classification score was calculated and used as the final ranking.

Alongside RoBERTa, we trained a multilingual pun detector based on mDeBERTa-v3-base, so that the same humor signal could be applied to both the English and the Hinglish data.[3] Instead of only filtering on it, we also used the detector's pun probability as a continuous weight that is blended with the relevance score.

[3]https://huggingface.co/microsoft/mdeberta-v3-base

Cross-encoder re-ranking We further tested whether a modern cross-encoder (`bge-reranker-v2-m3`) could improve the ranking by re-scoring the top 5000 retrieved documents.[4] This consistently lowered the MAP, since it reorders results toward topical relevance and overrides the humor signal.

WordNet As an additional humor cue, we computed a lexical-ambiguity score from WordNet, based on the number of senses of a document's content words, since puns often rely on word ambiguity.[5]

Two-stage humor ranking Finally, we experimented with a two-stage setup in which an initial candidate pool was retrieved to maximize recall, after which the candidates were re-ranked primarily according to the humor signal.

2.2. Task 2: Wordplay Translation

This task asks to translate english punning jokes into french. We submitted six runs, as shown in Table 1.

Translation Model We fine-tune MarianMT (`Helsinki-NLP/opus-mt-en-fr`, 136M parameters) on the JOKER translation corpus of 1,405 English puns paired with 5,838 French reference translations. The data is split into 72% training, 18% validation, and 10% test. Training uses a learning rate of 1e-5, batch size 8, 5 epochs with early stopping on validation BLEU, and mixed precision (FP16). The fine-tuned model achieves a BLEU score of 24.79 on our held-out test split.

Pun Detection Classifiers Building on the post hoc approach from the UvA's 2024 and 2025 participation [11, 12], we use pun detection classifiers to select pun-preserving translations from multiple candidates. This year, our focus is on comparing a *multilingual* classifier with a *language-specific* classifier. Both are trained on the JOKER 2023 pun detection data [13]:

- **XLM-RoBERTa-base** (278M parameters): a multilingual model fine-tuned on the combined English, French, and Spanish detection data. In the final cleaned thesis version, this combined dataset contains 11,292 samples. This model is used to test whether cross-lingual transfer improves pun detection.
- **CamemBERT-base** (111M parameters): a French-specific model fine-tuned on only the French detection data (3,999 samples), pretrained exclusively on French text.

For the final thesis experiments, the language-specific classifiers were trained with learning rate 2e-5, batch size 16, 4 epochs, and FP16 precision, while XLM-RoBERTa used learning rate 1e-5, batch size 8, 4 epochs, and FP32 precision for stability and memory reasons. The best checkpoint was selected based on the validation F1 score. In the final leakage-controlled repeated-split evaluation reported in the thesis, the French-specific CamemBERT model outperforms the multilingual XLM-RoBERTa model on French, with mean F1 scores of 0.718 and 0.584, respectively.

Pipeline For each English pun in the test set, the fine-tuned MarianMT generates five candidate French translations using beam search. Each candidate is scored by the pun detection classifier, and the candidate with the highest pun probability is selected as the final translation. This is a simplified version of the weighted scoring used in [12], selecting purely on pun probability to directly observe the effect of classifier choice. We generate two runs using the same candidates but different classifiers for selection:

- `UAms_task2_MarianMT_XLMRoBERTa`: multilingual XLM-RoBERTa classifier.
- `UAms_task2_MarianMT_CamemBERT`: French-specific CamemBERT classifier.

[4] https://huggingface.co/BAAI/bge-reranker-v2-m3
[5] https://wordnet.princeton.edu/

All experiments were run on Google Colab using a Tesla T4 GPU. Processing all 4,061 test puns took approximately 19 minutes.

Larger translation models Beyond MarianMT, we fine-tuned two larger multilingual translation models on the same corpus. Rather than selecting purely on pun probability, we ranked candidates by a weighted combination of the generation score, the pun probability of the multilingual mDeBERTa-v3-base detector reused from Task 1, and a pun match score. We compared NLLB-200-1.3B (1.3 billion parameters) and mBART-50 (610.9 million parameters) under this setup.[6] [7]

Content preservation selection In addition to selecting candidates purely on pun probability, we experimented with a simple rule-based content preservation score that rewards translations that preserve the named entities and numbers of the source sentence, matched using regular expressions. This score is blended with the detector's pun probability through a single mixing weight during candidate selection.

2.3. Task 3: Onomastic Wordplay Translation

This task asks to translate funny names from English into French. We submitted five runs, shown in Table 1. Each English name comes with a French reference and a short description of the character. The runs are built from three components. The first two reuse the Task 2 MarianMT translator, fine-tuned on the 353 onomastic training pairs. One is a literal system that translates the bare name. The other is a context system that additionally receives the description as input. For each, $k = 6$ beam candidates are scored by a name-specific rule, $s_{\text{name}} = 0.85, s_{\text{base}} + 0.20, s_{\text{hint}} + 0.05, s_{\text{keep}} - 0.15, s_{\text{copy}}$, rewarding any French name hint in the description and penalizing a direct copy of the English name. The third component uses no model. Instead, it extracts the French name directly from the description when a cue is present, such as `"known as X in French"`. Otherwise, it copies the English name. The remaining two runs are hybrids that apply this extraction where a cue is present and otherwise fall back to a translation model. One falls back on the literal model, and the other on the context model.

2.4. Task 4: Humor Generation

This task asks to generate short humorous texts. We submitted four runs in total for the English version, as shown in Table 1.

The goal for each run was to generate puns that use a given homophone. The input to the language model consists of a context and the homophone, along with its senses. A homophone consists of a pun word that appears in the sentence and an implied word. Originally, the methods developed used context as input, selecting a semantically similar homophone to create a pun. As this task returns homophones instead, this method was "reversed" to find compatible context words that support both definitions of the homophone. The contexts are extracted from the SemEval 2017 Task 7 dataset [14], using RAKE.[8] The full pipeline for each run contains the following three sections, in order:

Step 1: Context Retrieval First, the top-2 most relevant context words for each sense of the given homophone are retrieved. The top-2 most relevant context words are determined by the semantic similarity of the context word and either the pun or the implied word, and the semantic similarity between the context word and the senses of the homophone. the top-2 retrieved context words for each sense of the homophone are combined, resulting in a total of four unique sets of contexts per homophone.

[6] https://huggingface.co/facebook/nllb-200-1.3B
[7] https://huggingface.co/facebook/mbart-large-50
[8] https://pypi.org/project/rake-nltk/

Step 2: Pun Generation For every homophone and an associated set of context words, a language model attempts to generate a pun. Each of our four runs used a different model and/or method.

- **Finetuned T5**: Text-to-Text Transfer Transformer (T5) is a series of language models developed by Google, building on the original transformer architecture. As such, they are encoder-decoder models, capable of understanding natural language and producing text as output [15].
  Given a context and a pun word pair, `t5-base` was trained to generate sentences incorporating the context and both senses of the pun word pair. It was trained for 30 epochs, with a learning rate of 1e-4 and an effective batch size of 48
- **Pretrained and finetuned T5**: Before finetuning the model to generate puns, `t5-base` was first pretrained to incorporate words into sentences relating to a certain context. It was trained on data from The Pile, considering only instances originating from Pile-CC and English Wikipedia.[9] For each of the 243 pun words in the final subset, two hundred sentences were mined. For each sentence, context words were extracted using the RAKE algorithm.[10] Then, the model was trained to generate sentences incorporating the context and the given word, of which the output should be the mined sentence. It was trained for 3 epochs, with a learning rate of 1e-4 and an effective batch size of 48. The model was further fine-tuned using the method described in the previous section.
- **Gemma 2**: Gemma 2 is a set of lightweight open models developed by Google. They are based on a decoder-only transformer architecture. The 2B and 9B models were trained with knowledge distillation from larger models.
  For the task of generating puns, the model `google/gemma-2-9b-it`[11] was fine-tuned with the context and pun word pair contained in the input prompt. Version A was trained for 3 epochs, with an effective batch size of 32 and a learning rate of 1e-4. Version B was trained for 5 epochs, with an effective batch size of 32 and a learning rate of 1e-5. Low-Rank Adaptation (LoRA) was used for both to speed up training.

Step 3: Pun Selection For each input homophone, four sentences are generated, one for each of the sets of contexts that were retrieved. Each output is selected based on the ambiguity and distinctiveness measures. A detailed description of how these are calculated can be found in [16]. Ambiguity is a prerequisite for a pun, while distinctiveness is shown to correlate with the funniness of a pun [16]. So, for each context, the final output is selected as the pun with the highest distinctiveness score whose ambiguity score meets a threshold of 0.15. If no text meets this threshold for a certain homophone, the final output is the text with the highest distinctiveness score.

3. Experimental Results

In this section, we will present the results of our experiments in self-contained subsections following the CLEF 2026 JOKER Track tasks.

3.1. Task 1: Humor-aware Information Retrieval

We discuss our results for Task 1, asking to retrieve short humorous texts for a query.

Table 2 shows the performance of the Task 1 submissions on the test data. With the exception of our hybrid humor-weighted runs, whose fusion and humor weights were tuned on the 2026 training qrels, none of our approaches was trained or informed by the train data, nor was pooling used to locate relevant documents (the corpus's recall base is known to be complete).

[9] https://huggingface.co/datasets/monology/pile-uncopyrighted
[10] https://pypi.org/project/rake-nltk/
[11] https://huggingface.co/google/gemma-2-9b-it

Table 2
Evaluation of JOKER Task 1 (test data).

Run	MAP	GMAP	P@R	MRR	Precision				NDCG
					5	10	100	1,000	5
UAms_Task1en_bm25	10.87	0.00	7.85	20.80	6.99	4.72	2.82	0.37	10.35
UAms_Task1en_bm25_filter40	11.27	0.00	7.67	21.13	6.72	4.73	2.27	0.32	10.16
UAms_Task1en_bm25_f1	12.28	0.00	7.77	22.61	7.11	6.00	3.03	0.37	10.40
UAms_Task1en_rm3	10.35	0.00	6.78	18.69	6.37	4.81	2.85	0.38	9.13
UAms_Task1en_rm3_filter40	10.98	0.00	7.11	19.75	6.49	5.11	2.72	0.33	9.46
UAms_Task1en_rm3_f1	13.31	0.00	8.74	23.15	8.29	8.02	3.02	0.30	11.30
UAms_Task1en_BGE_M3_Hybrid_00	6.40	2.63	3.09	11.39	2.84	2.23	2.41	0.34	4.10
UAms_Task1en_BGE_M3_Hybrid_40	7.05	2.82	3.37	12.45	2.82	3.21	2.35	0.31	4.10
UAms_Task1en_BGE_M3_Hybrid_f1	8.75	3.74	5.60	15.88	5.11	6.00	2.46	0.34	4.10
UAms_task1en_hybrid_norerank	29.96	12.18	28.82	56.21	25.46	17.31	2.36	0.29	38.07
UAms_task1en_5000pool	21.91	9.29	20.91	37.67	19.96	16.64	2.36	0.29	26.07
UAms_task1en_hybrid_norerank_wordnet0.3	30.11	10.43	29.97	55.37	26.74	17.51	2.10	0.26	39.24
UAms_task1en_twostage_hw0.6_pool500	28.26	15.30	25.31	51.73	22.74	16.09	2.91	0.38	33.60
UAms_Task1hi_bm25	21.98	14.14	21.10	59.16	24.70	13.69	4.31	0.51	30.58
UAms_Task1hi_rm3	16.34	10.52	16.56	41.64	18.14	12.78	4.34	0.52	20.48
UAms_Task1hi_BGE_M3_Hybrid_00	5.27	3.26	2.71	12.61	2.97	2.02	3.53	0.48	3.93
UAms_Task1hi_BGE_M3_Hybrid_f1	4.05	2.93	0.76	4.94	0.78	0.74	3.63	0.48	3.93

We observe that standard lexical ranking approaches perform reasonably well, albeit not very well, and slightly worse when using blind feedback. The baseline rankers perform somewhat better for the Hinglish data. We hypothesize that this may be due to the nature of the Hinglish corpus, as it is a combination of English and Hindi. As the total vocabulary is larger, the ratio of documents in which a query word appears is smaller, which results in higher precision for keyword-based retrieval systems.

Linearly combining sparse and dense retrieval with BGE-M3 does not improve retrieval measures. This might be because the query is also the pun word in pun sentences. Additionally, pun sentences often convey a different global topic than the pun word, as they need to incorporate the alternative meaning of the pun word. Meanwhile, BGE-M3 prioritizes documents related to the query in any way. Sentences that are more explicit in their relation to the query are ranked higher. This is conflicting with the nature of pun sentences.

For the English submissions, incorporating a pun classifier into the ranking leads to a notable increase in both precision and recall (MRR, NDCG, and MAP). Using the F1-score between the relevance and pun classification results yields better measures than filtering a subset of the results. This is likely because, even if the classifier misclassifies a joke, it still appears in the final ranking. This observation does not extend to the Hinglish data, presumably because the classifier was trained exclusively on English data.

Our strongest English results come from fusing the lexical and dense rankings with RRF and re-weighting them with the pun detector. This hybrid humor-weighted run (UAms_task1en_hybrid_norerank) reaches a MAP of 29.96, above the lexical and BGE-M3 baselines. Two findings stand out. First, adding a cross-encoder re-ranker on top of this pipeline consistently lowers MAP (UAms_task1en_5000pool, 21.91). The re-ranker reorders the top of the ranking toward topical relevance and overrides the humor signal, confirming and strengthening the earlier UvA observation that cross-encoder re-ranking hurts on this task [12]. Second, adding a WordNet polysemy signal does not help: it pushes the MAP from 29.96 to 30.11, but retrieves fewer relevant documents, so we treat it as a negative result. A two-stage variant that first retrieves for recall and then ranks primarily on the humor signal (UAms_task1en_twostage_hw0.6_pool500) trades a little MAP (28.26) for the highest GMAP (15.30) and recall in the table, spreading the gains more evenly across queries.

3.2. Task 2: Wordplay Translation

We continue with Task 2, asking to translate english punning jokes into french. We submitted six runs. Two use the same fine-tuned MarianMT translator with different pun detection classifiers for candidate selection. The remaining four compare larger translation models and a content-preservation selection variant.

Table 3
CLEF 2026 JOKER Task 2: Test results (Pun location score)

Run	Pun Location Score
UAms_task2_MarianMT_XLMRoBERTa	28.11
UAms_task2_MarianMT_CamemBERT	27.32
UAms_task2_marian_c0.0	28.78
UAms_task2_marian_c.25	28.74
UAms_task2_nllb_c.25	26.98
UAms_task2_mbart_c.25	19.29
Top system (dsgt-arc)	37.78

Official Evaluation Results Table 3 shows the results of our Task 2 submissions on the 2026 test data, alongside the top-performing system for context. We make a number of observations.

First, the multilingual classifier run achieves a slightly higher pun location score (28.11) than the language-specific run (27.32), a difference of 0.79 points. This should be interpreted as a pipeline-level result, since the final thesis evaluation does not show higher French detection F1 for the multilingual model. Second, both of our runs outperform the Helsinki-NLP baseline (27.18) from the leading team, which uses the same base translation model without classifier-guided selection. This confirms that classifier-guided candidate selection adds value, continuing the finding from 2025 [12]. Third, the gap with the top system (9.7 points) is expected given the difference in approach: the leading team uses GPT for translation, while our system uses MarianMT (136M parameters) fine-tuned on a free Colab GPU.

Translation Model Scale and Content Preservation Beyond the classifier comparison, we submitted four further runs that vary the translation model and the selection signal. Scaling up the translation model did not help. Fine-tuned NLLB-200 (26.98) and especially mBART-50 (19.29) both fall below the smaller MarianMT (28.78). This indicates that on this small, domain-specific corpus, the additional capacity of the larger models is not effectively used. Adding a rule-based content-preservation signal that rewards keeping the source's named entities and numbers had a negligible effect (28.78 without it versus 28.74 with it), as the beam candidates already preserved most of the source content. The best of these runs therefore remains MarianMT with detector-guided selection.

Analysis of Classifier Disagreement The two classifiers select different translations for 2,547 out of 4,061 puns (62.7%). To analyze whether these differences vary by pun type, the final thesis analysis uses a balanced, manually categorized subset of 60 held-out English puns, comprising 20 homophonic, 20 polysemous, and 20 compound puns. The classifiers selected different translations in 12 of 21 cases (57%), with the disagreement rate varying by pun type: 71% for homophonic, 29% for polysemous, and 71% for compound puns.

The low disagreement on polysemous puns makes sense: both classifiers detect this type well, so they tend to assign the highest score to the same candidate. For homophonic and compound puns, where the classifiers have different strengths, they more often pick different candidates.

Qualitative Evaluation Table 4 shows an example where the two classifiers make different choices. In this compound pun (condiments/condoms), the multilingual classifier assigns a much higher pun

Table 4
CLEF 2026 JOKER Task 2: Translation example where classifiers disagree

Run	Text
Source	Practice safe eating – always use condiments.
Reference	Faites attention à votre santé en hiver, sortez couverts.
UAms_task2_MarianMT_XLMRoBERTa	Pratiquez la sécurité alimentaire – utilisez toujours des condiments. (pun=0.636)
UAms_task2_MarianMT_CamemBERT	Pratiquez une alimentation sans danger – utilisez toujours des condiments. (pun=0.165)

probability (0.636 vs 0.165) and selects a translation that better frames the joke. For polysemous puns such as "Some cardiologists are heartless," both classifiers select the same translation ("Certains cardiologues sont sans coeur"), correctly preserving the double meaning.

As in the 2025 edition [12], we observe that beam search constrains the pipeline to minor variations of the same literal translation, preventing the creative rewriting that pun translation often requires.

3.3. Task 3: Onomastic Wordplay Translation

We continue with Task 3, asking to translate funny names from English into French. We submitted five runs, built from a literal and a context translation model, a rule-based extraction, and two hybrids that combine the extraction with a model fallback. Systems are ranked by exact match against the professional French references.

Table 5
CLEF 2026 JOKER Task 3: Test results (exact match)

Run	Exact Match
UAms_task3_copy_extract	12.29
UAms_task3_extended_context	12.29
UAms_task3_two_option	6.11
UAms_task3_context	5.84
UAms_task3_literal	4.79

Official Evaluation Results Table 5 shows the results of our Task 3 submissions on the 2026 test data. We make three observations. First, the rule-based extraction (`UAms_task3_copy_extract`) and the extraction with a context-model fallback (`UAms_task3_extended_context`) tie for the best exact match (12.29), clearly ahead of the purely neural systems. This is because the character descriptions frequently state the French name directly, so reading it out of the description is more reliable than generating it. Second, the neural literal (4.79) and context (5.84) systems score lowest. A fine-tuned translator tends to invent a plausible-looking but non-matching French form, which exact match penalizes in full. Third, the choice of fallback matters. Where the extraction finds no cue, falling back to copying the English name (`UAms_task3_copy_extract`) or to the context model (`UAms_task3_extended_context`) both reach 12.29, whereas falling back to the literal model (`UAms_task3_two_option`, 6.11) is worse, since the model overwrites names that copying would have preserved.

Qualitative Evaluation Table 6 illustrates the contrast. Where the description contains a cue, the extraction recovers the established French name exactly, including its accents (*Astérix*, *Idéfix*, *Assurancetourix*). The neural translator instead produces a French-looking but wrong form (*Vitalstatistix* becomes *Vitrostatistix*, where the reference is *Abraracourcix*), which receives no credit under exact

Table 6
CLEF 2026 JOKER Task 3: Example outputs of the extraction and neural systems.

Source name	System output	
Asterix	Astérix	correct (extraction)
Dogmatix	Idéfix	correct (extraction)
Cacofonix	Assurancetourix	correct (extraction)
Vitalstatistix	Vitrostatistix	incorrect (reference *Abraracourcix*)

Table 7
CLEF 2026 JOKER Task 4: Results

Run	BLEU	BT rating	CI low	CI high
UAms_Task4_T5ft	2.20	541	467	612
UAms_Task4_T5ptft	2.54	591	528	634
UAms_Task4_Gemma2_A	2.63	727	678	773
UAms_Task4_Gemma2_B	2.55	734	658	774

Table 8
CLEF 2026 JOKER Task 4: Official LLM Arena Results

	Rank	Run	BT	CI low	CI high	n_b
English	74	`Qwen2.5-32B-Instruct`	984	938	1025	316
	81	`Qwen2.5-14B-Instruct`	927	882	959	341
	84	`Mistral-7B-Instruct-v0.2`	922	882	952	341
	96	`flan-t5-large`	549	469	607	326
	98	`flan-t5-base`	422	328	466	327
Spanish	60	`Llama-3-Instruct-Neurona-8b`	1062	1027	1105	315
	113	`Mistral-7B-Instruct-v0.2`	903	858	936	339
	126	`flan-t5-large`	393	276	479	303
	127	`flan-t5-base`	344	272	406	340
	134	`salamandra-7b-instruct`	-270	-1734	-142	335
French	41	`Chocolatine-2-14B-Instruct-v2.0.3`	1191	1149	1230	332
	72	`Mistral-7B-Instruct-v0.2`	961	910	1001	314
	126	`flan-t5-large`	558	490	604	329
	128	`flan-t5-base`	508	442	567	330

match. This confirms that for onomastic wordplay targeting a specific established name rather than a paraphrasable sentence, extraction from the character description outperforms neural translation.

3.4. Task 4: Humor Generation

We discuss the results for Task 4, asking to generate short humorous texts. Table 7 shows the BLEU scores and Bradley-Terry (BT) scores as formed by LLM-as-a-judge and human pairwise votes, along with the 95% confidence interval (CI).

These results show that adding a pre-training stage to the T5 model indeed improves its ability to generate successful puns. Furthermore, the Gemma 2 models improve on the scores achieved by the T5 models, suggesting that a bigger model is more capable of generating puns. There is little difference between the A and B versions of the Gemma models, indicating that changes in hyperparameters have minimal impact.

Table 8 shows the official LLM Arena scores of the Task 4 submissions for all three languages. First, in English, the larger Qwen model[12] outperforms the smaller and older models. Second, in Spanish,

[12]https://huggingface.co/Qwen/Qwen2.5-32B-Instruct

the Spanish-bilingual model[13] outcompetes multilingual models. Third, in French, we also see that the French model[14] outcompetes the multilingual models.

4. Discussion and Conclusions

This paper detailed the University of Amsterdam's participation in the CLEF 2026 JOKER track. We conducted a range of experiments for each of the track's four tasks. Our primary objective was to develop a post hoc approach that leverages wordplay or humorous text detection to filter out humorous translations or search results.

Our main findings are the following. First, we successfully developed effective humor-detection classifiers for English, French, and multilingual settings. Second, for humor-aware information retrieval, we could improve retrieval effectiveness by combining sparse and dense retrieval and reweighting the ranking using a multilingual neural pun detector. However, consistently incorporating a modern cross-encoder re-ranker on top consistently reduced effectiveness, as it reorders results toward topical relevance and overrides the humor signal. In addition, a WordNet polysemy signal added nothing beyond what the trained detector provided. Third, for wordplay translation, we generated multiple translation candidates and compared a multilingual (XLM-RoBERTa) and a language-specific (CamemBERT) pun classifier for selecting the best candidate. The multilingual classifier achieved a higher BLEU score (28.11 vs 27.32), and the two classifiers selected different translations in 62.7% of cases. This shows that classifier choice clearly affects translation selection, but the slightly higher BLEU score of the multilingual pipeline should not be taken as direct evidence of better classifier performance or cross-lingual transfer. Furthermore, scaling up the translation model did not help. The smaller MarianMT outperformed both fine-tuned NLLB-200 and mBART-50, and a rule-based content-preservation signal had a negligible effect. Fourth, for onomastic wordplay translation, our carefully engineered approaches with rule-based extraction and contextual fallback performed best under exact-match conditions. Fifth, for humor generation, we saw the potential of modern models to generate puns when prompted with specific guidance on the pun word and its double meaning. The language-specific nature of the humor generation task is reflected in the effectiveness of Spanish and French models, whereas for English, the largest general model performs best. More generally, together, these results suggest that on this small, domain-specific dataset, the gains stem from how the humor signal is combined rather than from larger models or heavier re-ranking.

Acknowledgments

This research was conducted as part of the final research projects of the Bachelor's in Artificial Intelligence at the University of Amsterdam, We thank the coordinator, Dr. Sander van Splunter, for his support and flexibility to work around the CLEF deadlines. We also thank the track and task organizers for their amazing service and effort in making realistic benchmarks available for analyzing and processing humorous text.

Jan Bakker is partly funded by the Netherlands Organization for Scientific Research (NWO NWA # 1518.22.105). Jaap Kamps is supported by the Netherlands Organization for Scientific Research (NWO CI # CISC.CC.016, 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* for **grammar and spelling checks**, as well as for **paraphrasing and rewording**. *Claude* was used as a debugging tool during development. After using these tools, the authors reviewed and edited the content where

[13] https://huggingface.co/Iker/Llama-3-Instruct-Neurona-8b
[14] https://huggingface.co/jpacifico/Chocolatine-2-14B-Instruct-v2.0.3

necessary and take full responsibility for the final content of this work. All ideas, analyses, and decisions presented in this work are the authors' own.

References

[1] L. Ermakova, R. Campos, A. Bosser, T. Miller, Overview of the CLEF 2025 JOKER lab: Humour in machine, in: J. Carrillo-de-Albornoz, A. G. S. de Herrera, J. Gonzalo, L. Plaza, J. Mothe, F. Piroi, P. Rosso, D. Spina, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction - 16th International Conference of the CLEF Association, CLEF 2025, Madrid, Spain, September 9-12, 2025, Proceedings, Lecture Notes in Computer Science, Springer, 2025, pp. 315–337. URL: https://doi.org/10.1007/978-3-032-04354-2_18. doi:10.1007/978-3-032-04354-2_18.

[2] L. Ermakova, R. Campos, A. Bosser, T. Miller, Overview of the CLEF 2025 JOKER task 1: Humour-aware information retrieval, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum, CLEF 2025, Madrid, Spain, 9-12 September 2025, CEUR Workshop Proceedings, CEUR-WS.org, 2025, pp. 2744–2760. URL: https://ceur-ws.org/Vol-4038/paper_218.pdf.

[3] L. Ermakova, R. Campos, A. Bosser, T. Miller, Overview of the CLEF 2025 JOKER task 2: Wordplay translation from english into french, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum, CLEF 2025, Madrid, Spain, 9-12 September 2025, CEUR Workshop Proceedings, CEUR-WS.org, 2025, pp. 2761–2779. URL: https://ceur-ws.org/Vol-4038/paper_219.pdf.

[4] 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.

[5] P. Vachharajani, et al., Overview of the CLEF 2026 JOKER Task 1: Humor-Aware Information Retrieval in English and Hinglish, in: [17], 2026.

[6] L. Ermakova, et al., Overview of the CLEF 2026 JOKER Task 2: Pun Translation from English to French, in: [17], 2026.

[7] L. Ermakova, et al., Overview of the CLEF 2026 JOKER Task 3: Onomastic Wordplay Translation from English to French, in: [17], 2026.

[8] I. Kuzmin, et al., Overview of the CLEF 2026 JOKER Task 4: Humor Generation, in: [17], 2026.

[9] J. Lin, X. Ma, S. Lin, J. Yang, R. Pradeep, R. F. Nogueira, Pyserini: A python toolkit for reproducible information retrieval research with sparse and dense representations, in: F. Diaz, C. Shah, T. Suel, P. Castells, R. Jones, T. Sakai (Eds.), SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021, ACM, 2021, pp. 2356–2362. URL: https://doi.org/10.1145/3404835.3463238. doi:10.1145/3404835.3463238.

[10] G. V. Cormack, C. L. A. Clarke, S. Büttcher, 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, ACM, New York, NY, USA, 2009, pp. 758–759. doi:10.1145/1571941.1572114.

[11] E. Schuurman, M. Cazemier, L. Buijs, J. Kamps, University of amsterdam at the CLEF 2024 joker track, in: G. Faggioli, N. Ferro, P. Galuscáková, A. G. S. de Herrera (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), Grenoble, France, 9-12 September, 2024, CEUR Workshop Proceedings, CEUR-WS.org, 2024, pp. 1909–1922. URL: https://ceur-ws.org/Vol-3740/paper-181.pdf.

[12] A. Kreefft-Libiu, F. Helms, C. Selçük, J. Bakker, J. Kamps, University of Amsterdam at the CLEF

2025 Joker track, in: Working Notes of CLEF 2025: Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[13] L. Ermakova, T. Miller, A. Bosser, V. M. Palma-Preciado, G. Sidorov, A. Jatowt, Overview of JOKER 2023 automatic wordplay analysis task 1 - pun detection, in: M. Aliannejadi, G. Faggioli, N. Ferro, M. Vlachos (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2023), Thessaloniki, Greece, September 18th to 21st, 2023, CEUR Workshop Proceedings, CEUR-WS.org, 2023, pp. 1785–1803. URL: https://ceur-ws.org/Vol-3497/paper-149.pdf.

[14] T. Miller, C. Hempelmann, I. Gurevych, SemEval-2017 task 7: Detection and interpretation of English puns, in: S. Bethard, M. Carpuat, M. Apidianaki, S. M. Mohammad, D. Cer, D. Jurgens (Eds.), Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 58–68. URL: https://aclanthology.org/S17-2005/. doi:10.18653/v1/S17-2005.

[15] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, Journal of machine learning research 21 (2020) 1–67.

[16] J. T. Kao, R. Levy, N. D. Goodman, A computational model of linguistic humor in puns, Cognitive Science 40 (2016) 1270–1285. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/cogs.12269. doi:https://doi.org/10.1111/cogs.12269. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/cogs.12269.

[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.