Overview of the CLEF 2024 JOKER Task 1: Humour-aware Information Retrieval

Liana Ermakova[1,*], Anne-Gwenn Bosser[2], Tristan Miller[3,4] and Adam Jatowt[5]

[1]Université de Bretagne Occidentale, HCTI, France
[2]École Nationale d'Ingénieurs de Brest, Lab-STICC CNRS UMR 6285, France
[3]Department of Computer Science, University of Manitoba, Winnipeg, Canada
[4]Austrian Research Institute for Artificial Intelligence (OFAI), Vienna, Austria
[5]University of Innsbruck, Austria

Abstract

This paper presents the details of Task 1 of the JOKER-2024 Track, where the aim is to retrieve short humorous texts from an underlying document collection. The intended use case for this task is to search for a joke on a specific topic. This can be useful for humour researchers in the humanities, for second-language learners as a learning aid, for professional comedians as a writing aid, and for translators who might need to adapt certain jokes to other cultures. For this task, we provided a collection consisting of 61,268 documents, where 4,492 texts were humorous. Ten teams submitted 26 runs in total for this task.

Keywords

information retrieval, wordplay, puns, computational humour, wordplay detection, test collection,

1. Introduction

This paper presents details of Task 1 of the JOKER-2024 Track[1], which was held as part of the 15th Conference and Labs of the Evaluation Forum (CLEF 2024)[2]. The overall objective of the JOKER track series [1, 2, 3], which began in 2022, is to facilitate collaboration among linguists, translators, and computer scientists to advance the development of automatic humour analysis. In each edition of the JOKER track, we construct and publish reusable, quality-controlled datasets to serve as training and test data for various humor processing tasks. In Task 1, participants build systems aiming to retrieve short humorous texts from a document collection based on a given query. For details on JOKER-2024's other two tasks, we refer the reader to their respective overviews [4, 5]. Further information and insights are also presented in the Track's overview paper [1].

Search engines generally do not account for humour, ambiguity, or subversion of linguistic rules as features for selecting relevant documents to be returned. However, humour-aware retrieval, such as retrieval of wordplay-containing passages, can be useful for certain use cases or for user groups who appreciate or are interested in humorous qualities of text [6, 7].

To foster research in humour-aware information retrieval, in JOKER 2024 we have introduced a novel task that consists of retrieving short humorous texts from a document collection. The intended use case is to search for a humorous text on a particular topic. Besides users who especially like humour, this could be useful for writers, for humour researchers, for second-language learners as a learning aid, for advertisement copywriters, for professional comedians as a writing aid, or even for translators who might need to adapt certain jokes to other cultures. Formally, for Task 1, the objective is to retrieve short humorous texts from a document collection based on a given query. The retrieved texts should fulfill two criteria: to be relevant to the query, and to be humorous, which in our task means to be

CLEF 2024: Conference and Labs of the Evaluation Forum, September 9–12, 2024, Grenoble, France
*Corresponding author.
0000-0002-7598-7474 (L. Ermakova); 0000-0002-0442-2660 (A. Bosser); 0000-0002-0749-1100 (T. Miller); 0000-0001-7235-0665 (A. Jatowt)

[1]https://www.joker-project.com
[2]https://clef2024.clef-initiative.eu/

Table 1
Statistics on the runs submitted to the CLEF JOKER 2024 Task 1

Team	# of runs
jokester [12]	1
LIS [13]	1
Arampatzis	10
Frane	1
Dajana&Kathy	1
AB&DPV [14]	1
RubyAiYoungTeam	1
Petra&Regina [15]	1
Tomislav&Rowan [16]	1
UAms [17]	8
Total	26

instances of wordplay. The intended use case is to search for a joke on a specific topic. For example, a search query of "math" would mean that the goal is to find math jokes, while the query "Tom" would mean that the goal is to find jokes about some person or entity named Tom.

The test collection was built based on the English corpora constructed within the previous edition of the CLEF JOKER track:

- JOKER 2023 Task 1 - pun detection [2, 8, 9];
- JOKER 2023 Task 2 - pun location and interpretation [2, 10, 9];
- JOKER 2023 Task 3 - pun translation [2, 11, 9].

This year, ten teams, out of the total 22 active JOKER participants, submitted 26 runs for Task 1 out of the 103 runs submitted to the track (see run statistics in Table 1).

This paper presents an overview of our data preparation process in Section 2. In Section 3, we describe the participants' runs, and we present the analysis of their results in Section 4. We provide some concluding remarks in Section 5.

2. Dataset

The data for this task extends that which was originally used for JOKER-2023's tasks on wordplay detection in English [8, 2, 9]. Those texts were annotated according to whether they are humorous; we supplement this data with texts from Task 3 of JOKER-2023 [11, 2, 9], used for humour translation, and with some new wordplay instances. We further extended the data with text passages collected from non-humourous sources, as well as data that was automatically generated in relation to the queries. Specifically, the non-humorous data related to queries was obtained from the following sources:

- negative examples from the JOKER corpus.
- Wikipedia extracts returned for the queries. We used the Wikipedia Python package[3] for this and then collected sentences to form non-humorous text instances.
- Descriptions of queries generated by Meta's Llama 2 with 7B parameters [18].

In total, we provided our participants with a collection consisting of 61,268 documents, where 4,492 texts are humorous. The latter encompasses 3,507 texts from JOKER 2023 and 985 new wordplay instances. The remaining 56,776 texts are non-humorous. These consist of 4,954 negative examples taken from the JOKER 2023 wordplay detection corpus, 12,523 texts generated using Llama 2, and 39,299 sentences from Wikipedia extracts. All the texts were typically one or two sentences long and were released in the form of JSON files.

[3] https://pypi.org/project/wikipedia/

Table 2: Statistics of relevant humourous texts per query

count	57
mean	30
std	43
min	1
25%	8
50%	18
75%	38
max	281

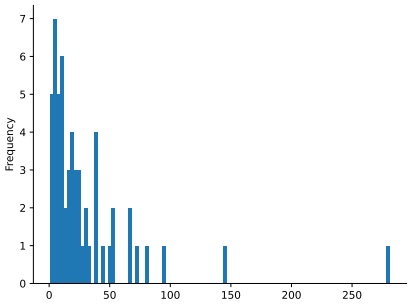

Figure 1: Histogram of # relevant humourous texts

For creating the set of queries, we harnessed data from CLEF 2023 JOKER Task 2 – Pun Location and Interpretation [10, 2, 9], and in particular, the locations of wordplay in texts, i.e. words or phrases carrying multiple meanings. In CLEF 2023 JOKER Task 2, puns were either homographic (identical spelling as in *I used to be a banker but I lost interest*) or heterographic (i.e. exploiting paronymy as *propane/prophane* in *When the church bought gas for their annual barbecue, proceeds went from the sacred to the propane.*) To expand the queries, we used the semantic annotations of pun locations (pun interpretation), i.e. pairs of lemmatized word sets, containing the synonyms (or, if absent, hypernyms) of the two words involved in the pun, excluding any that share the same spelling as the pun. The lists of query expansions were manually checked. The document was deemed humorous and relevant to the query if it came from the positive examples of the JOKER corpus and included the query term or its expansions.

Twelve queries with their judgments (qrels) were created for training or validating participants' systems. Then, another 45 queries were created as a test set.[4] For all 57 queries (combined test and training), 11,831 documents were deemed topically relevant. We considered a document to be topically relevant to a given query if it contained the term from this query, or its synonyms, or its hyperonyms. Among the topically relevant documents, 1,730 were considered to be humorous. The descriptive statistics of relevant humourous texts is given in Table 2 while Figure 1 presents the histogram of the number of relevant humourous texts per query. The average number of relevant humorous texts per query is 30, while the median is 18 texts.

2.1. Evaluation Measures

When it comes to evaluation measures, a set of standard information retrieval metrics were used:

map mean average precision – i.e., the mean of the average precision scores for each query

ndcg normalised discounted cumulative gain, the gain of each document based on its relevance, discounted logarithmically by its position in the ranking normalised over the ideal ranking

P1, P5, P10 precision – i.e. the ability of a system to present only relevant items, at different numbers of top ranked results

R5, R10, R100, R1000 recall – measuring the ability of systems to find all or many relevant items, at different top numbers of results

bpref binary preference, a sum-based metric showing how many relevant documents are ranked before irrelevant documents

[4]Note that we also included all the training-set queries in the test input file; however, they are excluded from the resulting scores.

MRR mean reciprocal rank, the average of the multiplicative inverse of the ranks of the first correct answer of results for a sample of queries

We used the `pyterrier` platform [19, 20] implementation of these metrics.

2.2. Input format

2.2.1. Document collection

We provide the training and test data in a JSON format with the following fields:

docid a unique document identifier

text the text of the instance, which may or may not contain wordplay

Input example:

```
[
  {
    "docid": "1",
    "text":  "Good laws have sprung from bad customs."
  },
  {
    "docid": "2",
    "text":  "The musical score to Topsyturveydom does not survive, but amateur
        productions in recent decades have used newly composed scores or performed the
        work as a non-musical play."
  },
  {
    "docid": "3",
    "text": "The organic compound primarily responsible for the characteristic odor of
        musk is muscone."
  },
  {
    "docid": "51135",
    "text":  "I've inherited a fortune, said Tom, willfully"
  },
  {
    "docid": "591",
    "text":  "My name is Will, I'm a lawyer."
  }
]
```

2.2.2. Queries

The train and test queries are also JSON files, this time with the following fields:

qid a unique query identifier from the input file

query the search query

Input example:

```
[
  {"qid":"qid_train_1","query":"steps"},
  {"qid":"qid_train_3","query":"math"},
  {"qid":"qid_train_4","query":"Tom"}
]
```

2.2.3. Qrels

Finally, we provide training/validation data in the format of JSON qrels files with the following fields:

qid a unique query identifier from the query input file

docid a unique document identifier from the corpus

qrel indication the document docid is relevant to the query qid and is a wordplay instance

Example of a qrel file:

```
[
  {
    "qid":   "qid_train_0",
    "docid": "27260",
    "qrel":  0
  },
  {
    "qid":   "qid_train_0",
    "docid": "591",
    "qrel":  1
  },
  {
    "qid":  "qid_train_0",
    "docid": "51135",
    "qrel":1
  }
]
```

2.3. Output format

We required results to be provided in a JSON format with the following fields:

run_id run ID starting with <team_id>_<task_id>_<method_used>, e.g. UBO_task_1_TFIDF

manual flag indicating if the run is manual 0,1

qid a unique identifier from the input file

docid an identifier of the document retrieved from the corpus to the qid query

rank retrieved document rank

score normalised document relevance score (in the [0–1] scale)

For each query, the maximum allowed number of distinct documents (docid field) is 1000. A sample output file is as follows:

```
[
  {
    "run_id":"team1_task_1_TFIDF",
    "manual":0,
    "qid":"qid_train_0",
    "docid":"27260",
    "rank":1,
    "score":0.97
  },
  {
    "run_id":"team1_task_1_TFIDF",
```

```
    "manual":0,
    "qid":"qid_train_0",
    "docid":"591",
    "rank":2,
    "score":0.8
  },
  {
    "run_id":"team1_task_1_TFIDF",
    "manual":0,
    "qid":"qid_train_1",
    "docid":"27261",
    "rank":1,
    "score":0.7
  }
]
```


3. Participants' Approaches

In total, ten teams submitted 26 runs (see run statistics in Table 1). The approaches used by the participating teams are as follows:

- The **jokester** team [12] provided a single run based on an approach that uses TF–IDF for feature weighting and a Logistic Regression classifier.
- The **Arampatzis** team[5] provided ten runs, testing a range of diverse models such as TF–IDF, LSTM, Random Forest, XGBoost, LightGBM (Light Gradient-Boosting Machine), SVM, Decision Tree, Gaussian Naive Bayes, KNN, and neural nets.
- A run submitted by **LIS** team [13] was based on T5 transformer model, query processing, expanding terms with their synonyms collected from WordNet, choosing the optimal tokenisation method for queries and documents, and then selecting the best threshold for the similarity score. Finally, a pre-trained model was applied to filter texts with puns.
- The **Frane** uses fine-tuned BERT models for estimating humorousness together with the well-known retrieval model such as BM25. The team submitted one run.
- The **Dajana&Kathy** team processed text using stemming, lemmatisation, and stop word removal, and employed TF–IDF and BM25, together with fine-tuning BERT for submitting their run
- The **AB&DPV** team [14] used TF–IDF for ranking humourous text within the collection for constructing thier run.
- The **RubyAiYoungTeam** team's run was submitted without any description of the employed method.
- The **Petra&Regina** team [15] submitted a single run employing logistic regression with TF–IDF vectorised documents and queries and iterative relevance scoring.
- The **Tomislav&Rowan** team [16] employed logistic regression with TF–IDF vectorised documents to create a single run.
- The **UAms** team [17] provided two runs based on BM25 and BM25+RM3 using default settings. Two other runs employ neural cross-encoder rerankings of the latter runs based on zero-shot application of an MSMARCO-trained ranker. The last four runs are based on two trained versions of the SimpleT5 model, one with a batch size of 6 and the other with a batch size of 8 and a trained BERT model using LoRa.

Note that we do not detail the zero-scored runs, nor the runs with problems that we could not resolve.

[5] No paper received

Table 3
Results on the test data. (Boldface indicates the best result per metric.)

run ID	map	ndcg	R5	R10	R100	R1000	bpref	MRR	P1	P5	P10
UAms_rm3_T5_Filter2	**0.12**	**0.28**	**0.09**	**0.15**	0.36	0.43	0.18	0.26	0.13	0.11	0.13
UAms_rm3_BERT_Filter	**0.12**	0.27	**0.09**	0.14	0.35	0.42	0.16	0.27	0.16	0.11	0.12
UAms_rm3_T5_Filter1	0.11	0.27	0.09	**0.15**	0.36	0.42	0.16	0.23	0.11	0.09	0.11
UAms_bm25_BERT_Filter	0.09	0.24	0.06	0.12	0.37	0.40	0.12	0.19	0.09	0.05	0.08
AB&DPV_TFIDF	0.09	0.24	0.07	0.13	0.33	0.37	0.10	0.25	0.13	0.12	**0.14**
UAms_Anserini_rm3	0.08	0.27	0.06	0.08	**0.38**	**0.50**	0.09	0.20	0.11	0.06	0.06
jokester_TFIDF_LogRegr	0.08	0.19	**0.09**	0.09	0.10	0.16	**0.21**	**0.51**	**0.44**	**0.23**	**0.14**
UAms_Anserini_bm25	0.08	0.24	0.06	0.08	0.37	0.42	0.09	0.19	0.11	0.05	0.06
UAms_bm25_CE100	0.04	0.17	0.03	0.04	0.37	0.37	0.06	0.08	0.00	0.04	0.03
UAms_rm3_CE100	0.04	0.18	0.03	0.04	**0.38**	0.38	0.06	0.07	0.00	0.04	0.03
LIS_MiniLM-T5	0.02	0.05	0.03	0.04	0.05	0.05	0.05	0.13	0.04	0.06	0.04

4. Results

4.1. Evaluation on Test Data

The majority of submitted runs had some issues – for example, some runs were submitted on only part of the data, and there were runs for the training data only. We tried to solve these problems whenever possible.

In Table 3 we show the main results for participants' runs on test data. We make the following observations based on the results:

- First, in general, both precision and recall are extremely low. Low precision is due to the presence of the query terms in the non-humorous texts which is considered as topical relevance by the retrieval systems. The low recall is probably related to the length of the text and the fact that in many texts, both humorous and topically relevant, the query terms do not appear.
- The runs based on pseudo-relevance feedback RM3 query expansion outperform the BM25 baselines.
- Cross-encoder rerankers do not exhibit better performance than the baseline models.
- Filtering trained on the wordplay detection task improved systems' results quite a lot.
- Simple solutions such as ones with TF–IDF and Logistic Regression remain quite competitive.
- Using T5 and BERT language models with RM3 is one of best approaches both in terms of precision and recall.

To evaluate the errors produced by the rankers, we compared the results with those obtained using topic relevance alone, disregarding the humorousness of the texts. Topical relevance results on the test data are given in Table 4. Traditional models without filtering, such as RM3, TF-IDF, BM25, showed high performance on topical relevance only with $MAP > 0.35$ and $NDCG > 0.55$ but the official results, which take into account humorousness of the texts, go down with $MAP < 0.1$ and $NDCG < 0.3$. Post-filtering applied with different ranking models improved MAP up to 50% (cf. UAms_rm3_T5_Filter2 and UAms_Anserini_rm3) according to the official results but dropped topical relevance. UAms_bm25_BERT_Filter demonstrated high scores according to both the official results and topic relevance alone.

4.2. Evaluation on Training Data

Here we also report the results on the training data in order to provide additional insights as to the performance and characteristics of different approaches. Table 5 shows the performance based on the submitted runs.

Looking at the results we can make the following observations:

Table 4
Topical relevance results on the test data.

run ID	map	ndcg	R5	R10	R100	R1000	bpref	MRR	P1	P5	P10
UAms_Anserini_rm3	0.37	0.60	0.06	0.10	0.39	0.64	0.64	0.82	0.73	0.61	0.61
AB&DPV_TFIDF	0.36	0.53	0.07	0.12	0.36	0.50	0.50	0.83	0.73	0.69	0.69
UAms_Anserini_bm25	0.35	0.55	0.07	0.11	0.38	0.56	0.56	0.79	0.64	0.61	0.60
UAms_bm25_BERT_Filter	0.30	0.48	0.07	0.11	0.35	0.46	0.46	0.77	0.62	0.62	0.60
UAms_rm3_T5_Filter1	0.25	0.44	0.06	0.10	0.30	0.40	0.40	0.86	0.78	0.69	0.63
UAms_rm3_CE100	0.22	0.40	0.05	0.10	0.39	0.39	0.39	0.79	0.64	0.56	0.55
UAms_rm3_BERT_Filter	0.22	0.39	0.06	0.09	0.27	0.34	0.34	0.84	0.76	0.68	0.61
UAms_bm25_CE100	0.22	0.39	0.05	0.10	0.38	0.38	0.38	0.78	0.62	0.56	0.55
UAms_rm3_T5_Filter2	0.22	0.38	0.06	0.10	0.27	0.34	0.34	0.80	0.64	0.71	0.63
jokester_TFIDF_LogRegr	0.03	0.09	0.03	0.03	0.04	0.05	0.07	0.63	0.62	0.39	0.24
LIS_MiniLM-T5	0.01	0.05	0.02	0.02	0.03	0.03	0.03	0.33	0.18	0.20	0.15

Table 5
Results on the training data. (Boldface indicates the best result per metric.)

run_id	map	ndcg	R5	R10	R100	R1000	bpref	MRR	P1	P5	P10
Arampatzis_DecisionTree	**0.40**	**0.55**	0.24	**0.30**	0.44	0.45	0.42	**0.92**	**0.92**	**0.68**	**0.53**
Arampatzis_SVM	0.36	0.52	**0.25**	0.28	0.44	0.45	0.39	0.83	0.75	**0.68**	0.52
Arampatzis_kNN	0.36	0.50	0.23	0.28	0.44	0.45	0.38	0.71	0.50	0.60	0.51
Arampatzis_GaussianNB	0.35	0.50	0.24	0.28	0.44	0.45	0.38	0.72	0.58	0.63	0.51
UAms_rm3_T5_Filter2	0.23	0.39	0.14	0.25	0.44	0.52	0.35	0.34	0.17	0.28	0.28
UAms_rm3_BERT_Filter	0.23	0.42	0.12	0.23	**0.50**	0.60	0.36	0.37	0.17	0.23	0.23
UAms_rm3_T5_Filter1	0.21	0.37	0.13	0.24	0.40	0.49	0.29	0.38	0.25	0.25	0.27
UAms_bm25_BERT_Filter	0.19	0.37	0.07	0.19	0.49	0.59	0.27	0.22	0.08	0.12	0.18
UAms_Anserini_rm3	0.17	0.37	0.09	0.18	0.45	**0.63**	0.30	0.24	0.08	0.17	0.18
Arampatzis_NeuralNetwork	0.17	0.34	0.09	0.17	0.43	0.45	0.14	0.41	0.33	0.28	0.25
Arampatzis_LSTM	0.17	0.33	0.09	0.19	0.44	0.45	0.11	0.20	0.08	0.18	0.19
ABDPV_TFIDF	0.17	0.34	0.07	0.14	0.39	0.50	0.21	0.26	0.17	0.15	0.16
UAms_Anserini_bm25	0.16	0.35	0.07	0.17	0.46	0.60	0.24	0.19	0.08	0.12	0.16
jokester_TFIDF_LogRegr	0.16	0.34	0.11	0.12	0.14	0.36	**0.49**	0.59	0.58	0.30	0.20
UAms_rm3_CE100	0.07	0.22	0.01	0.03	0.45	0.45	0.09	0.12	0.00	0.08	0.09
UAms_bm25_CE100	0.07	0.22	0.01	0.03	0.46	0.46	0.09	0.12	0.00	0.08	0.08
LIS_MiniLM-T5	0.00	0.01	0.00	0.00	0.01	0.01	0.01	0.01	0.00	0.00	0.00

- While precision is quite high, recall still poses many challenges, even on training data. This may also support the hypothesis that low recall may arise due to the absence of query terms in texts which are relatively short.
- The approaches using decision trees, and relatively standard approaches like SVM and kNN, achieve the best results. However, these results are from a team (Arampatzis) that have not submitted a system description paper, so they should be treated with caution.
- Considering the remaining results, the ordering of the best runs is similar to that of the test data.
- Considering topical relevance alone on the training data (see Table 6), in general, we observe similar trends as seen in the test set where unfiltered runs tend to have higher topical relevance alone but a significant drop according to the official ranking.
- The filtered runs exhibited identical scores in both the official ranking and for topical relevance alone, indicating that they retrieved (almost) exclusively humorous documents. However, the relative ranking between filtered and unfiltered runs differs, except for the Arampatzis runs, which are at the top of both tables.
- The topical relevance scores between the test and training data are similar, but the ranking that considers both topical relevance and humor is nearly twice as low on the test data, indicating potential overfitting in humor classification (cf. the UAms runs).

Table 6
Topical relevance results on the training data.

run ID	map	ndcg	R5	R10	R100	R1000	bpref	MRR	P1	P5	P10
Arampatzis_DecisionTree	0.40	0.55	0.24	0.30	0.44	0.45	0.42	0.92	0.92	0.68	0.53
AB&DPV_TFIDF	0.38	0.56	0.08	0.13	0.36	0.58	0.58	0.72	0.50	0.67	0.65
Arampatzis_SVM	0.36	0.52	0.25	0.28	0.44	0.45	0.39	0.83	0.75	0.68	0.52
Arampatzis_kNN	0.36	0.50	0.23	0.28	0.44	0.45	0.38	0.71	0.50	0.60	0.51
UAms_Anserini_rm3	0.35	0.58	0.05	0.09	0.37	0.67	0.67	0.73	0.58	0.58	0.52
UAms_Anserini_bm25	0.35	0.57	0.06	0.11	0.37	0.65	0.65	0.66	0.50	0.55	0.53
Arampatzis_GaussianNB	0.35	0.50	0.24	0.28	0.44	0.45	0.38	0.72	0.58	0.63	0.51
UAms_bm25_BERT_Filter	0.30	0.50	0.06	0.12	0.34	0.52	0.52	0.66	0.50	0.57	0.58
UAms_rm3_T5_Filter1	0.25	0.42	0.06	0.11	0.28	0.39	0.39	0.73	0.67	0.58	0.62
UAms_rm3_T5_Filter2	0.23	0.39	0.14	0.25	0.44	0.52	0.35	0.34	0.17	0.28	0.28
UAms_rm3_BERT_Filter	0.23	0.42	0.12	0.23	0.50	0.60	0.36	0.37	0.17	0.23	0.23
UAms_rm3_CE100	0.20	0.37	0.05	0.08	0.37	0.37	0.37	0.81	0.67	0.52	0.52
UAms_bm25_CE100	0.20	0.37	0.05	0.08	0.37	0.37	0.37	0.81	0.67	0.52	0.50
Arampatzis_NeuralNetwork	0.17	0.34	0.09	0.17	0.43	0.45	0.14	0.41	0.33	0.28	0.25
Arampatzis_LSTM	0.17	0.33	0.09	0.19	0.44	0.45	0.11	0.20	0.08	0.18	0.19
jokester_TFIDF_LogRegr	0.06	0.17	0.03	0.03	0.04	0.17	0.22	0.59	0.58	0.30	0.21
LIS_MiniLM-T5	0.00	0.02	0.01	0.01	0.01	0.01	0.01	0.23	0.08	0.08	0.09

5. Conclusions

This paper has given an overview and discussed the results of Task 1 of the JOKER-2024 challenge on the retrieval of humorous texts. Based on the data for wordplay detection and interpretation previously constructed within the CLEF JOKER track [8, 10, 2, 9], we constructed a unique reusable test collection for wordplay retrieval in English.

Ten participating teams submitted 26 runs in total for Task 1. The teams applied diverse methods, ranging from traditional approaches rankers such as TF–IDF, BM25, and RM3 to cross-encoders with and without post-filtering based on classical machine learning methods (logistic regression, and SVMs) to more modern ones, including SimpleT5 and BERT.

The participants' results confirm that humour-oriented information retrieval remains a rather challenging task with both precision and recall being extremely low. Filtering trained on the wordplay detection task significantly improved the systems' results. However, while topical relevance scores between the test and training data are similar, the ranking that considers both topical relevance and humor is nearly twice as low on the test data, suggesting potential overfitting in humor classification.

In general, our results confirm that retrieval models are humour-agnostic and humour detection is still a challenge for machine learning models and LLMs. Developing new test collections, including those for non-English languages, could help address this issue.

Additional information on the track is available on the JOKER website: https://www.joker-project.com/

Acknowledgments

This project has received a government grant managed by the National Research Agency under the program "Investissements d'avenir" integrated into France 2030, with the Reference ANR-19-GURE-0001. This track would not have been possible without the great support of numerous individuals. We want to thank in particular the colleagues and the students who participated in data construction and evaluation, in particular the students of the Université de Bretagne Occidentale. Please visit the JOKER website for more details on the track.[6]

[6] https://joker-project.com/

References

[1] L. Ermakova, T. Miller, A. Bosser, V. M. Palma-Preciado, G. Sidorov, A. Jatowt, Overview of CLEF 2024 JOKER track on automatic humor analysis, in: L. Goeuriot, G. Q. Philippe Mulhem, 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), Lecture Notes in Computer Science, Springer, 2024.

[2] L. Ermakova, T. Miller, A.-G. Bosser, V. M. P. 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.), CLEF'23: Proceedings of the Fourteenth International Conference of the CLEF Association, Lecture Notes in Computer Science, Springer, 2023.

[3] 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.

[4] V. M. Palma Preciado, G. Sidorov, L. Ermakova, A.-G. Bosser, A. Jatowt, Overview of the CLEF 2024 JOKER Task 2: Humour classification according to genre and technique, in: G. Faggioli, N. Ferro, P. Galuscakova, A. G. Seco de Herrera (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), CEUR Workshop Proceedings, CEUR-WS.org, 2024.

[5] L. Ermakova, A.-G. Bosser, T. Miller, A. Jatowt, Overview of the CLEF 2024 JOKER Task 3: Translate puns from English to French, in: G. Faggioli, N. Ferro, P. Galuscakova, A. G. Seco de Herrera (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), CEUR Workshop Proceedings, CEUR-WS.org, 2024.

[6] D. Gupta, M. Digiovanni, H. Narita, K. Goldberg, Jester 2.0 (demonstration abstract): Collaborative filtering to retrieve jokes, in: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '99, Association for Computing Machinery, New York, NY, USA, 1999, p. 333. URL: https://doi.org/10.1145/312624.312770. doi:10.1145/312624.312770.

[7] L. Friedland, J. Allan, Joke retrieval: Recognizing the same joke told differently, in: Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM '08, Association for Computing Machinery, New York, NY, USA, 2008, p. 883–892. URL: https://doi.org/10.1145/1458082.1458199. doi:10.1145/1458082.1458199.

[8] L. Ermakova, T. Miller, A.-G. 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 CLEF 2023 – Conference and Labs of the Evaluation Forum, volume 3497 of *CEUR Workshop Proceedings*, 2023, pp. 1785–1803.

[9] L. Ermakova, A.-G. Bosser, A. Jatowt, T. Miller, The JOKER Corpus: English–French parallel data for multilingual wordplay recognition, in: SIGIR '23: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, Association for Computing Machinery, New York, NY, 2023. doi:10.1145/3539618.3591885, to appear.

[10] L. Ermakova, T. Miller, A.-G. Bosser, V. M. Palma Preciado, G. Sidorov, A. Jatowt, Overview of JOKER 2023 Automatic Wordplay Analysis Task 2 – pun location and interpretation, in: M. Aliannejadi, G. Faggioli, N. Ferro, M. Vlachos (Eds.), Working Notes of CLEF 2023 – Conference and Labs of the Evaluation Forum, volume 3497 of *CEUR Workshop Proceedings*, 2023, pp. 1804–1817.

[11] L. Ermakova, T. Miller, A.-G. Bosser, V. M. Palma Preciado, G. Sidorov, A. Jatowt, Overview of JOKER 2023 Automatic Wordplay Analysis Task 3 – pun translation, in: M. Aliannejadi, G. Faggioli, N. Ferro, M. Vlachos (Eds.), Working Notes of CLEF 2023 – Conference and Labs of the Evaluation

Forum, volume 3497 of *CEUR Workshop Proceedings*, 2023, pp. 1818–1827.
[12] H. Baguian, H. N. Ashley, JOKER Track @ CLEF 2024: The Jokesters' approaches for retrieving, classifying, and translating wordplay, in: G. Faggioli, N. Ferro, P. Galuscakova, A. García Seco de Herrera (Eds.), Working Notes of CLEF 2024 – Conference and Labs of the Evaluation Forum, 2024.
[13] A. Gepalova, A.-G. Chifu, S. Fournier, CLEF 2024 JOKER Task 1: Exploring pun detection using the T5 transformer model, in: G. Faggioli, N. Ferro, P. Galuscakova, A. García Seco de Herrera (Eds.), Working Notes of CLEF 2024 – Conference and Labs of the Evaluation Forum, 2024.
[14] D. P. Varadi, A. Bartulović, JOKER 2024 by AB&DPV: From 'LOL' to 'MDR' using AI models to retrieve and translate puns, in: G. Faggioli, N. Ferro, P. Galuscakova, A. García Seco de Herrera (Eds.), Working Notes of CLEF 2024 – Conference and Labs of the Evaluation Forum, 2024.
[15] R. Elagina, P. Vučić, Convergential approach in machine learning for effective humour analysis and translation, in: G. Faggioli, N. Ferro, P. Galuscakova, A. García Seco de Herrera (Eds.), Working Notes of CLEF 2024 – Conference and Labs of the Evaluation Forum, 2024.
[16] R. Mann, T. Mikulandric, CLEF 2024 JOKER Tasks 1–3: Humour identification and classification, in: G. Faggioli, N. Ferro, P. Galuscakova, A. García Seco de Herrera (Eds.), Working Notes of CLEF 2024 – Conference and Labs of the Evaluation Forum, 2024.
[17] L. Buijs, M. Cazemier, E. Schuurman, J. Kamps, University of Amsterdam at the CLEF 2024 Joker Track, in: G. Faggioli, N. Ferro, P. Galuscakova, A. García Seco de Herrera (Eds.), Working Notes of CLEF 2024 – Conference and Labs of the Evaluation Forum, 2024.
[18] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al., Llama 2: Open foundation and fine-tuned chat models, arXiv preprint arXiv:2307.09288 (2023).
[19] C. Macdonald, N. Tonellotto, Declarative experimentation ininformation retrieval using pyterrier, in: Proceedings of ICTIR 2020, 2020.
[20] C. Van Gysel, M. de Rijke, Pytrec_eval: An extremely fast python interface to trec_eval, in: SIGIR, ACM, 2018.