Overview of JOKER 2023 Automatic Wordplay Analysis Task 2 – Pun Location and Interpretation

Liana Ermakova[1,*], Tristan Miller[2], Anne-Gwenn Bosser[3], Victor Manuel Palma Preciado[1,4], Grigori Sidorov[4] and Adam Jatowt[5]

[1] *Université de Bretagne Occidentale, HCTI, France*
[2] *Austrian Research Institute for Artificial Intelligence (OFAI), Vienna, Austria*
[3] *École Nationale d'Ingénieurs de Brest, Lab-STICC CNRS UMR 6285, France*
[4] *Instituto Politécnico Nacional (IPN), Centro de Investigacion en Computacion (CIC), Mexico City, Mexico*
[5] *University of Innsbruck, Austria*

Abstract

This paper presents an overview of Task 2 of the JOKER-2023 track on automatic wordplay analysis. The goal of the JOKER track series is to bring together linguists, translators, and computer scientists to foster progress in the automatic interpretation, generation, and translation of wordplay. Task 2 is focussed on pun location and interpretation. Automatic pun interpretation is important for advancing natural language understanding, enabling humor generation, aiding in translation and cross-linguistic understanding, enhancing information retrieval, and contributing to the field of computational creativity. In this overview, we present the general setup of the shared task we organized as part of the CLEF-2023 evaluation campaign, the participants' approaches, and the quantitative results.

Keywords

wordplay, puns, computational humour, wordplay interpretation, wordplay detection, pun location,

1. Introduction

This paper presents details and results of Task 2 of the JOKER-2023 Track on automatic wordplay analysis[1] which was held as part of the 14th Conference and Labs of the Evaluation Forum (CLEF 2023)[2]. JOKER-2023 is the second edition of the JOKER track which was started in 2022 [1]. Task 2 specifically focusses on the identification and interpretation of puns. The other two tasks in the track, on pun detection and translation, are covered in separate papers [2, 3]; an overall overview of the track [4] is also available.

Pun interpretation refers to the process of understanding and deciphering puns. A pun is a form of wordplay, usually humorous, that exploits multiple meanings of a word, or words with similar sounds but different meanings. Automatic pun interpretation plays a role in advancing

CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece
*Corresponding author.
0000-0002-7598-7474 (L. Ermakova); 0000-0002-0749-1100 (T. Miller); 0000-0002-0442-2660 (A. Bosser); 0000-0001-8711-1106 (V. M. Palma Preciado); 0000-0003-3901-3522 (G. Sidorov); 0000-0001-7235-0665 (A. Jatowt)

[1] https://www.joker-project.com
[2] https://clef2023.clef-initiative.eu/

Table 1
Statistics on submitted runs by task

Team	Task 2.1: Location			Task 2.2: Interpret.		Total
	EN	FR	ES	EN	FR	
Croland	1	1	1	1	—	4
Les_miserables	3	3	3	1	—	10
MiCroGerk	6	—	—	4	—	10
Smroltra	4	4	4	6	—	18
TeamCAU	3	—	—	—	—	3
ThePunDetectives	5	—	—	—	—	5
UBO	1	1	1	1	—	4
UBO-RT	—	—	—	1	1	2
AKRaNLU	2	2	2	1	1	8
Total	25	11	11	15	2	64

natural language understanding, enabling humour generation, aiding in translation and cross-linguistic comprehension, enhancing information retrieval, and contributing to the field of computational creativity.

Interpreting a pun involves two steps: recognizing the word or phrase carrying multiple meanings (pun location), and then identifying those meanings (pun interpretation). Pun interpretation systems need to identify potential sources of ambiguity in the context and narrow down the possible interpretations.

Pun comprehension, when done by humans, involves recognizing that there is a pun, and then understanding it, thereby finding humour in the unexpected or clever connection between the different meanings of the words involved. Automatic pun interpretation refers to the use of computational techniques and algorithms to automatically analyze and understand puns without human intervention. Some systems analyze the words involved in the pun, including their meanings and relationships with other words based on lexical resources such as dictionaries, thesauri, or semantic networks such as WordNet [5, 6]. Systems can consider the surrounding context of the pun to gain a better understanding of the intended meaning. This can involve analyzing the broader text or discourse in which the pun appears, including syntactic and semantic features.

Humour is an important aspect of interpersonal interactions and our social behavior. Humour can depend on subjective factors, which makes its automatic processing challenging. Thus, dealing with humour, even in its written form, becomes a rather complex task even if at first sight the problem may seem trivial. Wordplay processing is a specific task in a broader area of automatic humour processing which involves detection, classification, generation, or translation of humour.

The paper discusses two distinct subtasks of JOKER-2023: Task 2.1 on pun location in English, French, and Spanish; and Task 2.2 on pun interpretation in English. Each subtask is presented individually, covering various aspects such as the objectives, data collection process, evaluation metrics, approaches used by the participants, and the corresponding results. In total, nine teams submitted 64 runs in total for Task 2; the breakdown across teams, languages, and subtasks is given in Table 1.

2. Task 2.1: Pun location

2.1. Task description

Pun location (Task 2.1) is a finer-grained version of pun detection. The goal is to identify the words that carry the double meaning in a text which is known *a priori* to contain a pun. The double meaning here produces the humorous effect of wordplay. For example, the first of the following sentences contains a pun where the word *propane* evokes the similar-sounding word *profane*, although the latter is not included in the sentence explicitly, while the second sentence contains a pun exploiting two distinct meanings of the word *interest*:

- (1) When the church bought gas for their annual barbecue, proceeds went from the sacred to the propane.
- (2) I used to be a banker but I lost interest.

Note that for the pun detection task which is the basis of Task 1 of JOKER-2023 Track, the correct answer for these two instances would be "true". Now, for the pun location task, the correct answers are respectively "propane" and "interest". System performance is reported in terms of accuracy for this subtask.

2.2. Data

The pun location data is drawn from the positive examples of JOKER Task 1 [2, 4], with each text being accompanied by an annotation that reproduces the word being punned upon, as described above. These positive examples for the pun detection task were short jokes in English, French, and Spanish with single puns. A detailed description of the English and French pun detection and location datasets can be found in our SIGIR 2023 resource paper [7].

The statistics on data per language are given in Table 2. These statistics present the actual numbers used for the assessment, representing the effective figures for both the test and training data. Nevertheless, when providing files to the participants, we included the training data within the input of the test file. Incorporating the training data within the dataset, for which participants were required to make predictions, enables a comprehensive evaluation of the systems' performance on both the training and testing data. By incorporating the training data, we can assess the systems' ability to generalize to unseen test data by observing their performance on familiar training examples.

The test and training data sets were provided to participants as JSON or delimited text files with fields containing the text of the punning joke and a unique ID. For training data for the pun location task, there is an additional field reproducing the pun word. System output is expected as a JSON or delimited text file with fields for the run ID, text ID, the pun word, and a boolean flag indicating whether the run is manual or automatic.

Table 2
Dataset statistics for Task 2

Language	Train	Test
English	2,315	1,205
French	2,000	4,655
Spanish	876	960

Input format. The base data is provided in JSON and CSV formats with the following fields:

id a unique identifier

text the text of the instance of wordplay

Input example:

```
[{"id":"en_135",
"text":"Cleopatra was the Pharaohs one of all."},


{"id":"en_226",
"text":"At a flower show the first prize is often a bloom ribbon."}
]
```

Qrels. We provide training data as JSON or TSV qrels files with the following fields:

id a unique identifier from the input file

location the portion of the text containing the wordplay

Example of a qrel file:

```
[{"id":"en_135","location":"Pharaohs"},
{"id":"en_226",
"location":"bloom"}
]
```

Output format. Systems were expected to submit their results in a TREC-style JSON or TSV file with the following fields:

run_id run ID starting with <team_id>_<task_id>_<method_used> – e.g., UBO_task_2.1_-TFIDF

manual flag indicating if the run is manual 0,1

id a unique identifier from the input file

location the portion of the text containing the wordplay

Example of an output file:

```
[{"run_id":"team1_task_2.1_TFIDF",
"manual":0,
"id":"en_135",
"location":"Pharaohs"},

{"run_id":"team1_task_2.1_TFIDF",
"manual":0,
"id":"en_226",
"location":"bloom"}]
```

2.3. Participants' approaches

Eight teams participated in Task 2.1:

1. The AKRaNLU team participants [5] employ the token classification method with a tagging schema that relies on assigning a tag of 1 to every pun word and 0 to every word that is not a punning word.
2. The MiCroGerk team [8] chose an large language model (LLM) approach for Task 2.1, using T5 (SimpleT5), BLOOM, and models from OpenAI and AI21. They also submitted a baseline that uses last word in the sentence as a prediction, as well as a random baseline. It is noteworthy that the BLOOM model presented the worst results compared to the others.
3. The Smroltra team [6] observed that models based on GPT-3, SpaCy, T5, and BLOOM showed very good performance when it came to Spanish and English, while for French the results were worse. This was particularly the case for SpaCy, which is believed to be not as developed for French as for English.
4. TeamCAU [9] used various LLMs. T5 showed good results in comparison to BLOOM and models from AI21 (albeit for partial runs only).
5. FastText, Ridge, Naive Bayes, SimpleT5, and SimpleTransformersT5 were used by the participants of ThePunDetectives team [10]. They found the best results to be produced by the pre-trained models. In particular, T5 achieved good performance, as predicted by the authors.
6. For the location tasks, the UBO team [11] opted to use T5 (SimpleT5).
7. The Croland team [12] used GPT-3.
8. The Les_miserables team (who did not submit a system description paper) submitted two baseline runs, one where the system selects the final word of the sentence as the pun location, and another run that randomly predicts words; they also submitted a run using the T5 (SimpleT5) model.

2.4. Results

The eight teams submitted 47 runs in total for all three languages. Table 3 reports the participants' results for wordplay location tasks in English, French, and Spanish on the test set. This table provides a comprehensive overview of the participants' systems' performances and their respective scores or metrics achieved in locating wordplay instances in the given languages. As some participants submitted only partial runs, we provide two sets of accuracy scores: those labeled A are based on the total number of instances in the test set, while those labeled A* are calculated only using the actual number of attempted instances (#).

Accuracy scores for pun location in English and Spanish (A $\approx$ 80) are roughly twice as good as those for French (A $\approx$ 40). For comparison, the predictions made on the last word for test sets in English and Spanish are around 50% while for French this score goes almost up to 30%. Thus, the improvement for French over a very simple baseline is rather not high. The significant improvement over this simple last-word baseline for English and Spanish could be explained by the fact that participants used large language models (e.g., GPT-3 or BLOOM) that might have included in their training data some of the same puns found in our corpus. By contrast, the French wordplay data was largely constructed by us and not previously published online.

Table 4 shows the results of the participants for wordplay location in English, French, and Spanish on the training set. We observe a significant difference between the best performance on the test and training data for French obtained by T5 model trained on our data as well as the results of the best-performing team AKRaNLU for French. These results suggest overfitting issues. The performance of a few models is comparable to or even lower than the predictions made by returning the last word in the text.

Several teams submitted runs by applying the same methods yet implemented, trained, or fine-tuned differently. Significant disparities can be observed, even in the last-word baseline outcomes (such as Les_miserables_word and MiCroGerk_lastWord), which could be attributed to variations in tokenization methods. Distinguishing variations in prediction accuracy are also noticeable between differently trained T5 models as well as prompt-based language models.

3. Task 2.2: Pun interpretation

3.1. Task description

In this task, systems must describe the semantics –i.e., the two meanings – of the pun. In JOKER-2023, these semantic annotations are in the form of a pair of lemmatized word sets. Following the practice used in lexical substitution datasets, these word sets contain the synonyms (or if absent, then hypernyms) of the two words involved in the pun, except for any synonyms/hypernyms that happen to share the same spelling with the pun as written.

For example, for the punning joke introduced in Example 1 above, the word sets are {*gas, fuel*} and {*profane*}, and for Example 2, the word sets are {*involvement*} and {*fixed charge*, *fixed cost*, *fixed costs*}.

Results for Task 2.2 are scored as the average score for each of punning word senses. Systems need to guess only one word for each sense of the pun; a guess is considered correct if it matches any of the words in the gold-standard set. For example, a system guessing {*fuel*}, {*profane*} would

Table 3
Results for Task 2.1 (pun location) on the test data

	EN			FR			ES		
run ID	#	A	A*	#	A	A*	#	A	A*
Croland_GPT3	19	0.41	26.31	61	0.20	18.03	51	1.77	33.33
Les_miserables_random	1205	8.87	8.87	4655	4.37	4.98	960	6.14	6.14
Les_miserables_simplet5	1205	76.18	76.18	4655	39.92	45.49	960	55.41	55.41
Les_miserables_word	1205	49.54	49.54	4655	28.67	32.67	960	51.56	51.56
Smroltra_BLOOM	32	1.74	65.62	65	0.41	33.84	57	2.60	43.85
Smroltra_GPT3	32	2.15	81.25	65	0.56	46.15	57	5.20	87.71
Smroltra_SimpleT5	1205	79.50	79.50	4655	39.86	45.43	960	**82.81**	82.81
Smroltra_SpaCy	1205	44.48	44.48	4655	0.00	0.00	960	24.16	24.16
UBO_SimpleT5	1205	77.67	77.67	4655	40.39	46.03	960	57.70	57.70
AKRaNLU_tokenclassification_x	1205	77.51	77.51	4655	40.56	46.22	960	54.27	54.27
AKRaNLU_tokenclassification_y	1205	79.17	79.17	4655	**41.35**	47.13	960	56.14	56.14
TeamCAU_AI21	32	1.16	43.75	—	—	—	—	—	—
TeamCAU_BLOOM	32	1.24	46.87	—	—	—	—	—	—
TeamCAU_ST5	1205	80.66	80.66	—	—	—	—	—	—
ThePunDetectives_Fasttext	1205	5.06	5.06	—	—	—	—	—	—
ThePunDetectives_NaiveBayes	1205	2.07	2.07	—	—	—	—	—	—
ThePunDetectives_Ridge	1205	50.20	50.20	—	—	—	—	—	—
ThePunDetectives_SimpleT5	1205	80.41	80.41	—	—	—	—	—	—
ThePunDetectives_SimpleTransformersT5	1205	**83.15**	83.15	—	—	—	—	—	—
MiCroGerk_AI21	17	1.32	94.11	—	—	—	—	—	—
MiCroGerk_BLOOM	17	0.99	70.58	—	—	—	—	—	—
MiCroGerk_OpenAI	17	1.24	88.23	—	—	—	—	—	—
MiCroGerk_SimpleT5	1205	79.91	79.91	—	—	—	—	—	—
MiCroGerk_lastWord	1205	54.43	54.43	—	—	—	—	—	—
MiCroGerk_random	1205	13.94	13.94	—	—	—	—	—	—

receive a score of 1 for Example 1, and a system guessing {*fuel*}, {*prophet*} would receive a score of 1/2.

3.2. Data

For the English pun interpretation data, we manually annotated each pun according to its senses in WordNet 3.1 and then automatically extracted the synonyms (or if there were none, the hypernyms) of those words to form the two word sets. In some cases, one or both of the senses of the pun was not present in WordNet, or WordNet contained neither synonyms nor hypernyms for the annotated senses. (This was particularly the case with adjectives and adverbs, which WordNet does not arrange into a hypernymic hierarchy.) In these cases, we sourced the synonym/hypernym sets from human annotators. For the French data, we used a simplified

Table 4
Results for Task 2.1 (pun location) on the training data

	EN			FR			ES		
run ID	#	A	A*	#	A	A*	#	A	A*
Croland_task_1.1/2_EN_GPT3	71	0.52	16.90	29	0.30	20.69	39	1.03	23.08
Les_miserables_random	2315	8.25	8.25	2000	5.39	5.40	876	6.74	6.74
Les_miserables_simplet5	2315	85.05	85.05	2000	63.22	63.25	876	69.06	69.06
Les_miserables_word	2315	46.39	46.39	2000	33.83	33.85	876	51.48	51.48
Smroltra_task_1.2_EN_BLOOM	68	1.68	57.35	35	0.55	31.43	43	2.40	48.84
Smroltra_task_1.2_EN_GPT3	68	2.20	75.00	35	0.90	51.43	43	4.79	97.67
Smroltra_task_1.2_EN_SimpleT5	2315	84.58	84.58	2000	64.27	64.30	876	**84.59**	**84.59**
Smroltra_task_1.2_EN_SpaCy	2315	90.54	**90.54**	2000	57.37	57.40	876	22.26	22.26
UBO_task_2.1_SimpleT5	2315	87.60	87.60	2000	**77.76**	**77.80**	876	72.60	72.60
AKRaNLU_task_2.1_tokenclassification_x	2315	82.33	82.33	2000	55.17	55.20	876	67.81	67.81
AKRaNLU_task_2.1_tokenclassification_y	2315	85.75	85.75	2000	59.12	59.15	876	71.12	71.12
TeamCAU_task_1.2_EN_AI21	68	1.25	42.64	—	—	—	—	—	—
TeamCAU_task_1.2_EN_BLOOM	68	1.07	36.76	—	—	—	—	—	—
TeamCAU_task_1.2_EN_ST5	2315	85.44	85.44	—	—	—	—	—	—
ThePunDetectives_task_1.1.2_Fasttext	2315	11.79	11.79	—	—	—	—	—	—
ThePunDetectives_task_1.1.2_NaiveBayes	2315	5.18	5.18	—	—	—	—	—	—
ThePunDetectives_task_1.1.2_Ridge	2315	**90.67**	90.66	—	—	—	—	—	—
ThePunDetectives_task_1.1.2_SimpleT5	2315	85.26	85.26	—	—	—	—	—	—
ThePunDetectives_task_1.1.2_SimpleTransformersT5	2315	87.47	87.47	—	—	—	—	—	—
MiCroGerk_task_1.2_EN_AI21	33	0.90	63.63	—	—	—	—	—	—
MiCroGerk_task_1.2_EN_BLOOM	33	0.56	39.39	—	—	—	—	—	—
MiCroGerk_task_1.2_EN_OpenAI	33	1.12	78.78	—	—	—	—	—	—
MiCroGerk_task_1.2_EN_SimpleT5	2315	86.86	86.86	—	—	—	—	—	—
MiCroGerk_task_1.2_EN_lastWord	2315	57.01	57.01	—	—	—	—	—	—
MiCroGerk_task_1.2_EN_random	2315	14.12	14.12	—	—	—	—	—	—
MiCroGerk_OpenAI	17	1.24	88.23	—	—	—	—	—	—
MiCroGerk_SimpleT5	1205	79.91	79.91	—	—	—	—	—	—
MiCroGerk_lastWord	1205	54.43	54.43	—	—	—	—	—	—
MiCroGerk_random	1205	13.94	13.94	—	—	—	—	—	—

version of the annotation made in JOKER-2022 [1].

Input format The base data for test and training are provided in JSON and CSV formats with the following fields:

id a unique identifier

text the text of the instance of wordplay

Input example:

```
[{"id":"en_135",
"text":"Cleopatra was the Pharaohs one of all."},

{"id":"en_226",
"text":"At a flower show the first prize is often a bloom ribbon."}
]
```

Qrels We provide training data in the format of JSON or TSV qrels files with the following fields:

id a unique identifier from the input file

location the portion of the text containing the wordplay

interpretation synonyms or hypernyms of the two meanings of the wordplay

Example of a qrel file:

```
[{"id":"en_135",
"location":"Pharaohs",
"interpretation":"Pharaoh;Pharaoh of Egypt / fair"},


{"id":"en_226",
"location":"bloom",
"interpretation":"blossom;flower / bluish;blue;blueish"}
]
```

Output format Systems were expected to provide their results as a TREC-style JSON or TSV format with the following fields:

run_id run ID starting with <team_id>_<task_id>_<method_used> – e.g., UBO_task_2.2_BLOOM

manual flag indicating if the run is manual 0,1

id a unique identifier from the input file

location the portion of the text containing the wordplay

interpretation synonyms or hypernyms of the wordplay meanings

Example of an output file:

```
[{"run_id":"team1_task_2.2_manual",
"manual":1,
"id":"en_135",
"location":"Pharaohs",
"interpretation":"Pharaoh;Pharaoh of Egypt / fair"},
```

```
{"run_id":"team1_task_2.2_manual",
"manual":1,
"id":"en_226",
"location":"bloom",
"interpretation":"blossom;flower / bluish;blue;blueish"}]
```

3.3. Participants' approaches

The teams' approaches were as follows:

1. For pun interpretation, the AKRaNLU team participants [5] used the results from the pun location subtask to disambiguate the appropriate senses of the pun word based on the sentence content and find two synonyms for those senses, sourced from WordNet, that were most similar to sentence embedding.
2. The MiCroGerk team [8] submitted four runs for the interpretation task based on LLMs, such as T5 (SimpleT5), BLOOM, and models from OpenAI and AI21.
3. The Smroltra team [6] submitted six runs based on GPT-3 and BLOOM, SpaCy, T5 and their combinations with WordNet for location prediction.
4. The UBO team [11] applied T5 (SimpleT5) to predict the interpretation of puns in English.
5. The UBO-RT team [13] post-edited output of ChatGPT (C&O in Tables 5 and 6). A zero-shot strategy was used in their approach and the analysis of the results reveals quite poor capabilities of ChatGPT in interpreting puns, especially those involving homophonic components.
6. The Croland team [12] used GPT-3.
7. The Les_miserables team (who did not submit a system description paper) submitted a run using the T5 (SimpleT5) model to predict pun interpretation in English.

3.4. Results

We show in Table 5 the results of the pun interpretation task for English. We do not report the results for French as only two teams submitted runs for this language, one of which was heavily post-processed manually. This means it is impossible to draw any conclusions on the French dataset.

For the English pun interpretation task, seven teams submitted 15 runs in total. As we expected, the majority of participants opted to use LLMs, resulting in the generation of partial runs due to the efficiency constraints associated with these models. Only five runs out of the total number of submissions involved the entire testing data (1,192), hence the comparison

Table 5
Results for Task 2.2 (pun interpretation) on the test data

run	count	score	part_score
C&O_task_2.2_Chat GPT	92	5.45%	**70.65%**
Croland_task_2_EN_GPT3	29	0.08%	3.45%
Les_miserables_simplet5	1,192	**47.40%**	47.40%
MiCroGerk_task_2_EN_AI21	11	0.46%	50.00%
MiCroGerk_task_2_EN_BLOOM	2	0.04%	25.00%
MiCroGerk_task_2_EN_OpenAI	11	0.34%	36.36%
MiCroGerk_task_2_EN_SimpleT5	39	1.59%	48.72%
Smroltra_task_2_Bloom	32	0.59%	21.88%
Smroltra_task_2_GPT3	32	0.59%	21.88%
Smroltra_task_2_GPT3_WN	32	1.09%	40.63%
Smroltra_task_2_SimpleT5_WN	1192	41.44%	41.44%
Smroltra_task_2_bloom_WN	32	0.80%	29.69%
Smroltra_task_2_spaCy_WN	1,192	19.76%	19.76%
UBO_task_2.2_SimpleT5	1,192	46.85%	46.85%
akranlu_task_2.2_sentembwordnet	1,192	39.77%	39.77%

is somewhat difficult. Nevertheless, when focussing on the full runs, we observe that the maximum accuracy was obtained by the team Les_miserables who used the T5 model, achieving 47.4%. A similar result was also obtained by the UBO team who also applied the T5 model. For comparison, a baseline approach with SpaCy gives 19.76% accuracy. This underscores the utility of large language models for the interpretation of wordplay. The best partial score was obtained by the UBO-RT team who used the post-processed results generated by ChatGPT. But even this heavily manually post-processed run obtained only 70%.

Additionally, for completeness, Table 5 provides results of participating systems for the training data. Performance on the training data remains low, which may suggest the overall difficulty of this task. On the other hand, results of some models such us T5 and SpaCy exhibited notably superior performance on the training set, which suggests the possibility of overfitting.

4. Conclusion

In this paper, we have described Task 2 of the JOKER track at CLEF 2023, consisting of pun location and interpretation challenges. We extended our previously described dataset [7] by introducing semantic annotation for wordplay in English and French. Furthermore, we constructed a corpus for pun location in Spanish.

Multiple teams submitted runs using similar methods, but with variations in implementation, training, or fine-tuning approaches for both subtasks. These variations entailed large differences in the performance of the systems.

Our results in general suggest that wordplay location is still a challenge for LLMs despite their recent significant advances. Interestingly, we found that the results for the French language when it comes to pun location are quite low (half of the scores of English and Spanish) which

Table 6
Results for Task 2.2 (pun interpretation) on the training data

run	count	score	part_score
C&O_task_2.2_Chat GPT	201	6.57%	75.62%
Croland_task_2_EN_GPT3	61	0.22%	8.20%
Les_miserables_simplet5	2,315	52.66%	52.66%
MiCroGerk_task_2_EN_AI21	30	0.76%	58.33%
MiCroGerk_task_2_EN_BLOOM	7	0.19%	64.29%
MiCroGerk_task_2_EN_OpenAI	30	0.35%	26.67%
MiCroGerk_task_2_EN_SimpleT5	119	4.34%	**84.45%**
Smroltra_task_2_Bloom	68	0.82%	27.94%
Smroltra_task_2_GPT3	68	0.30%	10.29%
Smroltra_task_2_GPT3_WN	68	1.34%	45.59%
Smroltra_task_2_SimpleT5_WN	2,315	53.50%	53.50%
Smroltra_task_2_bloom_WN	68	1.06%	36.03%
Smroltra_task_2_spaCy_WN	2,315	52.74%	52.74%
UBO_task_2.2_SimpleT5	2,315	**67.15%**	67.15%
akranlu_task_2.2_sentembwordnet	2,315	48.10%	48.10%

we attribute to different data creation procedures. Some of the puns found in our corpus for English and Spanish might have been “known” by large language models that were used by participants, as the data were sourced from the web for these languages. On the other hand, French data was novel and largely constructed by us. This calls for the future to construct wordplay datasets from scratch rather than sourcing humour from external sources like the web.

The results suggest the overfitting problem for models trained on our data (e.g., via the SimpleT5 library). The difference between training and test data observed for the prompt-based models is small as they are not actually trained on our data, but might be influenced by examples from the training set used in the prompts.

Automatic pun interpretation was the second subtask of Task 2. It is quite a challenging task due to the inherent ambiguity and creativity of puns, yet it fits into the recent focus on explainable AI and explainable/interpretable decision systems. Puns often rely on cultural knowledge, background information, and linguistic subtleties that can be difficult to capture computationally. Still, researchers continue to explore various approaches, including rule-based methods, machine learning models, and deep learning techniques to improve automatic pun interpretation systems. Our results data indicate that the T5 model performs well for this task. However, we could compare the results only for English language, as for French there were only two runs that were not fully automatic. We received many partial runs due to token/time constraints of LLMs and, therefore, apart from effectiveness, the efficiency of the approaches should be considered in future research. The results suggest the difficulty of pun interpretations at least in the particular settings that we use in subtask 2.2 (reliance on WordNet synonyms and hypernyms).

Additional information on the track is available on the JOKER website: http://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. JOKER is supported by *La Maison des sciences de l'homme en Bretagne*. For their help and support in the first Spanish pun translation contest, we thank Carolina Palma Preciado, Leopoldo Jesús Gutierrez Galeano, Khatima El Krirh, Nathalie Narváez Bruneau, and Rachel Kinlay. We also thank all other colleagues and students who participated in data construction, the translation contests, and the CLEF JOKER track.

References

[1] 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.

[2] L. Ermakova, T. Miller, A.-G. Bosser, V. M. P. Preciado, G. Sidorov, A. Jatowt, Overview of JOKER 2023 automatic wordplay analysis task 1 - pun detection, in: [14], 2023.

[3] L. Ermakova, T. Miller, A.-G. Bosser, V. M. P. Preciado, G. Sidorov, A. Jatowt, Overview of JOKER 2023 Automatic Wordplay Analysis Task 3 - Pun translation, in: [14], 2023.

[4] 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.

[5] R. R. Dsilva, AKRaNLU @ CLEF JOKER 2023: Using sentence embeddings and multilingual models to detect and interpret wordplay, in: [14], 2023.

[6] O. Popova, P. Dadić, Does AI have a sense of humor? CLEF 2023 JOKER tasks 1, 2 and 3: Using BLOOM, GPT, SimpleT5, and more for pun detection, location, interpretation and translation, in: [14], 2023.

[7] 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.

[8] A. Prnjak, D. R. Davari, K. Schmitt, CLEF 2023 JOKER Task 1, 2, 3: pun detection, pun interpretation, and pun translation, in: [14], 2023.

[9] A. Anjum, N. Lieberum, Exploring Humor in Natural Language Processing: A Comprehensive Review of JOKER Tasks at CLEF Symposium 2023, in: [14], 2023.

[10] F. Ohnesorge, M. Á. Gutiérrez, J. Plichta, CLEF 2023 JOKER Tasks 2 and 3: using NLP models for pun location, interpretation and translation, in: [14], 2023.

[11] Q. Dubreuil, UBO Team @ CLEF JOKER 2023 track for Task 1, 2 and 3 - applying AI models in regards to pun translation, in: [14], 2023.

[12] J. Komorowska, I. Čatipović, D. Vujica, CLEF2023' JOKER Working Notes, in: [14], 2023.

[13] O. Brunelière, C. Germann, K. Salina, CLEF 2023 JOKER Task 2: using Chat GPT for pun location and interpretation, in: [14], 2023.

[14] Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2023.