Overview of JOKER – CLEF-2023 Track on Automatic Wordplay Analysis

Liana Ermakova[1[0000−0002−7598−7474]], Tristan Miller[2[0000−0002−0749−1100]], Anne-Gwenn Bosser[3[0000−0002−0442−2660]], Victor Manuel Palma Preciado[1,4[0000−0001−8711−1106]], Grigori Sidorov[4[0000−0003−3901−3522]], and Adam Jatowt[5[0000−0001−7235−0665]]

[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 Investigación en Computación (CIC), Mexico City, Mexico
[5] University of Innsbruck, Austria

Abstract The goal of the JOKER track series is to bring together linguists, translators, and computer scientists to foster progress on the automatic interpretation, generation, and translation of wordplay. Being clearly important for various applications, these tasks are still extremely challenging despite significant recent progress in AI in information retrieval and natural language processing. Building on the lessons learned from last year's edition, JOKER-2023 held three shared tasks aligned with human approaches to the translation of wordplay, or more specifically of puns in English, French, and Spanish: detection, location and interpretation, and finally translation. In this paper, we define these three tasks and describe our approaches to corpus creation and evaluation. We then present an overview of the participating systems, including the summaries of their approaches and a comparison of their performance. As in JOKER-2022, this year's track also solicited contributions making further use of our data (an "unshared task"), which we also report on.

Keywords: Wordplay · Puns · Humour · Wordplay interpretation · Wordplay detection · Wordplay generation · Machine translation.

1 Introduction

Intercultural communication relies heavily on translation. It is therefore vitally important that semantics-oriented language technology be capable of detecting, interpreting, and appropriately dealing with non-literal expressions such as wordplay. However, wordplay remains one of the most elusive aspects of translation, as it requires an attuned understanding of implicit cultural knowledge, and a keen grasp of language form to understand how to bend it to the desired effect. Furthermore, wordplay appears in all languages and is present in most discourse types. It is used by novelists, poets, playwrights, scriptwriters, and copywriters.

It is often employed in titles, headlines, or slogans for its salience and its playful or subversive character. But while modern translation is heavily aided by technological tools, there is little support for humour and wordplay, and even the most current language models struggle to imitate human humour [17].

If the objective of an AI-based translation tool able to deal with wordplay is to be attained, we will almost certainly need to rely on a multilingual parallel corpus: such a tool would necessarily require training on a sizeable quantity of data. This is essentially what the JOKER track at the Conference and Labs of the Evaluation Forum (CLEF) provides, together with tasks designed to establish and advance the current state of the art for wordplay processing.

While humour and wordplay are widely studied in the humanities and social sciences, they have been largely ignored in information retrieval, including dedicated neural net-based retrieval methods and large language models [9]. This is partly because modern AI tools tend to require quality and quantity of training data that has historically been lacking for humour and wordplay. Wordplay detection is useful for information retrieval, digital humanities, conversational agents, and other humour-aware text processing applications. Wordplay location is a prerequisite for the retrieval of jokes containing a specified punning word.

Building on insights gained at the 2022 edition of the JOKER lab [12], we have organized four shared tasks based on our newly expanded, multilingual, parallel corpus of wordplay in English, French, and (new in this year's edition) Spanish [10]:

Task 1 Pun detection in English, French, and Spanish.
Task 2 Pun location and interpretation in English, French, and Spanish.
Task 3 Pun translation from English to French and from English to Spanish.
Open Task We encouraged the use of our data for other tasks related to computational wordplay and humour. These could take the form of, for example, experiments on humour perception, humour evaluation, wordplay generation, or user studies.

Fifty teams registered for our JOKER track at CLEF 2023; of these, thirteen teams participated in the tasks, submitting a total of 186 runs for the numbered tasks. The statistics for these runs are presented in Table 1. In addition, we received three submissions for the open task, covering various areas: an attempt at automated sentiment analysis on the corpus, a pipeline for pun generation in English, and a user study evaluating how well non-native English speakers of varying proficiency levels and countries of origin did on the shared tasks that we had aimed at machines.

2 Task 1: Pun detection in English, French, and Spanish

A *pun* is a form of wordplay in which a word or phrase evokes the meaning of another word or phrase with a similar or identical pronunciation [18]. *Pun detection* is a binary classification task where the goal is to distinguish between texts containing a pun (the *positive examples*) and texts not containing any pun

Table 1. Statistics on submitted runs by task

Team	Task 1: Detection			Task 2.1: Location			Task 2.2: Interpret.		Task 3: Translation		Total
	EN	FR	ES	EN	FR	ES	EN	FR	EN→FR	EN→ES	
Croland	1	1	1	1	1	1	1		1	1	9
LJGG	3	3	3						4	5	18
Les_miserables	3	3	3	3	3	3	1				19
MiCroGerk	6			6			4			7	23
Smroltra	7	7	7	4	4	4	6		6	6	51
TeamCAU	6			3					3		12
TheLangVerse	1								1	1	3
ThePunDetectives	6			5					2	2	15
UBO	1	1	1	1	1	1	1		3	3	13
UBO-RT							1	1			2
AKRaNLU	2	2	2	2	2	2	1	1			14
Innsbruck	3										3
NPalma	1		1							2	4
Total	40	17	18	25	11	11	15	2	20	27	186

(the *negative examples*) [22]. Performance on this task is evaluated using the standard precision, recall, accuracy, and F-score metrics from text classification and information retrieval [21, Ch. 8.3].

2.1 Data

Most of the English- and French-language data used for our tasks is described in detail in a resource paper published at SIGIR 2023 [9]. Below we briefly describe the overall data collection process and then discuss in detail the way in which the Spanish-language data was created. For Task 1, the relevant portions of these subcorpora consist of positive and negative texts that are not otherwise annotated or marked up in any way. The positive examples are all short jokes (one-liners), each containing a single pun. In contrast to previously published punning datasets, our negative examples are generated by the data augmentation techniques of manually or semi-automatically editing positive examples in such a way that the wordplay is lost but most of the rest of the meaning still remains. More specifically, in each positive text we made some minimal edits – generally substituting a single word, which may or may not have been the word forming the pun. We adopted this approach in order to minimize the differences in length, vocabulary, style, etc. that manifested across the positive and negative subsets of previous pun detection datasets and on which classifiers could rely on, inadvertently or otherwise, to distinguish those subsets. For the French subcorpus, additional negative examples were sourced through machine translation of the English positive examples, a process through which the wordplay is almost always lost.

The Spanish data was collected primarily via two methods. The first of these was to scrape a manually seeded set of web pages known to collect jokes, and then to manually filter out non-puns and other inappropriate texts. Our data source was Twitter, for which we used Twarc[6] to extract some 195K tweets with the hashtags `#humor`, `#juegodepalabras`, and `#chiste` (meaning "humour", "pun", and "joke", respectively). Here, too, we manually filtered out non-punning examples or those containing extraneous information (images, URLs, emoticons, extra hashtags, etc.). All told, we were able to collect about a thousand examples, about a quarter of which were from web pages and the remainder from Twitter. Negative examples were then generated using essentially the same data augmentation technique used for the English and French data.

The data for each language was split into test and training sets and provided to Task 1 participants in simple JSON and delimited text formats with fields giving a unique ID, the text to classify, and (for training data) a boolean value indicating whether or not the text contains a pun. Participants could choose which language(s) they wished to submit classification runs for. The expected output format was a similarly simple, delimited text file with fields for the run ID, the text ID, the boolean classification result, and a boolean flag indicating whether the classification was made manually or automatically.

Table 2 provides statistics on the size of the dataset, broken down by language and task. Statistics specific to Task 1 are presented in Table 3.

Table 2. Overall dataset statistics

	Task 1		Task 2		Task 3			
Language	Train	Test	Train	Test	Train		Test	
					target	source	target	source
English	5,292	3,183	2,315	1,205	—	—	—	—
French	3,999	12,873	2,000	4,655	5,838	1,405	6,590	1,197
Spanish	1,994	2,241	876	960	644	217	5,727	544

Table 3. Task 1 data statistics

	Train			Test		
Language	Positive	Negative	Total	Positive	Negative	Total
English	3,085	2,207	5,292	809	2,374	3,183
French	1,998	2,001	3,999	5,308	7,565	12,873
Spanish	855	1,139	1,994	952	1,289	2,241

[6] `https://github.com/DocNow/twarc/`

2.2 Participants' approaches

The AKRaNLU team [7] described two methods for pun detection which are based on sentence embeddings with a binary classifier and a sequence classification using XLM-Roberta. A six-layer neural network with a classifier head for the three languages was used.

The NLPalma team [26] experimented with models based on the multilingual BERT architecture. The authors concluded that this approach is promising, but indicated that more fine-tuning of models should lead to better results.

The MiCroGerk participants [27] used six different runs to classify sentences, with systems based on FastText, T5 (based on SimpleT5 library), BLOOM alone, MLP (multilayer perceptron), Naive Bayes and Ridge along with the TF–IDF vectorizer and Count vectorizer. T5 obtained the highest score compared to the other methods.

TheLangverse team [15] used a combination of FastText and an MLP as a classifier layer, achieving somewhat good results compared to what they found in their surveys.

ThePunDetectives team [23] used several models, including Random, FastText, Ridge, Naive Bayes, T5 (SimpleTransformersT5), and RoBERTa (SimpleTransformersRoBERTa) for the classification task, with RoBERTa (barely) achieving the best results among their models.

The participants from the UBO team [8] used T5 (SimpleT5) to solve Task 1, achieving mixed results across languages, with French having the highest success rate.

TeamCAU [2] used different models including Large Language Models (LLMs), FastText, and TF–IDF. In the case of LLMs, they used BLOOM, Jurassic-2 through AI21's inference API. and T5 (SimpleT5). The authors reported that, among their runs, they obtained the best results using LLMs.

The Smroltra team [25] experimented extensively with different classification methods: Random Forest, FastText, Naive Bayes, Logistic Regresion, TF–IDF, MLP, and finally a T5 (SimpleT5) transformer which is already commonly used in the tasks concerning humour. They obtained quite similar results for the Spanish and French datasets. Their approaches were not as reliable for detecting English puns, despite the fact that most of the methods tend to be used mainly for English.

The Croland team [19] tackled Task 1 using OpenAI's GPT-3, under the assumption that LLMs should possess a good understanding of humour.

The Innsbruck team [28] experimented with different data augmentation (DA) techniques, including synonym replacement, back-translation, shortening, in order to improve humour recognition.

The LJGG team [16] experimented with different ways of training a T5 (SimpleT5) model.

Finally, the Les_miserables team (who did not submit a system description paper) submitted SimpleT5- and FastText-based predictions, as well as random baseline results.

2.3 Results

Tables 4, 5, and 6 report participants' results for wordplay detection in English, French, and Spanish, respectively. As some participants submitted only partial runs, we provide separate precision, recall, F_1, and accuracy scores for the total number of instances in the test set (P, R, F_1, A) and for only the number of instances (#) where a classification was attempted (P^*, R^*, F_1^*, A^*). Our results suggest that wordplay detection is still challenging for all models and all three languages. The improvement of the best runs over the random results are less than 15 points according to F_1 score for all three languages.

The best results according to the F_1 metric for English are achieved with T5 model by two teams: LJGG and UBO. Still the results are lower than 60%. We also observe that the results of the same methods depend heavily on implementation, fine-tuning, and/or used prompts.

For French, the best results were also achieved by the teams applying T5 with F_1 going up to 66.45. This improvement over English might be explained by higher similarity between train and test data in French as this data is coming from the translation of the overlapping sets English puns, while the test set in English contains different puns without semantic or vocabulary similarity. Surprisingly, Logistic Regression and TF–IDF classifier demonstrated comparable results. These results might suggest that efficient training of lighter models could help to achieve results comparable to ones from large pre-trained models which are very expensive and resource-consuming.

For Spanish, the best results were achieved again by T5 and the AKRaNLU team who applied sentence embeddings with a binary classifier and a sequence classification using XLM-Roberta ($F_1 = 59.64$). Note that Smroltra's random prediction obtained $F_1 = 51.92$ on the same data.

3 Task 2: Pun location and interpretation in English, French, and Spanish

Pun location (Task 2.1) is a finer-grained version of pun detection, where the goal is to identify which words carry the double meaning in a text known *a priori* to contain a pun. For example, the first of the following sentences contains a pun where the word *propane* evokes the similar-sounding word *profane*, and 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.

While for the pun detection task, the correct answer for these two instances would be "true", for the pun location task, the correct answers are respectively "propane" and "interest". System performance is reported in terms of accuracy.

Table 4. Results for Task 1 (pun detection) in English

run ID	#	P	R	F_1	A	P^*	R^*	F_1^*	A^*
Croland_EN_GPT3	3183	100.00	0.86	1.71	74.80	100.00	0.86	1.71	74.80
LJGG_t5_large_easy_en	3183	42.73	71.94	**53.61**	68.36	42.73	71.94	53.61	68.36
LJGG_t5_large_label_en	3183	25.41	100.00	40.53	25.41	25.41	100.00	40.53	25.41
LJGG_t5_large_no_label_en	3183	25.41	100.00	40.53	25.41	25.41	100.00	40.53	25.41
Les_miserables_fasttext	3183	25.78	80.96	39.11	35.94	25.78	80.96	39.11	35.94
Les_miserables_random	3183	26.43	51.29	34.88	51.33	26.43	51.29	34.88	51.33
Les_miserables_simplet5	3183	28.13	88.75	42.72	39.52	28.13	88.75	42.72	39.52
MiCroGerk_EN_BLOOM	13.00	8.33	0.12	0.24	74.26	8.33	100.00	15.38	15.38
MiCroGerk_EN_FastText	3183	25.87	82.94	39.44	35.28	25.87	82.94	39.44	35.28
MiCroGerk_EN_MLP	3183	29.04	72.92	41.54	47.84	29.04	72.92	41.54	47.84
MiCroGerk_EN_NB	3183	25.98	95.42	40.84	29.75	25.98	95.42	40.84	29.75
MiCroGerk_EN_Ridge	3183	26.74	85.16	40.70	36.94	26.74	85.16	40.70	36.94
MiCroGerk_EN_SimpleT5	3183	30.75	83.06	44.88	48.16	30.75	83.06	44.88	48.16
Smroltra_EN_FastText	3183	25.62	80.34	38.85	35.72	25.62	80.34	38.85	35.72
Smroltra_EN_Logistic-Regression	3183	26.14	86.15	40.11	34.62	26.14	86.15	40.11	34.62
Smroltra_EN_MLP	3183	27.78	72.43	40.16	45.14	27.78	72.43	40.16	45.14
Smroltra_EN_NBC	3183	26.12	95.55	41.02	30.19	26.12	95.55	41.02	30.19
Smroltra_EN_Random	3183	25.54	66.99	36.98	41.97	25.54	66.99	36.98	41.97
Smroltra_EN_SimpleT5	3183	31.97	83.68	46.27	50.61	31.97	83.68	46.27	50.61
Smroltra_EN_TFIDF	3183	26.90	84.05	40.76	37.92	26.90	84.05	40.76	37.92
TeamCAU_EN_AI21	40	27.58	0.98	1.90	74.17	27.58	80.00	41.02	0.43
TeamCAU_EN_BLOOM	40	30.00	0.37	0.73	74.45	30.00	30.00	30.00	65.00
TeamCAU_EN_FastText	3183	25.71	80.84	39.02	35.78	25.71	80.84	39.02	35.78
TeamCAU_EN_RandomForestWithTFidfEncoding	3183	25.69	83.43	39.28	34.46	25.69	83.43	39.28	34.46
TeamCAU_EN_ST5	3183	26.99	93.32	41.87	34.15	26.99	93.32	41.87	34.15
TeamCAU_EN_TFidfRidge	3183	26.74	85.16	40.70	36.94	26.74	85.16	40.70	36.94
TheLangVerse_fasttext-MLP	3183	26.31	75.40	39.01	40.08	26.31	75.40	39.01	40.08
ThePunDetectives_Fasttext	3183	26.07	80.22	39.35	37.16	26.07	80.22	39.35	37.16
ThePunDetectives_Naive-Bayes	3183	25.43	99.62	40.52	25.66	25.43	99.62	40.52	25.66
ThePunDetectives_Random	3183	25.96	50.55	34.31	50.80	25.96	50.55	34.31	50.80
ThePunDetectives_Ridge	3183	27.44	88.75	41.92	37.51	27.44	88.75	41.92	37.51
ThePunDetectives_Roberta	3183	26.11	91.96	40.67	31.82	26.11	91.96	40.67	31.82
ThePunDetectives_SimpleT5	3183	29.21	93.20	44.48	40.87	29.21	93.20	44.48	40.87
UBO_SimpleT5	3183	36.51	85.53	51.18	58.52	36.51	85.53	51.18	58.52
AKRaNLU_sentemb	3183	26.29	86.40	40.32	34.99	26.29	86.40	40.32	34.99
AKRaNLU_seqclassification	3183	25.41	100.00	40.53	25.41	25.41	100.00	40.53	25.41
Innsbruck_DS_backtranslation	3183	27.35	84.91	41.38	38.86	27.35	84.91	41.38	38.86
Innsbruck_DS_r1	3183	27.32	86.89	41.57	37.92	27.32	86.89	41.57	37.92
Innsbruck_DS_synonym	3183	27.15	86.89	41.37	37.41	27.15	86.89	41.37	37.41

Table 5. Results for Task 1 (pun detection) in French

run ID	#	P	R	F_1	A	P^*	R^*	F_1^*	A^*
Croland_FR_GPT3	12873	100.00	01.14	02.27	59.24	100.00	01.14	02.27	59.24
LJGG_t5_large_easy_fr	12873	55.13	64.29	59.36	63.70	55.13	64.29	59.36	63.70
LJGG_t5_large_label_fr	12873	41.23	100.00	58.39	41.23	41.23	100.00	58.39	41.23
LJGG_t5_large_no_label_fr	12873	41.23	100.00	58.39	41.23	41.23	100.00	58.39	41.23
Les_miserables_fasttext	12873	58.57	19.76	29.55	61.15	58.57	19.76	29.55	61.15
Les_miserables_random	12873	41.14	49.81	45.06	49.92	41.14	49.81	45.06	49.92
Les_miserables_simplet5	12873	59.72	74.88	**66.45**	68.82	59.72	74.88	66.45	68.82
Smroltra_FR_FastText	12873	55.24	25.00	34.42	60.72	55.24	25.00	34.42	60.72
Smroltra_FR_Logistic-Regression	12873	58.43	60.39	59.40	65.95	58.43	60.39	59.40	65.95
Smroltra_FR_MLP	12873	56.49	62.88	59.52	64.73	56.49	62.88	59.52	64.73
Smroltra_FR_NBC	12873	56.73	63.18	59.78	64.94	56.73	63.18	59.78	64.94
Smroltra_FR_Random	12873	42.14	67.70	51.95	48.36	42.14	67.70	51.95	48.36
Smroltra_FR_SimpleT5	12873	61.21	67.69	64.29	68.99	61.21	67.69	64.29	68.99
Smroltra_FR_TFIDF	12873	58.77	62.09	60.38	66.41	58.77	62.09	60.38	66.41
UBO_SimpleT5	12871	67.80	58.76	62.95	71.49	67.80	58.76	62.95	71.48
AKRaNLU_sentemb	12873	41.18	73.88	52.88	45.71	41.18	73.88	52.88	45.71
AKRaNLU_seqclassification	12873	41.23	100.00	58.39	41.23	41.23	100.00	58.39	41.23

Table 6. Results for Task 1 (pun detection) in Spanish

run ID	#	P	R	F_1	A	P^*	R^*	F_1^*	A^*
Croland_ES_GPT3	2241	98.07	05.35	10.15	59.75	98.07	05.35	10.15	59.75
LJGG_t5_large_easy_es	2230	50.34	54.09	52.15	57.83	50.34	54.26	52.23	57.75
LJGG_t5_large_label_es	2230	42.55	99.68	**59.64**	42.70	42.55	100.00	59.70	42.55
LJGG_t5_large_no_label_es	2230	42.55	99.68	**59.64**	42.70	42.55	100.00	59.70	42.55
Les_miserables_fasttext	2230	0.00	0.00	0.00	57.51	0.00	0.00	0.00	57.44
Les_miserables_random	2230	43.43	51.78	47.24	50.87	43.43	51.94	47.31	50.76
Les_miserables_simplet5	2230	51.10	17.01	25.53	57.83	51.10	17.07	25.59	57.75
NLPalma_BERT	2230	55.94	40.54	47.01	61.17	55.94	40.67	47.10	61.12
Smroltra_ES_FastText	2238	40.75	0.625	49.33	45.47	40.75	0.625	49.33	45.39
Smroltra_ES_Logistic-Regression	2238	0.50	49.05	49.52	57.51	0.50	49.05	49.52	57.46
Smroltra_ES_MLP	2238	55.45	44.32	49.27	61.22	55.45	44.32	49.27	61.17
Smroltra_ES_NBC	2238	47.69	56.40	51.68	55.19	47.69	56.40	51.68	55.13
Smroltra_ES_Random	2241	42.05	67.85	51.92	46.63	42.05	67.85	51.92	46.63
Smroltra_ES_SimpleT5	2238	44.31	46.21	45.24	52.47	44.31	46.21	45.24	52.41
Smroltra_ES_TFIDF	2238	53.34	46.11	49.46	59.97	53.34	46.11	49.46	59.91
UBO_SimpleT5	2230	51.28	62.92	56.50	58.85	51.28	63.11	56.58	58.78
AKRaNLU_sentemb	2230	41.39	72.26	52.63	44.75	41.39	72.49	52.70	44.61
AKRaNLU_seqclassification	2230	42.55	99.68	**59.64**	42.70	42.55	100.00	59.70	42.55

In pun interpretation (Task 2.2), systems must indicate the two meanings of the pun. In JOKER-2023, 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*}.

Task 2.2 is evaluated with the precision, recall, and F-score metrics as used in word sense disambiguation [24], except that each instance is scored as the average score for every of its 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 receive a score of 1 for Example 1, and a system guessing {*fuel*}, {*prophet*} would receive a score of $1/2$.

3.1 Data

The pun location data is drawn from the positive examples of Task 1, with each text being accompanied by an annotation that reproduces the word being punned upon, as described above.

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 did contain 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 French data, we used a simplified version of the annotation made in JOKER-2022 [11].

As in Task 1, the data was split into test and training sets and 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; for training data for the pun interpretation task, there is an additional field giving the two synonym/hypernym sets. System output is expected as a JSON or delimited text file with fields for the run ID, text ID, the pun word (for pun location) or its synonym/hypernym sets (for pun interpretation), and a boolean flag indicating whether the run is manual or automatic.

3.2 Participants' approaches

The AKRaNLU team participants [7] 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. For pun interpretation, the results

from the pun location subtask were used 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.

The MiCroGerk team [27] chose an LLM approach for Task 2, 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.

The Smroltra team [25] 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. On the other hand, for interpretation, the other methods except for BLOOM were not as effective as expected, including even GPT-3 and various combinations of the methods used with WordNet for location prediction.

TeamCAU [2] used various LLMs. T5 showed good results in comparison to BLOOM and models from AI21 (albeit for partial runs only).

FastText, Ridge, Naive Bayes, SimpleT5, and SimpleTransformersT5 were used by the participants of ThePunDetectives team [23]. They found the best results to be produced by the pre-trained models. In particular, T5 achieved good performance, as predicted by the authors.

For the location and interpretation tasks, the UBO team [8] opted to use T5 (SimpleT5).

The UBO-RT team [4] approached pun location and interpretation in English and French using post-edited output of ChatGPT. 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.

The Croland team [19] used GPT-3.

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.

3.3 Results

Table 7 reports the participants' results for wordplay location in English, French, and Spanish. As some participants submitted only partial runs, we provide two sets of accuracy scores: those labelled A are based on the total number of instances in the test set, while those labelled A^* are based on the actual number of attempted instances (#).

Accuracy scores for pun location in English and Spanish ($A \approx 80$) are twice as good as those for French ($A \approx 40$). This could be explained by the fact that participants used large language models 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 7. Results for Task 2.1 (pun location)

	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						

Owing to various scheduling and technical issues, the pun interpretation results were not ready at the time the manuscript for this paper was submitted. We will provide them in a future article, to be published either in the CLEF CEUR proceedings [1] or on a public preprint server such as arXiv. A link to this article will be provided on the JOKER website at `http://www.joker-project.com/`.

4 Task 3: Translation of puns from English to French and Spanish

In Task 3, participating systems attempt to translate English punning jokes into French and Spanish. The translations should aim to preserve, to the extent possible, both the form and meaning of the original wordplay – that is, to implement the pun→pun strategy described in Delabastita's typology of pun translation strategies [5,6]. For example, Example 2 above ("I used to be a banker but I lost interest") might be rendered into French as "*J'ai été banquier mais j'en ai perdu tout l'intérêt*". This fairly straightforward translation preserves the pun, since *interest* and *intérêt* share the same ambiguity.

4.1 Data

Our French training data contains 5,838 translations of 1,405 distinct puns in English as in Tasks 1 and 2. These translations come from translation contests and the JOKER-2022 track [11,12]. A detailed description of the corpus can be found in our SIGIR 2023 paper [9]. For the test set, we provided participants with 4,290 distinct puns in English to be translated into French and Spanish. Then, we manually evaluated 6,590 French translations of 1,197 distinct puns in English pooled from the participants' runs used as the final test data.

We also provide new sets of English–Spanish translations of punning jokes, similar to English–French datasets we produced for JOKER-2022. These translations were sourced via a translation contest in which professional translators were asked to translate 400 English puns. In total, they produced 2,459 pairs of translated puns. These translations underwent an expert review to ensure compliance with the data set's criteria of preserving both wordplay and the general meaning. We kept 644 translations of 217 distinct English puns for training data. We manually evaluated 5,727 translations of 544 distinct English puns.

The training and test data was provided in JSON and in delimited text formats with fields containing the text of the punning joke and a unique ID; for training there were one or two additional fields containing gold-standard translations of the text into French and/or Spanish. Systems were expected to output a JSON or delimited text file containing the run ID, text ID, the text of the translation(s) into French and/or Spanish, and a boolean flag indicating whether the run was manual or automatic.

4.2 Evaluation

As we have previously argued [12,11], vocabulary overlap metrics such as BLEU are unsuitable for evaluating wordplay translations. We therefore continue JOKER-2022's practice of having trained experts manually evaluate system translations according to features such as lexical field preservation, sense preservation, wordplay form preservation, style shift, humorousness shift, etc. and the presence or absence of errors in syntax, word choice, etc.

Participants' runs were subject to whitespace trimming, lower-casing, and were pooled together. We then filtered out French and Spanish translations identical to the original wordplay in English, as we considered these wordplay instances to be untranslated. The runs are ranked according to the number of successful translations – i.e., translations preserving, to the extent possible, both the form and sense of the original wordplay.

4.3 Participants' approaches

The LJGG team [16] submitted runs for translation from English to French and Spanish. Their model is a three-stage architecture based on T5 (SimpleT5). The two stages calculate the information necessary to concatenate the English sentence, which forms an input for the third neural network. For training the models, they enlarged Task 3's dataset with the data prepared for Task 1. They also used the DeepL translator to compare their results and found that the DeepL translations are better.

The NLPalma team [26] approached the translation of wordplay from English to Spanish using BLOOMZ & mT5, which is an improved version of BLOOM.

The MiCroGerk team [27] used SimpleT5-, BLOOM-, OpenAI-, and AI21-based models and the models from the EasyNMT package (Opus-MT, mBART50_m2m, and M2M_10) for the English–Spanish translation task. The OpenAI- and AI21-based models proved to be the best, with the lowest-ranked models being SimpleT5. According to the authors, however, there is still plenty of room for improvement.

The UBO team [8] used the models from the EasyNMT package – namely, Opus-MT, mBART50_m2m, and M2M_100.

The TheLangVerse team [15] made use of the j2-grande model from the AI21 platform. They also combined the datasets to provide more content for fine-tuning, obtaining results comparable to those obtained from their surveys.

Opus-MT and M2M_100 from the the EasyNMT package were selected by participants of ThePunDetectives team [23]. The authors found that M2M_100 made translations that diverged from the original senses at the expense of precision. In contrast, Opus-MT presented a slightly better translation capability, being able to comprehend some types of humour.

The solution of the Smroltra team [25] was to use the GPT-3, BLOOM, Opus-MT, and mBART50_m2m models from EasyNMT; SimpleT5; and the Google Translate service for both English–Spanish and English–French translations. The best results were obtained using GPT-3, while the worst came from T5, which

produced incoherent sentences. GPT-3 and BLOOM obtained the highest scores on both datasets, although according to the authors, the translation of the datasets requires more data and time.

Finally, the Croland team [19] approached the task using GPT-3.

4.4 Results

Tables 8 and 9 present the scores for participants' runs submitted for translations into French and Spanish, respectively. We report the following scores:

#E number of manually evaluated translations
#T number of submitted translations used for evaluation
#M number of translations preserving the meaning of the source puns
%M percentage of translations preserving the meaning of the source puns
#W number of translations containing wordplay
%W percentage of translations containing wordplay
#S number of translations containing wordplay and preserving the meaning of the source puns
%S percentage of translations containing wordplay and preserving the meaning of the source puns
%R percentage of translations containing wordplay and preserving the meaning of the source puns over the total test set

We rank the runs according to #**S**. For French, the best results were obtained by the Jurassic-2 model and T5. (Note that participants trained the T5 model on the training set while other LLMs were used in a few-shot setup.) For Spanish, the best results were obtained by systems using Google Translate or T5. As in 2022 [12,11], we observe that the success rate of wordplay translation is extremely low even in the case of LLMs, with the maximum value of 6% over the total evaluated test set for French. This score goes up to 18% for Spanish.

5 Open Task

We received three submissions for the Open Task, raising different challenges.

5.1 Pun generation for text transformation and conversational systems

Glemarec & Charles [14] proposed an experiment for wordplay generation. Their motivation was to integrate similar techniques into interactive systems (narratives, virtual agents) to favor engagement. This work is an update and expansion of a paper presented at the previous JOKER lab at CLEF 2022 [13]. In the latter work, the authors had proposed a pipeline for pun generation in French, using Jurassic [20] as a Large Language Model and substituting a word in a source sentence containing a homophonic word, to provide a context appropriate for creating a new punning sentence. In this year's submission, they used the GPT-3

Table 8. Results for Task 3 (pun translation, English to French)

run ID	#E	#T	#M	%M	#W	%W	#S	%S	%R
Croland_task_3_EN_FR_GPT3	16	28	4	25	0	0	0	0	0
LJGG_Google_Translator_EN_FR_-auto	1,076	1,197	580	53	67	6	63	5	5
LJGG_task3_fr_mt5_base_auto	2	1,197	2	100	1	50	1	50	0
LJGG_task3_fr_mt5_base_no_label_-auto	1	1,197	1	100	0	0	0	0	0
LJGG_task3_fr_t5_large_auto	90	1,197	24	26	2	2	2	2	0
LJGG_task3_fr_t5_large_no_label_-auto	140	1,197	80	57	15	10	15	10	1
Smroltra_task_3_EN-FR_BLOOM	31	32	8	25	0	0	0	0	0
Smroltra_task_3_EN-FR_EasyNMT-Opus	786	1,197	427	54	58	7	56	7	4
Smroltra_task_3_EN-FR_EasyNMT-mbart	1139	1,197	613	53	68	5	64	5	5
Smroltra_task_3_EN-FR_GPT3	30	32	8	26	0	0	0	0	0
Smroltra_task_3_EN-FR_GoogleTrans-lation	1109	1,197	602	54	71	6	67	6	5
Smroltra_task_3_EN-FR_SimpleT5	1043	1,197	562	53	66	6	65	6	5
TeamCAU_task_3_EN-FR_AI21	30	32	8	26	0	0	0	0	0
TeamCAU_task_3_EN-FR_BLOOM	32	32	8	25	0	0	0	0	0
TeamCAU_task_3_EN-FR_ST5	1090	1,197	577	52	71	6	69	6	5
TheLangVerse_task_3_j2-grande-finetuned	1176	1,197	636	54	76	6	**72**	6	6
ThePunDetectives_task_1,3_EN-FR_-M2M100	13	340	9	69	2	15	2	15	0
ThePunDetectives_task_1,3_EN-FR_-OpusMT	183	340	92	50	19	10	17	9	1
UBO_task_3_SimpleT5	73	1,195	47	64	5	6	5	6	0
UBO_task_3_SimpleT5_x	1148	1,195	616	53	71	6	67	5	5
UBO_task_3_SimpleT5_y	791	1,194	429	54	61	7	59	7	5

Table 9. Results for Task 3 (pun translation, English to Spanish)

run ID	#E	#T	#M	%M	#W	%W	#S	%S	%R
Croland_task_3_ENESGPT3	45	47	9	20.00	3	6.66	3	6.66	0
LJGG_task3_es_mt5_base_auto	34	544	16	47.05	5	14.70	5	14.70	0
LJGG_task3_es_mt5_base_no_label_auto	34	544	16	47.05	5	14.70	5	14.70	0
LJGG_task3_es_t5_large_auto	34	544	16	47.05	5	14.70	5	14.70	0
LJGG_task3_es_t5_large_no_label_auto	34	544	16	47.05	5	14.70	5	14.70	0
LJGG_task_3_GoogleTranslatorENESauto	544	544	274	50.36	106	19.48	**99**	18.19	18
NLPalma_task_3_BLOOMZ_x	359	359	215	59.88	85	23.67	80	22.28	14
NLPalma_task_3_BLOOMZ_y	359	359	215	59.88	85	23.67	80	22.28	14
Smroltra_task_3_EN-ES_EasyNMT-Opus	529	544	263	49.71	100	18.90	93	17.58	17
Smroltra_task_3_EN-ES_EasyNMT-Opus_x	529	544	263	49.71	100	18.90	93	17.58	17
Smroltra_task_3_EN-ES_EasyNMT-Opus_y	529	544	263	49.71	100	18.90	93	17.58	17
Smroltra_task_3_EN-ES_GoogleTranslation	532	544	267	50.18	103	19.36	96	18.04	17
Smroltra_task_3_EN-ES_SimpleT5	531	544	265	49.90	101	19.02	94	17.70	17
Smroltra_task_3_ENESBLOOM	45	47	8	17.77	2	4.44	2	4.44	0
TheLangVerse_task_3_j2-grande-finetuned	415	544	200	48.19	70	16.86	65	15.66	11
ThePunDetectives_task_1.3_EN-ES_-M2M100	33	430	16	48.48	7	21.21	7	21.21	1
ThePunDetectives_task_1.3_ENESOpusMT	428	430	208	48.59	71	16.58	66	15.42	12
MiCroGerk_task_3_EN-ES_OpenAI	6	17	3	0.5	1	16.66	1	16.66	0
MiCroGerk_task_3_EN-ES_mbart50_-m2m_x	543	544	274	50.46	106	19.52	99	18.23	18
MiCroGerk_task_3_EN-ES_AI21_x	1	17	1	1	0	0	0	0	0
MiCroGerk_task_3_EN-ES_mbart50_-m2m_y	543	544	274	50.46	106	19.52	99	18.23	18
MiCroGerk_task_3_EN-ES_m2m_-100_418M	43	544	23	53.48	11	25.58	11	25.58	2
MiCroGerk_task_3_EN-ES_SimpleT5	5	544	4	0.8	3	0.6	3	0.6	0

API in addition to libraries for recognizing paronyms in English, hence not requiring the services of a phonetic lexicon. Several examples of generated outputs are presented in their paper. Although there is no quantitative or qualitative evaluation provided, and despite that the target language being different (which makes it difficult to compare to the last year's results), the curated examples provided seem successful at providing new humorous puns.

5.2 Sentiment analysis for wordplay

Thomas-Young & Ermakova [29] presented a sentiment analysis of the corpora, using the Microsoft Azure Service. They deem their results inconclusive: the use of sentiment analysis at the word level as well as for context analysis seems to require specially designed models.

5.3 Comparison of machine and human performances

Große-Bolting et al. [15] considered the performance not just of machine learning algorithms but also humans in JOKER tasks using the English corpus.

For the evaluation of human competence on JOKER tasks, a survey was conducted in four countries where English is not a mother tongue: Poland, France, Spain and Germany. The survey used ten randomly selected punning sentences from a curation of 100 in our corpus. In addition to questions for estimating the English proficiency of respondents, questions were asked for determining if respondents could locate the pun, understand the pun, and provide a translation of the pun in their native languages for ten random entries of the JOKER corpus. The answers allowed the authors to check how well the participants performed on the location, interpretation, and translation tasks.

Standard metrics such as recall, precision and F_1 were used to compare respective performances. Participants scored 0.74 F_1 for classification, 0.2 F_1 for location, albeit with low inter-rater reliability. The authors noted that the low score humans achieved in pun location can be beaten by a simple system which always selects the last word of the punning sentence (which, the authors claim, achieves an F_1 of 0.35).

Their results echo the work of Bell [3], who noted that being able to understand humour in a foreign language is a particular challenge for learners.

6 Conclusion

In this paper, we described the JOKER track at CLEF 2023, consisting of three interconnected shared tasks on automatic wordplay analysis and translation, as well as an open task. These tasks aim to advance the automation of creative-language translation by developing the requisite parallel data and evaluation metrics for detecting, locating, interpreting, and translating wordplay. Thirteen teams submitted 176 runs for the shared tasks. We received many partial runs due to token/time constraints of LLMs.

Our results in general suggest that wordplay detection and location are still a challenge for LLMs despite their recent significant advances. For the pun detection task for all three languages, the improvement of the best runs over the random results are less than 15 percentage points according to F_1 score. We also observe that the results of the same methods depend heavily on implementation, fine-tuning, and/or prompts used. For French, we can see a slight improvement over English, which might be explained by higher similarity between training and test data in French; this data comes from the translation of overlapping sets English puns, while the test set in English contains different puns without semantic or vocabulary similarity. Surprisingly, Logistic Regression and the TF–IDF classifier demonstrated comparable results. These results might suggest that efficient training of lighter models could help to achieve results comparable to large pre-trained models, which are very expensive and resource-consuming.

Accuracy scores for pun location in English and Spanish are twice as high as those for French. This could be explained by the fact that participants used large language models that might have included in their training data some of the puns found in our corpus, which were sourced from the web directly or indirectly by applying LLMs. By contrast, the French wordplay data was largely constructed by us and not previously published online.

We observe that the success rate of wordplay translation is extremely low even in the case of LLMs, with the maximum value of 6% over the total evaluated test set for French and 18% for Spanish.

Further details on the shared tasks and the submitted runs can be found in the CLEF CEUR proceedings [1]. Additional information on the track is available on the JOKER website: `http://www.joker-project.com/`

Acknowledgements. 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*. We thank Carolina Palma Preciado, Leopoldo Jesús Gutierrez Galeano, Khatima El Krirh, Nathalie Narváez Bruneau, and Rachel Kinlay for their help and support in the first Spanish pun translation contest. We also thank Quentin Dubreuil, Keith Salina, Constance Germann, Océane Brunelière, Aurianne Damoy, Angelique Robert, and all other colleagues and students who participated in data construction, the translation contests, and the CLEF JOKER track.

References

1. Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum. CEUR Workshop Proceedings, CEUR-WS.org (2023)
2. Anjum, A., Lieberum, N.: Exploring Humor in Natural Language Processing: A Comprehensive Review of JOKER Tasks at CLEF Symposium 2023. In: Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum [1]

3. Bell, N.D.: Learning about and through humor in the second language classroom. Language Teaching Research **13**(3), 241–258 (Jul 2009). https://doi.org/10.1177/1362168809104697
4. Brunelière, O., Germann, C., Salina, K.: CLEF 2023 JOKER Task 2: using Chat GPT for pun location and interpretation. In: Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum [1]
5. Delabastita, D.: There's a Double Tongue: an Investigation into the Translation of Shakespeare's Wordplay, with Special Reference to Hamlet. Rodopi, Amsterdam (1993)
6. Delabastita, D.: Wordplay as a translation problem: a linguistic perspective. In: Ein internationales Handbuch zur Übersetzungsforschung, vol. 1, pp. 600–606. De Gruyter Mouton (7 2008). https://doi.org/10.1515/9783110137088.1.6.600
7. Dsilva, R.R.: AKRaNLU @ CLEF JOKER 2023: Using sentence embeddings and multilingual models to detect and interpret wordplay. In: Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum [1]
8. Dubreuil, Q.: UBO Team @ CLEF JOKER 2023 track for task 1, 2 and 3 - applying AI models in regards to pun translation. In: Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum [1]
9. Ermakova, L., Bosser, A.G., Jatowt, A., Miller, T.: 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). https://doi.org/10.1145/3539618.3591885
10. Ermakova, L., Miller, T., Bosser, A.G., Preciado, V.M.P., Sidorov, G., Jatowt, A.: Science for fun: The CLEF 2023 JOKER track on automatic wordplay analysis. In: Kamps, J., Goeuriot, L., Crestani, F., Maistro, M., Joho, H., Davis, B., Gurrin, C., Kruschwitz, U., Caputo, A. (eds.) Advances in Information Retrieval: 45th European Conference on Information Retrieval, ECIR 2023, Dublin, Ireland, April 2–6, Proceedings, Part III. Lecture Notes in Computer Science, vol. 13982, pp. 546–556. Springer, Berlin, Heidelberg (Apr 2023). https://doi.org/10.1007/978-3-031-28241-6_63
11. Ermakova, L., Miller, T., Regattin, F., Bosser, A.G., Borg, C., Élise Mathurin, Corre, G.L., Araújo, S., Hannachi, R., Boccou, J., Digue, A., Damoy, A., Jeanjean, B.: Overview of JOKER@CLEF 2022: Automatic wordplay and humour translation workshop. In: Barrón-Cedeño, A., Martino, G.D.S., Esposti, M.D., Sebastiani, F., Macdonald, C., Pasi, G., Hanbury, A., Potthast, M., Faggioli, G., Ferro, N. (eds.) Experimental IR Meets Multilinguality, Multimodality, and Interaction: Proceedings of the Thirteenth International Conference of the CLEF Association (CLEF 2022). Lecture Notes in Computer Science, vol. 13390, pp. 447–469. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-13643-6_27
12. Ermakova, L., Regattin, F., Miller, T., Bosser, A.G., Borg, C., Jeanjean, B., Élise Mathurin, Corre, G.L., Hannachi, R., Araújo, S., Boccou, J., Digue, A., Damoy, A.: Overview of the CLEF 2022 JOKER Task 3: Pun translation from English into French. In: Faggioli, G., Ferro, N., Hanbury, A., Potthast, M. (eds.) Proceedings of the Working Notes of CLEF 2022 – Conference and Labs of the Evaluation Forum, Bologna, Italy, September 5th to 8th, 2022. CEUR Workshop Proceedings, vol. 3180, pp. 1681–1700 (Aug 2022)
13. Glemarec, L., Bosser, A., Boccou, J., Ermakova, L.: Humorous wordplay generation in french. In: Faggioli, G., Ferro, N., Hanbury, A., Potthast, M. (eds.) Proceedings of the Working Notes of CLEF 2022 – Conference and Labs of the Evaluation

Forum, Bologna, Italy, September 5th to 8th, 2022. CEUR Workshop Proceedings, vol. 3180, pp. 1793–1806. CEUR-WS.org (2022)
14. Glemarec, L., Charles, F.: BU-Pier Team @ CLEF JOKER 2023 Open Task: Slip of the tongue generation to improve social interaction with virtual agents. In: Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum [1]
15. Große-Bolting, G., Ledworowska, A., Cross, I.: JOKER: Automatic Wordplay Analysis task 1 - Pun detection task 3 - pun translation open task - human performance on JOKER wordplay classification. In: Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum [1]
16. Gutiérrez-Galeano, L.: LJGG @ CLEF JOKER Tasks 1 and 3: A comparison of T5 and mT5 for different languages and tasks. In: Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum [1]
17. Jentzsch, S., Kersting, K.: ChatGPT is fun, but it is not funny! Humor is still challenging large language models (2023)
18. Kolb, W., Miller, T.: Human–computer interaction in pun translation. In: Hadley, J.L., Taivalkoski-Shilov, K., Teixeira, C.S.C., Toral, A. (eds.) Using Technologies for Creative-Text Translation, pp. 66–88. Routledge (2022). https://doi.org/10.4324/9781003094159-4
19. Komorowska, J., Čatipović, I., Vujica, D.: CLEF2023' JOKER Working Notes. In: Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum [1]
20. Lieber, O., Sharir, O., Lentz, B., Shoham, Y.: Jurassic-1: Technical details and evaluation. White paper, AI21 Labs (Aug 2021), https://uploads-ssl.webflow.com/60fd4503684b466578c0d307/61138924626a6981ee09caf6_jurassic_tech_paper.pdf
21. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
22. Miller, T., Hempelmann, C.F., Gurevych, I.: SemEval-2017 Task 7: Detection and interpretation of English puns. In: Proceedings of the 11th International Workshop on Semantic Evaluation. pp. 58–68 (Aug 2017). https://doi.org/10.18653/v1/S17-2005
23. Ohnesorge, F., Gutiérrez, M.Á., Plichta, J.: CLEF 2023 JOKER Tasks 2 and 3: using NLP models for pun location, interpretation and translation. In: Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum [1]
24. Palmer, M., Ng, H.T., Dang, H.T.: Evaluation of WSD systems. In: Agirre, E., Edmonds, P. (eds.) Word Sense Disambiguation: Algorithms and Applications, chap. 4, pp. 75–106. No. 33 in Text, Speech, and Language Technology, Springer (2007)
25. Popova, O., Dadić, P.: 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: Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum [1]
26. Preciado, V.M.P., Preciado, C.P., Sidorov, G.: NLPalma @ CLEF 2023 JOKER: A BLOOMZ and BERT approach for wordplay detection and translation. In: Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum [1]
27. Prnjak, A., Davari, D.R., Schmitt, K.: CLEF 2023 JOKER Task 1, 2, 3: pun detection, pun interpretation, and pun translation. In: Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum [1]

28. Reicho, S., Jatowt, A.: Innsbruck @ CLEF JOKER 2023 Track's Task 1: Data augmentation techniques for humor recognition in text. In: Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum [1]
29. Thomas-Young, T.: Why Sentiment Analysis is a joke with JOKER data? word-level and interpretation analysis (CLEF 2023 JOKER task 2). In: Proceedings of the Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum [1]