# CROWD SCORE: A METHOD FOR THE EVALUATION OF JOKES USING LARGE LANGUAGE MODEL AI VOTERS AS JUDGES

**Fabrício Góes, Zisen Zhou**
Computing and Mathematical Sciences Department
University of Leicester
Leicester, UK
{fabricio.goes,zz254}@leicester.ac.uk,

**Piotr Sawicki, Marek Grzes**
School of Computing
University of Kent
Canterbury, UK
{p.sawicki,m.grzes}@kent.ac.uk

**Daniel G. Brown**
David R. Cheriton School of Computer Science University of Waterloo
Waterloo, Canada
dan.brown@uwaterloo.ca

**ABSTRACT**

This paper presents the Crowd Score, a novel method to assess the funniness of jokes using large language models (LLMs) as AI judges. Our method relies on inducing different personalities into the LLM and aggregating the votes of the AI judges into a single score to rate jokes. We validate the votes using an auditing technique that checks if the explanation for a particular vote is reasonable using the LLM. We tested our methodology on 52 jokes in a crowd of four AI voters with different humour types: affiliative, self-enhancing, aggressive and self-defeating. Our results show that few-shot prompting leads to better results than zero-shot for the voting question. Personality induction showed that aggressive and self-defeating voters are significantly more inclined to find more jokes funny of a set of aggressive/self-defeating jokes than the affiliative and self-enhancing voters. The Crowd Score follows the same trend as human judges by assigning higher scores to jokes that are also considered funnier by human judges. We believe that our methodology could be applied to other creative domains such as story, poetry, slogans, etc. It could both help the adoption of a flexible and accurate standard approach to compare different work in the CC community under a common metric and by minimizing human participation in assessing creative artefacts, it could accelerate the prototyping of creative artefacts and reduce the cost of hiring human participants to rate creative artefacts. [1]

***Keywords*** Large Language Models · Jokes Evaluation · Computational Creativity · Crowd Score · Prompt Engineering · AI judges · Personality Induction

## 1 Introduction

In Computational Creativity (CC), there are two main strategies to assess the creativity of artefacts: evaluation metrics and human judges [1, 2]. Evaluation metrics, proposed by humans, are usually used by generative systems to evaluate novelty and value of potential creative artefact candidates. The best ones are ultimately evaluated by humans, since they are still the ultimate judges on creativity. Despite evidence that non-expert judges cannot appropriately evaluate the creativity of a human or machine [3], studies have relied on hiring non-expert volunteers on crowd sourcing platforms, such as Amazon Mechanical Turk, to evaluate/rate artifacts in the creative domain [4]. These studies usually do not ask volunteers to explain the reasoning behind their scores, but accept their judgement as valid. In this paper, we assume that machines, much like non-expert humans, can be used to judge the creativity of artefacts. This assumption is backed on recent advances of large language models (LLMs) such as GPT-3 [5], that enable emergent behaviour through few-shot and augmenting prompting [6, 7], that is, abilities that were not present in smaller models. In particular, jokes

[1] Abstract fully written by GPT-3 using the introduction and related work sections as input.

and humour are challenging for machines because they involve complex concepts such as irony, sarcasm, and puns [8]. Jokes often rely on cultural context and knowledge, which can be difficult for a machine to access. However, recent work [9, 7, 10, 11, 12, 13, 14] shows that prompting or fine-tuning LLMs for humour detection is a viable approach. On top of it, some recent publications show that LLMs can be configured/prompted to assume different personalities with zero and few-shot prompting [15]. This favours the creation of a crowd of AI voters, where each one's vote is aggregated into a rating/score that can accurately measure the level of funniness of jokes, the CC-Crowd Score. In order to validate those votes, we apply an auditing technique that checks if the explanation for a particular vote is reasonable using the LLM. We believe that this method could be applied to other creative domains such as story, poetry, slogans, etc. It could both help the adoption of a flexible and accurate standard approach to compare different work in the CC community under a common metric and by minimizing human participation in assessing creative artefacts, we can accelerate the prototyping of creative artefacts and reduce the cost of hiring human participants to rate creative artefacts. In this paper, we focus on evaluating funniness of jokes as a case study.

We tested our methodology to assess the funniness of jokes from [4] in a crowd of four voters with different humour types [16]. Our results show that: i) few-shot prompting leads to better results than zero-shot for the voting question, where picking the least appropriate opposite word can reduce balanced accuracy by 26% and 25% for zero-shot and few-shot respectively; ii) personality induction showed that aggressive and self-defeating voters are significantly more inclined to find more jokes funny of a set of aggressive/self-defeating jokes than the affiliative and self-enhancing voters; and iii) the Crowd Score follows the same trend as human judges by assigning higher scores to jokes that are also considered funnier by human judges.

Our main contributions are:

- The Crowd Score, a novel method to assess jokes with LLMs using their intrinsic evaluation metrics. It relies on AI voters as judges for creativity instead of human judges, in which a crowd of AI voters are induced in a LLM and their votes are aggregated into a single score to rate jokes.
- An auditing technique to validate the votes of the AI judges using LLMs.
- A case study with 52 jokes and 4 induced personalities to assess the funniness of jokes.
- A set of prompt templates that could be customized to assess other creative artefacts.

The rest of this paper is organized as follows. In Section 2, we present the background and recent related work to support this research. In Section 3, we describe our Crowd Score method in details. Section 4 presents and analyzes the experimental results. This paper is concluded in Section 5.

## 2 Related Work

A major issue on humour research evaluation has been the lack of comprehensive datasets of jokes with ratings of funniness. [17] points out that most humour datasets are usually annotated in a binary fashion (funny or not funny), which does not capture the level of funniness. A fine grained evaluation of jokes reduces the chance of misjudging jokes that are considered borderline on average, but funny for particular groups of people. However, some recent publications make available databases with annotations on the rating of jokes [4, 18]. [17] also tackles that issue using crowd-sourcing to ask human volunteers to rate the funniness of modified headlines.

The use of large language models (LLMs), such as GPT-3, is becoming more prevalent for generating humorous texts. For instance, [9] rely on human participants to evaluate Chinese crosstalks (comic dialogues) generated by a fine-tuned GPT-3 in regard to general quality, humour, coherence and ethical risky content. The results show that the best generation achieved 65% of general quality, and that the humour criterion is not satisfied. It is important to remark that they used the standard BLEU metric to evaluate the performance of GPT-3. [12] also generates puns using GPT-3. Based on context words with ambiguous meanings, they provide the target pun word and its two meanings to GPT-3 and prompt it to generate puns. They also ask humans to evaluate if the generated puns are indeed puns and rate how funny they are. In [11], the authors propose a pun generator which, given a pun word pair, retrieves a context word and phrase, and they use GPT-2 to produce a pun. Human evaluators judged that their method achieved the highest funniness among other pun generators. In [10], the authors propose an approach to predict the funniness of news headlines using the BERT model. The approach was tested with the humour datasets from the SemEval-2020 workshop, achieving high performance. We can observe that there are two main approaches to LLM's: fine-tuning [9, 7, 10, 11] and prompting [12, 13, 14]. However, most related work relies on human evaluators as the final judges of humour. We depart from this practice and claim that LLMs can be used as humour evaluators, even in their current stage.

An innovative approach is described in [7], in which the authors evaluate if large language models are capable of evaluating and explaining captions of the New Yorker Caption Contest. Those captions are humourous sentences

describing a cartoon. Results show that a fine-tuned GPT-3 cannot recognize the captions' relevance, evaluate or explain them as effectively as humans. However, their partial capacity can be sufficient to work as creative collaborators. This work strongly corroborates our claim that despite being imperfect, current LLMs can be used as judges of humour. We move one step further and rely on concepts introduced in [14] to induce LLMs with different personalities to provide different opinions/rating about the humour of a joke.

In [14], the authors propose a method to induce a certain personality on large language models. The personality is based on the Big Five personality factors. The paper compares two approaches. The naive one uses zero-shot learning to prompt a personality using "You are a/an X person", where X is one of the five factors, and the "Word Level Auto-Prompt" approach imposes the 3 most significant words to describe each factor. Both approaches are evaluated by prompting the model to answer the Big Five questionnaire. Results show that it is possible to simulate the desired personality using both methods. Another work [19] uses GPT-3 to simulate responses of humans by varying their names and other details, under some human experiments such as the Ultimatum Game. Results show that GPT-3 responses are consistent with prior human studies. Another study [20] used GPT-3 to generate data to train conversational models and compared them with real data generated by humans. The authors assume that large language models can be used to replace humans when data is scarce. In our proposal, we combine the possibility of inducing a certain personality with the capability of LLMs to evaluate the humour level of jokes to create a method, as an alternative to traditional metrics, that could also be applied to evaluate jokes.

The use of LLMs as an alternative to traditional metrics is supported by recent research in [21], where the authors show that automatic reference-free and reference-based metrics, such as BLEU, BERTScore, BLANC and QuestEval, are ineffective to evaluate the quality of news summaries generated by zero-shot GPT-3, when compared to the human evaluation. Instead of relying on a single metric, we use the idea of aggregation as in [22], where aggregation is used to Q&A tasks for large language models. By using prompt chains, an input claim is converted into a question, and multiple noisy answers are generated. Those answers, which are binary (yes/no), are aggregated using weak supervision into a final prediction. Our work proposes aggregation of binary votes generated by multiple voters with induced personalities in large language models, instead of imperfect/noisy answers as in [22].

Another issue is the reliability of prompting GPT-3. In [23], prompts are created to induce reliable behaviors on GPT-3. The paper shows that retrieving evidence passages for Q&A problems can improve the performance of GPT-3. It also shows that GPT-3 memorized answers can be updated by adding conflicting passages in the prompts, allowing it to output different answers in accordance with this new context. Differently, in our work, we employ a voting question prompt to ensure that we pick the best pair of words in the binary classification/voting. We also propose the concept of auditing, where the LLM is used to check if the explanations of the votes by each voter are reasonable/consistent before computing the respective votes as valid.

Finally, in terms of creativity, on top of the other related work presented, [24] assessed GPT-3's creativity using the Guilford's Alternative Uses Test and compared with human responses on value, novelty and surprise. They have used a very detailed, handcrafted prompt to instruct GPT-3 to list the creative uses of objects. Their results show that humans rating outperform GPT-3 only by a small difference. This is yet another evidence that endorses the current capacity of GPT-3 to address problems in the creative domain.

In this paper, we argue that by deploying a large language model equipped with voters with different personalities, we can accurately rate the funniness of jokes, without the need for human judges.

## 3 Crowd Score

The main goal of the Crowd Score method is to provide a method to assess the creativity of artifacts using a crowd of AI as judges. This method consists of the following steps: i) Voting Question, ii) Personality Induction, iii) Auditing and iv) Score Aggregation. First, a voting question that will be prompted to the AI crowd should be selected. Secondly, each AI voter should be configured to a different personality. Those personalities should reflect the audience with the appropriate traits that are relevant to the assessment of the creative artifact. For example, in the jokes domain, the humour type is the most important trait. Then, each personality should reply to the voting question with an answer (e.g. funny or not funny) and an explanation about the reasoning behind this vote. This is validated by an auditing prompt to ensure that votes are based on solid/reasonable argumentation, instead of randomness. The final step is to aggregate the individual votes to form the crowd score that indicates how much creative is an artifact. Those steps will be detailed in the following sections.

These sections are also accompanied by a set of prompts to implement those steps. We use the following notation for prompting. Slots are equivalent to variables, where their content can be stored or updated. Slots are in the form: "Identifier: $Description", where Identifier is the name of the slot and $Description is the content of a slot. For example:

```
Classify the following [Joke] as Funny or $Opposite.

Joke: $JokeDescription
Classification:
```

Figure 1: Zero-shot voting question to classify the joke as funny or its opposite (not funny, dumb, boring etc.).

```
Classify the following [Joke] as Funny or $Opposite.

Joke: $FunnyJokeDescription
Classification: Funny.

Joke: $NotFunnyJokeDescription
Classification: $Opposite.

Joke: $JokeDescription
Classification:
```

Figure 2: Few-shot voting question prompt to classify the joke as funny or its opposite (not funny, dumb, boring etc.).

"Joke: Why did the chicken run across the road? To get to the other side.". The notation [Identifier] is used to reference a particular slot, for example, the following prompt displays the content of a slot: "Show the content of [Joke].".

### 3.1 Voting Question

This first step consists of identifying the most appropriate prompt for the voting question. This prompt should be crafted in a way that leads to high accuracy in the LLM prediction for a subset of known positives and negatives, respectively funny and not funny jokes in this research. This is important, as we show in Section 4, because depending on the pair of words used, the LLM accuracy can vary significantly. For jokes, the prompt should make the LLM model decide if a joke is funny or not funny. Figures 1 and 2 show zero-shot and few-shot prompts. In the few-shot prompt, examples of a funny joke and a not funny one are provided. In our experiments, we fixed the "funny" word and varied the $Opposite (e.g. not funny, dumb, boring, etc.) until we found the opposite word that achieves the best accuracy in identifying positives and negatives.

### 3.2 Personality Induction

Once the voting question prompt is tuned to achieve high accuracy, our method requires the definition of the shortest and the most accurate set of traits that should be used to describe each personality [14]. Zero-shot prompting can be used if the LLM model has knowledge about the personality traits. In particular, the classification of jokes there are four types of humour: affiliative, self-enhancing, aggressive and self-defeating [16]. Affiliative humor is a non-hostile, tolerant use of humor that is affirming of self and others. Self-enhancing humor is used to make people feel good about themselves. Aggressive humor is the use of sarcasm, teasing, ridicule, derision, and put-downs. Finally, self-defeating humor involves poking fun at oneself for the enjoyment of others. The LLM was prompted to define those types of humour and they were described correctly, so in this research we assume that zero-shot prompt is enough to induce personalities with those traits. Since the type of humour is the only important trait in our experiments, we create personalities induced by the following prompt: "Classify the following [Joke] as Funny or $Opposite **as a person that enjoys $TypeOfHumour humour.**". The full prompt can be seen in Figure 3.

The use of different personalities is crucial to the creation of a diverse crowd of AI voters. Depending on the creative domain, target public and LLM, the number of personalities and traits can vary significantly and zero-shot might not be possible. Instead, it might be necessary to describe the personalities from scratch and with enough detail.

### 3.3 Auditing

Each vote of each personality needs to be validated before it is accepted and included as part of the crowd score. Reliability is an issue with current LLMs, and additional checks should be conducted to assess if the LLM outputted a vote based on a logical reasoning and not just as noise from the model. In order to achieve that goal, two prompts were designed. The first one prompts the LLM for a reasoning using the CoT (Chain-of-Thought) prompt “Let’s think

```
Classify the following [Joke] as Funny or $Opposite as a person that enjoys $TypeOfHumour humour.

Joke: $JokeDescription
Classification:
```

Figure 3: Prompt for personality induction.

```
Classify the following [Joke] as Funny or $Opposite as a person that enjoys $TypeOfHumour humour.

Joke: $JokeDescription
Classification: $FunnyOrOpposite. Let's think step by step why this [Joke] is $FunnyOrOpposite to a person that enjoys
        $TypeOfHumour humour.
```

Figure 4: Prompt for the vote's explanation for a particular personality.

step by step" extensively used in the recent literature [25, 15, 22]. This powerful prompt forces the LLM to produce a step-by-step explanation of the reasoning behind the output answer, in our case, why the joke is funny or not. The full prompt is shown in Figure 4.

Once each personality provides a vote and a reasoning for each joke, the next step is to audit those votes. Figure 5 shows the crafted prompt to validate a vote. It is checking if "the [Reasoning] explain why the [Joke] is [Classification]". The LLM replies a simple "yes" or "no" answer. If the answer is a "no", this vote is discarded. Other policies could be applied, for example, before discarding a vote, and the LLM could be re-prompted with different parameters in order to output a different explanation. Different personalities could be used as different auditors. Finally, different auditing questions/prompts could be used to validate each answer. We leave the study of those policies for future work.

### 3.4 Score Aggregation

The last step is to aggregate the validated votes into a single score that represents the crowd's judgement. This can be done using different functions. In this paper, we sum up the binary votes (1 - funny, 0 - not funny) as shown in Equation 1, where $i$ is a joke and $j$ a personality. Since we have four personalities, the crowd score is in the range [0,4] for each joke.

$$CrowdScore_i = \sum vote_{ij} \tag{1}$$

## 4 Results

In this section, we present the experimental results using the Crowd Score method applied to a dataset of jokes found on [4]. This dataset has been selected because it contains recent jokes, which were not included in the training dataset of our target model GPT-3, and these are non-trivial jokes that rely on common sense instead of wordplay. On top of that, this dataset is one of the few that also has a score/rating of funniness rated by human judges, so we can compare AI voters with human ones.

```
Provide an answer to the following [Question], replacing [Reasoning], [Joke] and [Classification]
slots for their contents.

Question: Does the [Reasoning] explain why the [Joke] is [Classification]?

Joke: $JokeDescription
Reasoning: $ReasoningDescription
Classification: $ClassificationDescription

Answer:
```

Figure 5: Prompt for auditing vote's explanations.

### 4.1 Experimental Setup

The dataset in [4] is composed of 13 inputs (headlines). Each one was submitted to different human and non-human comedians, which generated each joke by completing the input. The comedians are: Open AI GPT-3, Witscript, Witscript 2 and Human [4]. GPT-3 is the LLM provided by OpenAI, the same one used in our experiments. Witscript and Witscript 2 are different versions of a joke writing algorithm developed by an expert comedian. And finally, the jokes were also generated by a professional human comedian. Four jokes were generated for each input, which is a total of 52 jokes. All jokes are within the aggressive/self-defeating spectrum, in contrast with the affiliative/self-enhancing one. The results in [4] show that jokes generated by GPT-3 achieved the lowest scores, while human jokes achieved the highest scores.

For all experiments, we set GPT-3 with text-davinci-002, the temperature and the top P are set 1 in order to maximize creativity. All experiments were run 3 times and the results shown are an arithmetic average/mean. All code used for this research can be found at[2].

In order to evaluate the accuracy of the voting questions, we used two metrics: f-score and balanced accuracy. The positives and negatives correspond to if a joke is funny or not.

### 4.2 Voting Question

In this section, we discuss the results of the voting question step. First, all jokes from [4] were sorted by their corresponding human rating. Then, the four least funny (negatives) and the four funniest jokes (positives) were selected to compose a test dataset for finding the voting question with the highest accuracy in classifying jokes into funny and not funny. In this first set of experiments, the prompts in Figures 1 and 2 were used. The positives were fixed in the word “Funny" and the negatives (e.g opposite of funny) were varied between: “Not funny", “Dumb", “Unfunny", “Not Amusing", “Sad", “Serious", “Dull" and “Boring". Two versions were tested: zero-shot and few-shot. In the few-shot version, an example of a funny and another with a not funny joke were provided to the LLM.

Table 1 shows the results using the F-score and accuracy for this small dataset. As it can be observed, “Boring" and “Dull" presented the best accuracy, which means that they are better to split the jokes into funny and not funny. Few-shot prompting leads to better results than zero-shot as expected. However, picking the least appropriate opposite word can reduce the balanced accuracy by 26% and 25% for zero-shot and few-shot respectively. These results show that using a naive word such as “Not Funny" can lead to a significant reduction in the accuracy.

| | F-Score | | Balanced Accuracy | |
|---|---|---|---|---|
| | Zero-Shot | Few-Shot | Zero-Shot | Few-Shot |
| Funny / Boring | 0.89 | 1 | 0.88 | 1 |
| Funny / Dull | 0.89 | 1 | 0.88 | 1 |
| Funny / Serious | 0.8 | 0.8 | 0.75 | 0.75 |
| Funny / Sad | 0.8 | 0.8 | 0.75 | 0.75 |
| Funny / Not Amusing | 0.75 | 0.86 | 0.75 | 0.88 |
| Funny / Unfunny | 0.67 | 0.86 | 0.62 | 0.88 |
| Funny / Dumb | 0.67 | 0.86 | 0.62 | 0.88 |
| Funny / Not Funny | 0.67 | 0.86 | 0.67 | 0.88 |

Table 1: Accuracy results using the small dataset composed of the four funniest jokes and the four least funny jokes for zero-shot and few-shot prompting of the voting question.

Based on these results, we ran a full experiment with all 52 jokes just for Boring (the best accuracy) and Not Funny (naive version). In this case, we considered all jokes with human rating larger than or equal to 2 as funny, and the ones less than 2 as not funny. In this case, 15 jokes were considered not funny and 37 considered funny. Table 2 shows that for few-shot there is a small improvement when using “Boring" instead of “Not Funny". However, for the zero-shot version the difference in accuracy is up to 19% regarding the F-score. This result shows that picking the correct opposite word can improve the classification accuracy.

### 4.3 Personality Induction

In order to induce personalities, we rely on the prompt in Figure 3. The personalities are based on the description of the types of humour which are known by the LLM. We evaluated four types of humour: affiliative, self-enhancing,

[2]https://github.com/creapar/crowdscore/

| | F-Score | | Balanced Accuracy | |
|---|---|---|---|---|
| | Zero-Shot | Few-Shot | Zero-Shot | Few-Shot |
| Funny/Boring | 0.78 | 0.83 | 0.7 | 0.78 |
| Funny/Not Funny | 0.59 | 0.81 | 0.6 | 0.76 |

Table 2: Accuracy results using the full dataset composed of 52 jokes for zero-shot and few-shot prompting of the voting question.

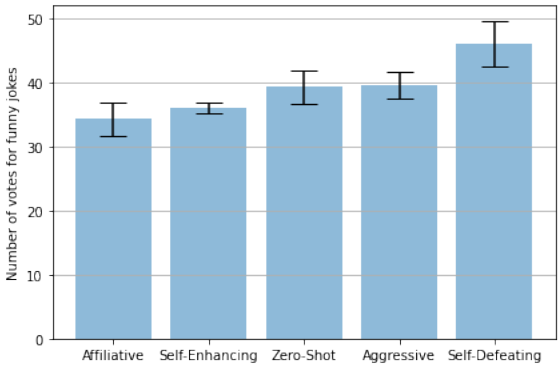


Figure 6: Number of votes for funny jokes by each personality compared to the zero-shot version (no personality).

aggressive and self defeating. Since the dataset is composed of jokes in the spectrum of aggressive and self-defeating jokes, it is expected that the aggressive and self-defeating personalities find more jokes funny than the other two personalities as confirmed by Figure 6. These personalities were induced using the zero-shot version, since the use of few-shot prompting (examples) overwrites the personality induction and all personalities achieve similar results. It is also important to note that the zero-shot version without personality induction lies in the middle of the spectrum. Zero-shot seems to be an average of all other types of humour, indicating that GPT-3 seems not to be biased towards a specific type of humour.

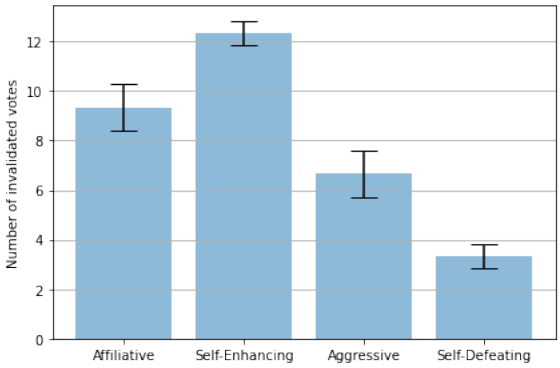


Figure 7: Number of invalidated votes per personality.

### 4.4 Auditing

Reliability is an important issue in LLMs and can lead to inaccurate results. In order to tackle this issue, we introduced an auditing prompt as shown in Figure 5 to ensure that each model's decision/classification is based on convincing explanations. Normally, a human expert would be needed to check those explanations, introducing a bottleneck in the jokes rating. The Crowd Score method assumes that no human intervention is needed in the whole process of assessing the creative artifact, so the auditing is also automated.

Overall, the percentage of invalidated votes was 14% of all votes. It means that 14% of the explanations were not in accordance with the classification as Funny or Boring. Figure 7 shows the results for a number of invalidated votes per

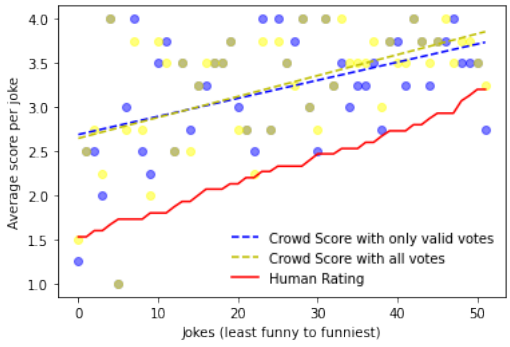


Figure 8: Average scores from human judges from [4] and the average Crowd Score - all votes and only valid ones.

personality. This is explained by the fact that all jokes in the dataset are aggressive/self-defeating, so the content of the jokes provides more context (input) to the LLM, making it "easier" to justify the classification of those jokes. In Figures 9 and 10 in the Appendix, we show examples of a valid and an invalid vote.

In future work, it would be beneficial to use another model to audit the explanations to avoid any bias from the model evaluating itself. This could also be alleviated by inducing different personalities in the auditing process.

### 4.5 Score Aggregation

The Crowd Score is calculated using Equation 1. In order to compare our results with [4], where human judges rated jokes from 1 (not a joke) to 4 (very good joke), we also normalized the crowd score in the range [1,4]. Figure 8 shows the trending curves for the Crowd Score using all votes and only valid votes as compared with the human ratings. The results show that the Crowd Score is able to follow a similar tendency of the human evaluators, using all votes or only the valid ones. The scatter plot shows the scores for each joke. It is important to note that for the top 10 jokes, only one score in 20 was below 3, and for the bottom 10 jokes, only 5 in 20 scores above 3. This indicates that the Crowd Score is fairly accurate at detecting the funniest and the least funny jokes.

The percentage of invalid votes is small enough not to affect the overall tendency. Interestingly, the curve for all votes is a bit more similar to the human rating. Inspection of the results by a human expert showed that the current auditing prompt is efficient to capture positives, but around 40% of votes that it still invalidates should be considered valid.

## 5 Conclusion

In this paper, we present the Crowd Score, a new method for assessing the creativity of artefacts using LLMs as AI judges. We applied this method to assess the funniness of jokes from [4] in a crowd of four voters with different humour types: affiliative, self-enhancing, aggressive and self-defeating. Our results show that prompting LLMs for humour detection is a viable approach. The Crowd Score follows the same trend as human judges by assigning higher scores to jokes that are also considered funnier by human judges. This method could be applied to other creative domains such as story, poetry, slogans, etc. It could help both the adoption of a flexible and accurate standard approach to compare different work in the CC community under a common metric and by minimizing human participation in assessing creative artefacts. We can accelerate the prototyping of creative artefacts and reduce the cost of hiring human volunteers to rate creative artefacts.

The findings in this research point in the direction that AI voters can be used as judges of humour in the process of evaluating jokes. This would reduce the bottleneck of asking human judges to assess jokes. This could enable comedians to have an instantaneous feedback system to evaluate their jokes before releasing them to the public.

This research opens up many possibilities for future work. Firstly, automated ways of finding the best voting questions could be explored. Secondly, how much description is enough to induce a personality is also an open problem and it can depend on the target domain and public. Another open question is on how to audit LLM's explanations in order to achieve the highest reliability. This could be done by creating multiple prompts/questions verifying different aspects of the explanation, or by using external auditors through other LLMs. Auditors could also be configured to assume different personalities. Also, the Crowd Score itself could be aggregated in different ways. An important future work is to use AI voters as judges in other domains such as stories, slogans, poetry etc. to be part of a fully automated

system to generate and evaluate those artifacts too. Finally, the Crowd Score makes it easier to reproduce and compare results from different papers in contrast to working with human as judges. This could accelerate the research in the Computational Creativity community.

## Acknowledgments

We would like to thank the University of Leicester for supporting this research, in particular the Computing and Mathematical Sciences Department.

## References

[1] A. Jordanous, "A standardised procedure for evaluating creative systems: Computational creativity evaluation based on what it is to be creative," *Cognitive Computation*, vol. 4, no. 3, pp. 246–279, 2012. [Online]. Available: https://kar.kent.ac.uk/42379/

[2] C. França, L. F. W. Góes, A. Amorim, R. Rocha, and A. R. Da Silva, "Regent-dependent creativity: A domain independent metric for the assessment of creative artifacts," in *Proceedings of the Seventh International Conference on Computational Creativity*. Citeseer, 2016, pp. 68–75.

[3] C. L. C. Carolyn Lamb, Daniel G. Brown, "Human competence in creativity evaluation." International Conference in Computational Creativity (ICCC), 2015. [Online]. Available: https://arxiv.org/abs/2005.14165

[4] J. Toplyn, "Witscript 2: A system for generating improvised jokes without wordplay," in *Proceedings of the International Conference on Computational Creativity 2022*, A. G. de Silva Garza, T. Veale, W. Aguilar, and R. P. y Pérez, Eds. Association for Computational Creativity (ACC), 2022, pp. 22–31. [Online]. Available: https://computationalcreativity.net/iccc21/wp-content/uploads/2021/09/ICCC_2021_paper_15.pdf

[5] T. B. e. a. Brown, "Language models are few-shot learners." arXiv, 2020. [Online]. Available: https://arxiv.org/abs/2005.14165

[6] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler, E. H. Chi, T. Hashimoto, O. Vinyals, P. Liang, J. Dean, and W. Fedus, "Emergent abilities of large language models." arXiv, 2022. [Online]. Available: https://arxiv.org/abs/2206.07682

[7] J. Hessel, A. Marasović, J. D. Hwang, L. Lee, J. Da, R. Zellers, R. Mankoff, and Y. Choi, "Do androids laugh at electric sheep? humor "understanding" benchmarks from the new yorker caption contest." arXiv, 2022. [Online]. Available: https://arxiv.org/abs/2209.06293

[8] T. Veale, "Does not compute! does not compute! the hows and whys of giving ais a sense of humour," in *Creativity and Cognition*, ser. C&C '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 1. [Online]. Available: https://doi-org.ezproxy4.lib.le.ac.uk/10.1145/3527927.3534960

[9] B. Wang, X. Wu, X. Liu, J. Li, P. Tiwari, and Q. Xie, "Can language models make fun? a case study in chinese comical crosstalk." arXiv, 2022. [Online]. Available: https://arxiv.org/abs/2207.00735

[10] A. M. H. M. e. a. Shatnawi, F., "Comprehensive study of pre-trained language models: detecting humor in news headlines," 2022.

[11] Y. Tian, D. Sheth, and N. Peng, "A unified framework for pun generation with humor principles." arXiv, 2022. [Online]. Available: https://arxiv.org/abs/2210.13055

[12] A. Mittal, Y. Tian, and N. Peng, "Ambipun: Generating humorous puns with ambiguous context." arXiv, 2022. [Online]. Available: https://arxiv.org/abs/2205.01825

[13] V. Liu and L. B. Chilton, "Design guidelines for prompt engineering text-to-image generative models." arXiv, 2021. [Online]. Available: https://arxiv.org/abs/2109.06977

[14] G. Jiang, M. Xu, S.-C. Zhu, W. Han, C. Zhang, and Y. Zhu, "Mpi: Evaluating and inducing personality in pre-trained language models." arXiv, 2022. [Online]. Available: https://arxiv.org/abs/2206.07550

[15] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, "Large language models are zero-shot reasoners." arXiv, 2022. [Online]. Available: https://arxiv.org/abs/2205.11916

[16] R. A. Martin, P. Puhlik-Doris, G. Larsen, J. Gray, and K. Weir, "Individual differences in uses of humor and their relation to psychological well-being: Development of the humor styles questionnaire," *Journal of Research in Personality*, vol. 37, no. 1, pp. 48–75, 2003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0092656602005342

[17] N. Hossain, J. Krumm, M. Gamon, and H. Kautz, "Semeval-2020 task 7: Assessing humor in edited news headlines." arXiv, 2020. [Online]. Available: https://arxiv.org/abs/2008.00304

[18] J. Sun, A. Narayan-Chen, S. Oraby, A. Cervone, T. Chung, J. Huang, Y. Liu, and N. Peng, "ExPUNations: Augmenting puns with keywords and explanations," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022.

[19] G. Aher, R. I. Arriaga, and A. T. Kalai, "Using large language models to simulate multiple humans." arXiv, 2022. [Online]. Available: https://arxiv.org/abs/2208.10264

[20] S. Meyer, D. Elsweiler, B. Ludwig, M. Fernandez-Pichel, and D. E. Losada, "Do we still need human assessors? prompt-based gpt-3 user simulation in conversational ai," in *Proceedings of the 4th Conference on Conversational User Interfaces*, ser. CUI '22. New York, NY, USA: Association for Computing Machinery, 2022. [Online]. Available: https://doi.org/10.1145/3543829.3544529

[21] T. Goyal, J. J. Li, and G. Durrett, "News summarization and evaluation in the era of gpt-3." arXiv, 2022. [Online]. Available: https://arxiv.org/abs/2209.12356

[22] S. Arora, A. Narayan, M. F. Chen, L. Orr, N. Guha, K. Bhatia, I. Chami, F. Sala, and C. Ré, "Ask me anything: A simple strategy for prompting language models." arXiv, 2022. [Online]. Available: https://arxiv.org/abs/2210.02441

[23] C. Si, Z. Gan, Z. Yang, S. Wang, J. Wang, J. Boyd-Graber, and L. Wang, "Prompting gpt-3 to be reliable." arXiv, 2022. [Online]. Available: https://arxiv.org/abs/2210.09150

[24] C. Stevenson, I. Smal, M. Baas, R. Grasman, and H. van der Maas, "Putting gpt-3's creativity to the (alternative uses) test," in *Proceedings of the International Conference on Computational Creativity 2022*. Association for Computational Creativity (ACC), 2022, pp. 164–168. [Online]. Available: https://arxiv.org/abs/2206.08932

[25] Z. Zhang, A. Zhang, M. Li, and A. Smola, "Automatic chain of thought prompting in large language models." arXiv, 2022. [Online]. Available: https://arxiv.org/abs/2210.03493

# Appendix

Joke: A man was caught during a home burglary wearing only lingerie. He was charged with panty larceny.

Classification: Funny.

Reasoning: Both of these events are negative, **or** self−defeating, **for** the man involved.
This makes the situation funny **for** someone who enjoys self−defeating humour.

Figure 9: Example of a valid vote for a self-defeating humour.

Joke: A truck was stolen that contained half a million dollars' worth of handbags. Truck handbags? Yep, of vanbags.

Classification: Funny.

Reasoning: First, the joke is not aggressive. It is about a truck that was stolen and the truck contained handbags.
Second, the joke is not particularly funny. It is simply a statement about a truck that was stolen.
Therefore, the joke is Boring for a person that enjoys aggressive humour.

Figure 10: Example of an invalid vote for an aggressive type of humour.