oddsidemargin has been altered.
textheight has been altered.
topmargin has been altered.
textwidth has been altered.

The required page layout has been changed.

Please set up your document as in the example skeleton thesis document. Do not change the page layout, or include packages like geometry, savetrees, or fullpage, which change it for you.

We're not able to reliably undo arbitrary changes to the style. Please remove the offending package(s), or layout-changing commands and try again. If you can't figure out the problem, try adding your LaTeX code a part at a time to the example document.

Enhancing humour in Large Language Models

George Karabassis

MInf Project Report
Masters of Informatics
School of Informatics
University of Edinburgh

2025

Abstract

Humour generation remains a challenging task for large language models (LLMs), mainly due to its subjectivity. This study investigates the effectiveness of fine-tuning open-source LLMs using social media data, primarily from Reddit, to enhance their ability to generate humorous content. By collecting Reddit prompts and responses, three models—DeepSeek R1, LLaMA 3, and Gemma 2.0—were fine-tuned and evaluated through both the author of this paper and six automated LLM-based judges. The evaluation included an in-depth analysis that fine-tuned models often outperformed their base counterparts, especially in replicating short and human-like responses to given prompts. More specifically, Gemma demonstrated the largest performance gain, with an approximate 27% improvement based on individual humour evaluation across various LLM-instructed humour style personalities, and notably generated more than twice the amount of successful humorous responses compared to other LLMs. Gemma also exhibited the highest alignment with human response ratings, outperforming other LLMs in approximating human-like humour judgments. Challenges remain with the liability of LLM judges and the performance of the fine-tuned LLMs. The findings indicate the potential of fine-tuning to improve humour generation in LLMs and suggest future directions involving larger datasets, more diverse models, and a further development of LLM judges to improve the automatic evaluation procedure.

Research Ethics Approval

This project obtained approval from the Informatics Research Ethics committee. Ethics application number: ???
Date when approval was obtained: YYYY-MM-DD
The participants' information sheet and a consent form are included in the appendix.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*George Karabassis*)

Acknowledgements

I would like to express my sincere gratitude to my supervisors, Dr. Björn Ross and Dr. Xue Li, for their invaluable guidance and support throughout this project. I began this journey with little prior knowledge in the field, but through their assistance, I expanded my knowledge and abilities in conducting research independently. Despite the many challenges I faced over the past year, my supervisors continually pushed me to improve and persevere, for which I am very thankful.

Table of Contents

Chapter 1

Introduction

1.1 Motivation

Large Language Models (LLMs), like GPT-4o, are increasingly being integrated into a wide range of daily tasks, including learning [67], entertainment, programming, etc [98]. One area that is understudied is experimenting with an LLM to respond to social media text-based posts with human-like humour [44]. The LLM responses end up being formal and "robotic"; often missing the nuance [74]. While there is some research on evaluating how funny LLMs are perceived by participants, they are still subjective to the audience, especially when considering a culturally diverse group of participants [17]. Humour plays a vital role in building social rapport, relatability, and engagement. A more humorous LLM can feel more human-like and approachable, which is especially important in informal online settings like social media, where wordplay and sarcasm often dominate user interactions [89].

It's crucial to emphasise the role of social media in daily life, as the majority of people actively engage with these platforms. [6]. Users use humour in their responses to social media posts [26]. However, humour can be risky: some responses may be seen as offensive to some online users [105]. This can lead to several notable issues such as discrimination, diminishment and in some cases, social conflicts.

1.2 Proposed Solution

To address these challenges, this project focuses on the following research questions:

R1. Does fine-tuning improve the humour generation capabilities of an LLM?

R2. Can a fine-tuned LLM produce funnier responses to social media posts than humans?

Three LLMs were fine-tuned with the objective of generating humorous responses to social media-style prompts. The performance of the three fine-tuned LLMs were compared and the model with the most humourous generated responses was designated as FunnyGPT. FunnyGPT takes as input a prompt from a social media context (a.k.a, Reddit post) and produces neutral funny responses which can then be used by the user. FunnyGPT will be designated by fine-tuning Llama 3.0, DeepSeek R1, Gemma 2.0 with their zero shot (state of the art) equivalent. Given that the evaluation of humorous text is subjective to the reader, five different humour judge LLMs evaluators instructed based on the four different humour style personalities [59]. Each

judge and the project author compared four responses: the original human-written comment from Reddit and three generated responses; one from each fine-tuned model.

The findings show that fine-tuning provides a significant improvemnet in humour generation making it comparable to human generated humour. Overall, Gemma has shown a significant improvement with humour generation, while DeepSeek has seen the worst improvement.

1.3 Structure

This report is organised into six main chapters:

- **Background**: Analysis of current research, similar solutions, and their limitations.
- **Methodology**: fine-tuning decisions, datasets used, dataset pre-processing, LLM judges, and evaluation strategy.
- **Implementation**: Development of the dataset pre-processing and fine-tuning procedures.
- **Evaluation**: Comparing three fine-tuned LLMs with original human responses and their zero-shot responses and analysis of the results.
- **Conclusion**: Project achievements, limitations and suggestions for future work.

Overall, the project starts with the Background chapter where critical analysis of previous research work is done and identified current research gaps. Subsequently, the Methodology chapter contains the selection, pre-processing, and fine-tuning of three LLMs using humour-based datasets. The evaluation chapter presents a comparison between fine-tuned models, their zero-shot counterparts, and human-written responses, assessed through both the author of this paper and LLM-based evaluators. At the end, future work, achievements and future work are discussed.

Chapter 2

Background

2.1 Humour Theory

Humour is studied across multiple fields, including Psychology, Sociology and Literature. In recent years, it has gained significant attention in Artificial Intelligence (AI) [25], especially when working with LLMs. While humour is highly subjective, several studies have analysed humour into several categories.

2.1.1 Categories of Humour

One famous study concluded that humour can split into four different categories: affiliative, self-enhancing, aggressive, and self-defeating [59]. Each category is further analysed:

- **Affiliative**: Positive, social humor used to build relationships. Mostly used between close friendships for "amusement" purposes.
- **Self-Enhancing**: Internal humor used for coping and staying positive. This is common in posts where users laugh off their own misfortunes.
- **Aggressive**: Humor used to mock or put others down using elements of sarcasm, teasing and ridicule. Common in places such as X (formerly Twitter) or Reddit.
- **Self-Defeating**: Humor used to make fun of oneself to be liked by the community.

These four humour categories have been studied among a group of young adults on how they are used [43]. These four humour styles have been observed in online situations including from users' comments posted in online platforms.

Further studeis have provided deeper analysis and split humour further [64]:

- **Superiority Theory:** Feeling superior to others, often by mocking mistakes, misfortune, or ignorance (e.g., slapstick, sarcasm).
- **Relief Theory:** Release for built-up tension, stress, or social anxiety (e.g., dark humor, nervous jokes).
- **Incongruity Theory:** Unexpected or illogical contrasts, where the punchline subverts expectations (e.g., absurd jokes, wordplay).

- **Violation Theory:** Social norms or rules broken in a non-threatening way (e.g., edgy jokes, irony).
- **Sociological Perspectives:** Cultural values, group identity, and power dynamics, shaping social interaction (e.g., political satire, inside jokes).
- **Linguistic Dimensions:** Language structure, ambiguity, and phonetics, such as puns, double meanings, or misinterpretations (e.g., dad jokes, wordplay).

These categories will each include a different type of humour, e.g. "That tiny human feels like an ant at the base of 33 fire-breathing skyscrapers!". The humorist in their sentence has a superior position against the "tiny human". Another category, the Relief Theory highlights a relief humorous expression of an event e.g. "Glad I'm not the one fueling this rocket; I can barely keep my car's tank full!".

2.1.2 Humour used in Social Media

Humour is often used in social media, especially in the comment sections of the original post (often referred to as "thread") [28]. Sarcasm is the most common way for users to respond in a humorous way to a social media post [88]. Usually, users will respond with sarcasm when the original poster (OP), including, but not limited to disruption, political problems, statements from high-profile people etc [72]. For example, a social media post with the content of "Oh great, the internet is down again. Guess I'll have to bond with my family now—thanks for the forced quality time!", is based on an "internet disruption" event. Some notable reasons for users responding sarcastically include:

- **Attention**: Sarcasm and humour grab more attention and engagement than pure frustration [51].
- **Social Acceptance**: Highlight a frustration in a more socially acceptable way.

Self-deprecating Humour, is another way for users to express their feelings. According to studies, self-deprecating humor is used on social media to gain more attention and engagement [79]. This type of humour can enable the community to engage more sympathetically with the OP. Some drawbacks include using this humour in improper situations, such as if the OP indicates a crisis, it may be perceived as offensive or diminishing.

Another kind of humour which falls outside of the scope of this project is memes. Memes are also very prevalent in responses, which can cover everyday situations, including events [26]. Memes can be displayed in the form of text, visual, animation, etc.

2.2 Computational Humor

2.2.1 History of Computational Humour Generation

The field of Computational Humour Generation is a subfield of Computational Linguistics and AI. It begins with the first publication paper from a study done at the University of Edinburgh [8], describing a computer program, JAPE (Joke Analysis and Production Engine) which generates "question-answer-type" puns from a non-humorous lexicon. This system was developed further to help children with communication difficulties [95]. HAHAcronym was another notable example with the goal of converting existing acronyms into new ones by retaining the comical

effect, e.g. “Association for Computing Machinery” to “Association for Confusing Machinery.” [86].

During the 2010s and before the introduction of Transformer Neural Networks [52], Machine Learning concepts such as supervised learning and Recurrent Neural Network (RNN), were commonly used. In a study Feed-Forward Neural Networks (FFNN) and RNNs were used to rank a selected poll of tweets based on how “funny” they are [73]. However, the model introduced in this study is unable to capture the full context of each tweet, something that the Transformer Neural Networks solve.

With the introduction of Transformer-based architectures marked a turning point in the development of computational humour. Thanks to the emergence of LLMs, several studies have been conducted to evaluate the understanding and the generation of humor.

2.2.2 LLMs Used for Computational Humour

LLMs have played a significant part in the development of this field [3], thanks to the Transformer Neural Networks which can capture more context of a given text. Notably, most research done currently is to assess how well LLMs generate and understand humours by evaluating the benchmarks of some of the most commonly used LLMs, including GPT-3 [61], Llama [44], PaLM and BERT [48]. However, these studies are mostly limited to evaluating the humour capabilities of these models in the pre-training stage (i.e. zero-shot), with no fine-tuning have been implemented to improve the humour generation. From those, GPT-3.5 has been studied more than the other models, despite GPT-4o being more advanced. LLM fine-tuning has also been used to improve the ability of AI Humour Generation where studies have shown that fine-tuning yields a significant improvement with Humour generation [102].

GPT-3 Series

GPT-3 is the most studied LLM for producing AI-generated humour as of 2024. According to a few studies, GPT-3 is has an advantage at reproducing learnt humorous statements but lacks the ability to generate novel and original jokes [44]. One study from the University of Southern California found that humour generated by GPT 3.5 is perceived as funnier than human-generated humour [32]. This indicates that GPT 3.5 would be a useful LLM for the evaluation in this project. Given that GPT-3 is now a predecessor of GPT-4 series, and GPT-3 isn’t available yet for fine-tuning [20], it wasn’t used as part of the evaluation process.

GPT-4 Series

Given that GPT-4 were released recently, on 14th of March 2023, there isn’t much research done to confirm its ability to generate humour. In one study done in the Edinburgh Festival in 2023 [62], comedians rated LLMs (including GPT-4) with a mean of 54.6% in the Creativity Support Index (CSI) with the ability of LLMs to produce original humour. Furthermore, in another study, GPT-4 showed promising results regarding translating humour from Arabic to English, focusing its ability to use slang words [2]. Despite GPT-4 being more advanced than GPT-3, it still demonstrates limitations in its ability in producing original humour [57].

Despite GPT-4 showing limitations in humour generation, GPT-4 consistently outperforms other LLMs in humour comprehension tasks, when given a structured prompt [92].

Llama 3.0

One study has shown that fine-tuning Llama 3.0 can generate funnier statements than GPT-4 [99]. LLama is often chosen for fine-tuning research applications, due to the smaller parameter size, 8 Billion (8B), making it more efficient for customization. Additionally, Llama is based on the Low-Rank Adaptation (LoRA) method [41] which allows updating a subset of parameters during fine-tuning making it less expensive computationally and requires less memory [103]. Based on these findings, Llama is a good candidate for fine-tuning.

DeepSeek R1

A recently published study on the 31st of March 2025, proved that DeepSeek performs very well with humour analysis and evaluation, outperforming several other LLMs, including Mistral. The study conducted at the University of Cambridge compared 13 different LLMs, including Mistral, Gemini and Llama [29]. Furthermore, the study showed that DeepSeek produces a noticeable longer outputs compared to the other LLMs (like Gemini), making it less efficient, compared to other models, such as Gemini. Given that DeepSeek is among the newest LLMs, further research on its capabilities remains limited at the time of this writing.

2.2.3 Datasets

Several datasets are available containing humouristic statements, including datasets containing humouristic statements from social media platforms, such as X and Reddit.

- **Reddit**: A 10-year collection of jokes shared by Reddit users in the r/Jokes channel [97].
- **Reddit Sarcasm**: Sarcastic comments posted by users on Reddit from 2009-2016 [78].
- **Twitter Sarcasm**: A database containing sarcastic tweets [70].
- **Sarcastic Statement**: Over 1.3 million sarcastic statements [19].

Additionally, given that the responses of a prompt may be continuous, where, for instance, user A replies to an original post and user B subsequently responds to User A, a chain-based database is more suitable for fine-tuning and evaluation:

- **Humour Chain**: Thread-based dataset which includes data from the most upvoted user prompts from Reddit with their responses [107].
- **Twitter Threads**: Contains a list of datasets with threads of tweets taken from X (formerly known as "Twitter"). This dataset includes the total number of retweets, likes and total responses per tweet. A higher number of likes and responses could mean that the original tweet had humorous elements [33].
- **Sarcastic Reddit Comments**: Derived from the Self-Annotated Reddit Corpus (SARC), is a substantial resource for sarcasm detection research. It comprises approximately 1.3 million sarcastic comments and parent comments [16].

The databases are a combination of data gathered from social media, including Reddit and Twitter. To produce more accurate results, the datasets containing social media comments should be used. It's important to note that data within the dataset may contain offensive language that should not be used. While this is beyond the scope of this project, a basic cleansing of the datasets used is important to remove prompts containing offensive language.

Datasets humour bias

Datasets including the *Sarcastic Reddit Comments* includes several meatadata per comment, including the total number of downvotes and upvotes. In social media settings, it's a well-known phenomenon that the audience will interact with comments they perceive as "funnier". According to a study done on Reddit, comments receiving more than 200 upvotes in total are most likely to be perceived as "funny" [96]. While this may not always be the case, it's very possible for comments with a lower count of total upvotes, to have the potential to appear humourous to the readers. The time of each comment being posted plays a role on how many users got the chance to read it, as the comment may be prioritised less compared to other comments [85]. Additionally, peak times of social media usage is also an important factor.

Due to the subjective nature of humour, challenges arise with the attempt to mitigate this problem. There are several potential ways that would be used to mitigate this:

- **Prompt**: "I have to be the last one to come to this realization"
- **Comment**: "Well, GTA Online definitely has plenty of children."

2.2.4 Existing Fine-Tuned LLMs

While most fine-tuned LLMs are specifically for Humour detection, there are some which focus on generating humorous content. While the humorous detection isn't related to the scope of this project, a suitable fine-tuned LLM, or a combination of fine-tuned LLMs could be used to assist with the evaluation process of this project.

Fine-Tuned GPT-2

A fine-tuned LLM was created based on GPT-2 to generate programming-based jokes [7]. This model was fine-tuned with a relatively small dataset, comprising only 220 distinct entries. While this may not be an ideal fine-tuned LLM, but a good foundation base indicating that a smaller dataset can be used to fine-tune an LLM to produce desirable humour on a specific topic, a.k.a, programming.

Fine-Tuned BERT for joke classification

This is an example of a fine-tuned LLM which was created to classify sentences as to whether they are funny or not [46]. The fine-tuned BERT model is able to distinguish between funny and not funny sentences. Joke sentences show an "X" pattern in attention heatmaps, where the setup includes the context and the punchline are different than expected. The model focuses on tokens connecting these elements, as in "Why did the scarecrow win an award? Because he was outstanding in his field," revealing its ability to detect humour-critical relationships. This model has been fine-tuned to perform as a classification model by distinctly responding with a "funny" or "not funny" labels, limiting the idea of using this model as a judge to compare different responses.

Fine-Tuned BERT model to detect sarcasm

Sarcasm is the main type of humour that most people will see in social media responses. This fine-tuned LLM improves the ability of BERT to detect whether a given sentence is sarcastic or not [34]. This repository offers three fine-tuned sarcasm detection models (BERTweet and RoBERTa-large) and datasets: SAD (2,340 manually labeled tweets), S3D-v1 (100,000 tweets labeled by BERTweet), and S3D-v2 (100,000 tweets labeled by an ensemble). The fine-tuned BERTweet model achieved an F1-score of 78.39 on the SARC dataset and 78.87 on the S3D dataset, showing a relatively strong sarcasm detection performance. It outperformed BERT

and RoBERTa-large, especially in tweet-based and mixed-domain datasets [71]. While this model works similar to the previous model, the dataset used is focused primarily on sarcastic statements.

2.2.5 Existing AI humour generation applications

Several real-world applications have been developed based on fine-tuning LLMs. Most of these applications, haven't disclosed the steps taken to fine-tune the models nor the dataset used.

AI Comedian

AI Comedian is an online chatbot dedicated to generating humorous text based on a topic the user will input [22]. The technical details of how this chatbot was created are not disclosed, but it's possible that an existing LLM was used which was fine-tuned. While this is a good starting point, the responses are very long, whereas a typical comment with humouristic intent will mostly be compromised of around 13 tokens (words) [82].

Punchlines.ai

Punchlines is an online chatbot which allows the user to insert an input prompt [1]. The model will output a humorous response. The model uses the GPT-3 and it's fine-tuned with monologue jokes to produce funnier responses. The responses are shorter than those from AI comedians, however, the responses of the model are meant to be a continuation of the input the user has entered. Most of the time, the responses won't simulate a different user commenting on the input. On top of that, some responses contained political elements which may seem uncomfortable to certain groups of people.

FlowGPT: Funny Chat with AI

FlowGPT has a lot of GPTs with different models indicating different characters. Each model is fine-tuned to produce a specific persona. One of them is Funny Chat with AI [23]. The technical details, regarding the dataset used for fine-tuning are not disclosed. After some tests with the chatbot, it can produce short and funny enough responses, however, the responses can be distinguished from human replies. Some responses may not be perfectly understood by the audience, and the response may not be perceived as "funny".

2.3 Challenges of Computational Humour

2.3.1 Limitations

The fine-tuning of LLMs is currently constrained by several technical and practical limitations.

Fine-Tuning limitations While GPT-4 and other models offered by OpenAI are available for fine-tuning, the fine-tuning is limited to prompt engineering and it's not available to be fine-tuned using a custom dataset [69]. This includes GPT-4, Gemini and GPT-3 series. While this poses a limitation on the choice of LLMs to be used, other LLMs have shown to have comparable humour generation capabilities, including DeepSeek R1, Llama and Gemma, which are available for fine-tuning.

Dataset humour diversity Dataset scarcity is one reason AI humour generation is challenging [39], meaning that careful consideration of database choice is needed. The dataset needs to contain a broad range of humorous expressions across different cultural backgrounds. Humour

can be deeply integrated in sociocultural context; what might be considered funny in one subculture, can be offensive in another subcultural setting [14].

Limited cultural humour understanding Further reasons include a limited understanding of diverse cultural humour, for example, a funny statement originating from China, may not be perceived as "funny" in the UK. Other limitations include specific humour patterns [44]. The more humouristic patterns and subcultural settings the dataset contains, the larger it requires to be to ensure a sufficient representation of each pattern.

Limited resources LLMs often require advanced computational equipment to perform training procedures and these resources may often require a substantial amount of funding to experiment with [37]. Several LLMs are packed with billions of parameters, meaning that fine-tuning with a few thousand of data entries can take up to a few hours to complete. This hinders the ability to use larger datasets with more humouristic patterns.

Restricted LLM availability The available LLMs which are available for fine-tuning, often come with a few billion parameters. Fine-tuning uses backpropagation to update the model's internal weights of the parameters based on the patterns identified with the training data [42]. Models with a higher number of parameters posses a greater ability to capture complex patterns, including diverse form of humour [11]. This require more advanced computational and memory requirements to fine-tune [37]. Consequently, the LLMs selection were limited to a relatively smaller number of parameters to align with the available resources.

Humour limited to trained data Furthermore, most studies confirm that the studied LLMs are limited to producing humorous statements based on learnt patterns and trained data [39]. Additionally, LLMs don't have the ability to understand human feelings yet and they can't generalise on humour as much yet [47].

2.3.2 Research Gap

The study of computational humour using LLMs is still a new research area, meaning that there are significant gaps at the time of writing.

Human-like humour generation LLMs still lack the ability to produce humouristic content that reassembles human-like characteristics [44]. There is limited research done to make LLMs to produce more "human-like" humorous responses. Although there isn't a definitive method for enabling LLMs to generate human-like responses, using datasets containing real examples of human produced humour is a critical step. The dataset must contain a large humouristic chain-humour content fetched from real social media environments. Given the probabilistic nature of the LLMs, there is a limited guarantee that fine-tuned LLMs will be indistinguishable to human-generated humour.

Producing Short-form responses Currently, zero-shot models still lack the ability to produce short responses. It is shown that shorter sentences (punchlines) are perceived as funnier than longer sentences [83]. Given this, this shows that short punchlines posted as comments in social media posts, are perceived as "funnier" to users than longer statements.

Limited to one-liner humour Research to this date is limited to one-liner comments. This isolates the given statement form broader conversational or contextual settings. This limits the "understanding" of social/cultural nuances, resulting LLM-generated responses to be more generalised and lack an in-depth analysis of the given joke [81].

Limited study on fine-tuned models There's a gap on the effectiveness of fine-tuned models with humour generation. While some models, including Llama 3.0, have been studied with fine-tuning [99], there are many open-source LLMs which haven't been evaluated with fine-tuning. This is mostly the case with the newest LLMs, including DeepSeek R1.

2.3.3 LLM fine-tuning methods for human-like responses

Some machine learning techniques have shown to improve LLM responses by producing more natural responses, including Representation Alignment from Human Feedback (RAHP) [54] and Reinforcement Learning [106]. The RAHP method works by identifying the differences in neural representations between preferred and non-preferred responses and incorporating these differences using Low-Rank Adaptation (LoRA). This simplifies the need for rewarding models and produces more human-like responses by directly aligning the model's internal activity patterns with human preferences. On the other hand, the Reinforcement Learning method works by using human participants to rank the output of the model using a rewarding model. The use of KL regularisation keeps the outputs close to the pre-trained model's distribution. Through iterations of human ranking, the model is able to produce more natural and human-like responses.

Several of these advanced methods, including Reinforcement Learning and KL regularisation were not implemented in this paper due to resource constraints. Both RAHP and RLHF require access to large human feedback datasets and computational infrastructure to support iterative model updates, making it beyond the scope of this work.

Chapter 3

Methodology

This section explains the decisions taken including data pre-processing, detoxification of the dataset, selection of LLMs to be fine-tuned using the pre-processed dataset and the choice of LLM judges to be used as part of the evaluation section to assess the humour generation of each fine-tuned LLM.

To maintain consistency, the following terminologies were used in this section:

Social media post: In LLMs, the social media post is referred to as "prompt".

Comment section: In LLMs, the comment section is referred to as "responses".

3.1 Datasets

3.1.1 Dataset Requirements

During the fine-tuning phase, datasets were selected based on the objective of simulating a social media environment. More specifically, using a social media post (prompt), followed by a user comment (response). Challenges involve choosing the right dataset and filtering the data to remove potential toxic content. Given that looking for more in-depth methods of filtering out toxic and offensive content is out of the scope of this project, existing pre-trained models were utilised to perform the initial filtering of toxic content.

Before choosing a closely aligned dataset for this method, several steps need to be followed prior [55]:

- **Relevancy** The content of the dataset aligns closely with the generation of humor on social media comment sections.
- **Format** Dataset format matches with the prompt-response task (input: original poster, output: humour).
- **Trained Data** Verify that the dataset is not included in the pre-training of the selected model (find citation on this).

The first two steps can be taken by manually inspecting each entry of each potential dataset. The last step, can be evaluated using several techniques, to assess the contextual similarity of the input and output context (see 3.3.1).

Given that there were multiple datasets with authentic humouristic content from social media, most were one-liner comments, where the parent comment or original post was not given. There were limited choices for chain-humour datasets available for use.

3.1.2 Dataset Choice

Dataset Name	Abbreviation	Training Rows	Test Rows
Sarcastic Reddit Comments	SRCD	1,300,000	Not specified
Reddit Chain-humour	RCHD	5,840	650

Table 3.1: Summary of the datasets used

SRCD dataset [16] contains the total number of upvotes and downvotes per comment. This dataset has specifically scraped prompt-responses containing the "sarcasm" tag. Sarcasm has shown an increase of engagement, but mainly in areas where the user has expressed frustration or in debate-based group discussions [12]. For this reason, it's important if this dataset is used, to filter out prompts that contain a small number of upvotes or containing high number of downvotes. In social media, on average, a humorous prompt will yield a higher engagement rate [101], which often results with more reactions and in our case, thumbs ups. Additionally, it contains both original prompt and comment content, making it easy to be reformatted. This is a relatively large dataset with nearly 1 million prompt-responses, which can be a benefit for lengthy fine-tuning experiments.

The RCHD [107] dataset contains the most up-voted prompt-responses from multiple humorous-based Reddit subreddits. While this dataset is relatively small (around 6 thousand rows), it's a great choice to mix it with the SRCD to provide more diverse responses.

These two datasets contain a diverse set of prompt-response pairs. Given that both datasets come from the same source (Reddit) and follow the same structure, were both combined to be used for fine-tuning all LLMs used.

Combination of datasets

For the experiments, both SRCD and RCHD were combined to diversify the responses from the fine-tuned LLMs. This is also referred to multi-tasking fine-tuning, where multiple datasets of the same format can be used for fine-tuning. Existing studies show that the fine-tuned LLM can yield better responses by diversifying the outputs [10].

3.2 Pre-processing

Pre-processing steps can have a significant impact on the fine-tuned LLM [66]. Some steps that were taken include:

1. **Handling Incomplete data**: Usually deleting incomplete data is the most effective method, but other techniques such as estimation can be used [75] [double check]. Only entries which contained both prompts and responses were kept in the dataset.
2. **Remove unwanted data**: Data with offensive content were removed to reduce bias.
3. **Shuffle content**: Given that two datasets were merged, the merged dataset was treated as one and the content was shuffled.

These techniques ensured a cleaner and more balanced dataset, which reduces the bias of the LLM. This reduced the likelihood of potentially offending users from subcultural settings.

3.2.1 Dataset Format

Two datasets were merged to be used for fine-tuning the LLMs. For congruency, before merging the datasets, they had to be converted into a standardised format. One data entry had the following structure.

- **Prompt**: Why do the chickens cross the road?
- **Response**: To cross to the other side.

The new formatted dataset was saved before being used for fine-tuning. This made it easier to use the reformatted dataset for multiple LLMs without the need for reproducing the dataset. The prompt and response varied based on the OP and comment.

3.2.2 Dataset Filtering

While the chosen datasets contain responses with high number of votes, given the limitation of resources available to fine-tune the models, only subsets were used from each dataset. The SRCD dataset included the total number of upvotes and downvotes, meaning that the comments with the highest number of upvotes and lowest number of downvotes were kept. Specifically, the following procedure was applied to each dataset to extract the samples to produce the best possible results:

- **Dataset size**: The final dataset consisted of $5,000$ prompt-responses pairs, $2,500$ from each dataset.
- **SRCD Filtering**: The dataset was sorted based on the total number of upvotes and each pair had no more than zero downvotes. The first $2,500$ pairs were filtered in and the rest were removed.
- **RCHD Filtering**: This dataset doesn't include the total number of upvotes. Given this, $2,500$ pairs were randomly chosen.

It's possible that lower ranked pairs (less upvotes) from the SRCD dataset may contain subcultural votes that are only understood to a smaller subset group of people. While this can be true, there are several reasons why they were omitted for now:

- **Unreliable humour**: Lower-ranked responses may not consistently reflect humourous intent, making them less suitable for humour-focused fine-tuning.
- **Less engagement**: Lower ranked responses were interacted with a limited audience, suggesting the content could be off-topic or unfamiliar to the readers.
- **Annotation noise**: These responses are more likely to introduce noise to the model, causing it to produce "less funny" responses to a wider audience setting.
- **Computational limitations**: Given the limited computational resources available, prioritisation was given to pairs with a wider audience appeal.

The pairs with the highest number of votes, often were shown to contain elements that were understood by a wider audience.

- **Prompt**: "I have to be the last one to come to this realization"
- **Comment**: "Well, GTA Online definitely has plenty of children."

The comment gained a total of 3644 upvotes and 0 downvotes. While the comment may not align properly with the parent comment, the main element "GTA" is widely understood.

An important observation is that comments with a lower count of total upvotes may not necessarily be less humourous [96]. An example of a comment being upvoted and downvoted 8 and 0 times is shown below:

- **Prompt**: "Have an upvote for your honesty"
- **Comment**: "Um, I think you should take some time away from reddit OP."

Can be categorised as aggressive which is the most common form style of humour that appears in social media comments, as shown by a study conducted with young adults [43].

While this is a simple filtering method, it's a simplistic way of maximising the chance of eliminating comments that are less funny to a wider audience.

3.2.3 Dataset Cleansing

In social media platforms and in online environments in general, users are more likely to express themselves freely, even if their expression may be seen as offensive by the majority of the audience [87]. this can be seen on Reddit where most users are anonymous, with an increased likelihood of offensive comments being posted [65].

Because of this, it's important to remove content containing offensive and bias words including but not limited to politics, body shaming, religion, age, racism, sexual orientation, accessibility, swear words, etc. The simplest way of filtering content is to check for specific offensive keywords, such as "fat", "idiot". Phrases such as "He gained so much weight that he looks like an elephant" present challenges for keywords based methods, as there isn't a specific keyword that may be labeled as "offensive". Additionally, these phrases containing indirect or implicit bias, are more likely to escape detection [80]. More abstractly, the context-dependency nature of language makes it particularly challenging to detect offensive content [21]:

- **Word ambiguity**: Words can be offensive in one context but not in another. E.g. the word "crazy" on it's own may be seen as offensive. "crazy good" most likely won't appear as offensive to humans, but LLMs may label it as offensive.
- **Subtle**: Toxic comments can be direct or indirect, where indirect toxicity is harder to detect as it's often not subjective [104]

Latest research shows that some pre-trained models can be suitable for reducing toxic humour in the dataset, although more research is needed in this area to detect offensive humour. Using pre-trained models such as BERT and ERNIE have shown significant improvements in term of accuracy in removing offensive content in data [104].

Out of four total models considered, Detoxify was chosen for this task. Detoxify is the most suitable choice for this case, given that the model is efficient, even with limited computational resources. Additionally, Detoxify provides with a multilabel classification, given that a statement may be labeled as "toxic" but not "offensive". Statement labeled as "normal", are kept in the dataset, indicating that there haven't been detected any toxic statements.

Other models were considered, including BERT and fine-tuned models, such as HateXplain were considered. A detailed analysis of each model considered for this task is analysed on the appendix A.4.

Dataset Toxicity Removal

Based on all research already done in this area, the following steps will be used to detoxify the whole dataset:

1. **Keyword-based removal**: Content removal based on offensive terms/tokens. This mainly includes racist slurs and offensive words.
2. **Bias-based removal**: Content removal based on bias phrases including context in politics, gender orientation, sexual orientation. For example "As a woman", "As a transgender" [60].
3. **Pretrained model**: Lastly, a pre-trained model was used to capture context-dependencies to filter out direct and indirect aggression.

Figure 3.1: Steps taken to detoxify the dataset.

While these methods may not fully eliminate offensive data, but it illustrates a state-of-the-art methodology which can be expanded in the future.

Some content with deeper meanings may not be captured completely, including:

- **Prompt:** That can be a full-time job for some people.
- **Response:** What is she famous for? All I found was her being hot on Instagram.

and

- **Prompt:** want to go bowling?
- **Response:** hey its me ur hooker

The first prompt-response may be seen as humiliating diminishing by a certain group of people and the second example, is an example of an ambiguous response, where the term of "hooker" may have different interpretations, depending on the context. This shows that at the time of writing, there isn't a model which can completely eliminate toxic prompt-responses.

The dataset cleansing process is performed before the fine-tuning process and is stored to be cross-used for all chosen LLMs for fine-tuning. Using the same dataset for all LLMs, allows a fairer comparison between all models.

3.3 Fine-tuning LLMs

Choosing the most suitable LLMs for fine-tuning is a very crucial step. The non-deterministic nature of LLMs further amplifies the complexity of choosing the right LLMs. Despite this, there are several criteria that influenced the choice of the LLMs.

- **Potential improvement**: Predicting the amount of improvement per fine-tuned LLM on humour generation compared to its zero-shot equivalent.
- **Data overlap validation**: Validation of whether several examples from the dataset was already included in the pre-trained data to avoid potential overfitting.
- **Resources Requirements**: LLMs models often require enough VRAM to be loaded. For example, usually LLMs with about 7 billion parameters require 14GB of VRAM if loaded under half precision (FP16) [4].

Studies have shown that performance improvements are not guaranteed with LLMs [24]. LLMs lose some of their generalisation capabilities after the fine-tuning process, as the fine-tuning is updating their parameters on a specific pattern scope. Additionally, overfitting is a common problem with fine-tuning, especially if the dataset is too small compared to the total parameters possessed by the model. One study based on Llama 3.0 8B (8 billion parameters) identifies a threshold in dataset size, showing that fine-tuning an LLM with a dataset of as few as $1,000$ - $2,000$ segments (entries) will lead the LLM to overfit [94]. A Datasets consisting of around $5,000$ segments is the minimum required to improve the chosen LLM beyond the zero-shot performance. Datasets with segments $100,000$+ will yield a significant improvement in terms of performance.

There isn't a definitive answer on how many data entries are needed to prevent overfitting on fine-tuning. Given that this study was applied for Llama with 8B parameters, other models with similar sizes of parameters may have similar thresholds.

While a relatively large dataset is preferred for better fine-tuning results, it's important to consider the kind of data that's included in the selected dataset. It's generally known that the fine-tuning LLMs with datasets outside their original pre-training distribution yield to higher performance outputs [53].

3.3.1 Dataset overlap detection

To ensure the fine-tuning process will improve each chosen LLM, an overlapping testing was performed by doing a pairwise comparison of 50 prompts by comparing each human response with the equivalent zero-shot LLM response. There are various techniques that can be used for this purpose. a notable one is using the Cross-Encoder architecture.

Cross-Encoder architecture

The Cross-Encoder architecture utilities pre-trained models including BERT or RoBERTa are the most advanced pre-trained models used to capture contextual similarity between two sentences.

Given two input sentences A and B, a Cross-Encoder jointly encodes them as a single sequence, with a separator SEP added in between. This concatenated input is passed through a transformer T to produce contextual embeddings:

$$H = T([CLS, A, SEP, B, SEP])$$

The similarity score is derived from the [CLS] token's hidden state h_{CLS} using a linear layer with sigmoid activation:

$$s = \sigma(w^\top h_{CLS} + b)$$

where w and b are learnable parameters, and σ is the sigmoid function. This allows token-level interaction between A and B. This improves upon the previous method by enabling

better detection of nuanced humour features such as irony and figurative language compared to independent encoders [76].

Alternative methods were considered such as the Word Sense Disambiguation with Cosine Similarity [35, 49]. However, this method would present challenges with figurative language including metaphors, personification and hyperboles [9], due to figurative expressions often mismatch between the literal and intended meanings. For more information refer to A.0.1.4.

3.3.2 LLMs selection

Given that multiple LLMs were pre-trained with different datasets and architectures, the following selected LLMs are expected to produce different responses before and after fine-tuning.

- **LLama3.2 3B**: Llama has shown improvements with humour classification, including sarcasm and irony, scoring an accuracy of 89% [99]. The model which was introduced in this study, was used for the fine-tuning experiments. It has 3 billion parameters.
- **R1 (DeepSeek) 8B**: DeepSeek R1 is a relatively new LLM which has shown to match the GPT-4o capabilities in terms of performance. While there isn't any indication of how good it is for humour generation, this was the first time that this model was used for this purpose. It has 8 billion parameters.
- **Gemma 2 9B**: Gemma 2 is also a good choice given that it's trained with human prompt-response pairs [90]. It has 9 billion parameters.

Given the high demand of resources required for fine-tuning these LLMs, the list above lists the variations of specified models (DeepSeek, Llama and Gemma) containing a relatively small number of parameters.

LLMs considered but not used

Several LLMs were considered but not used, including GPT-4o, Mistral-7B and Qwen 2.5 7B. GPT-4o was not included due to limitations offered by OpenAI for fine-tuning. Mistral and Qwen posed limitations with underperformance compared to the previous listed LLMs (see A.5).

LLMs by default can be vague and ambiguous [100] as they are trained with vast amounts of data to be used for generic purposes. This can lead to several limitations including outputting long sentences and thought-chain process [50].

3.3.3 Prompting Engineering

Given that humour generated responses from all LLMs must relate to the prompt, prompts have to be carefully crafted and optimised to give more specific and targeted responses. There are several techniques that were applied to maximise the potential of generating more humourous responses:

- **Temperature**: According to a study, temperature value less than 0.5 has shown that 73% of the models used showed a peak performance with humour generation [29].
- **Limit generalisation**: LLMs were instructed to specifically return a response to the given prompt, and hence, minimise the chance of outputting chain-of-thought [15]. Each LLM

is instructed using the same prompt (see A.0.2). More specifically, the prompt follow the steps below to generate the response:

  - **Instruction**: Generate a funny response
  - **Input**: Why do the chickens cross the road?
  - **Output**: To cross to the other side.

- **End of text**: Each prompt had a token which acted as the end of the output response of the LLM.

The whole process of fine-tuning varied across different LLMs. While the dataset used, prompting and structure remained the same, fine-tuning largely depends on each LLM's architecture. Some LLMs took longer for the fiine-tuning process to complete than others. For example, DeepSeek took the longest.

3.4 LLM judges for humour analysis

LLMs have been used by researchers recently to evaluate humour sentences, including GPT-4 [31]. Specifically, GPT-4 has shown good performance on categorising humorous sentences vs non-humorous ones [93]. The evaluation using LLM models have shown mixed results.

LLM judges have scored highly on humour that humans have also rated highly. This is limited to aggressive and self-defeating jokes. LLMs have shown to struggle to evaluate humour that are mostly of outside their trained preferences. Additionally, LLMs have shown to misinterpret humour requiring contextual understanding, such as sarcasm and subcultural settings.

To mitigate this, I proposed a combination of human and LLM judges as part of the evaluation process. While having more than one participants increase the internal validity and consistency of the humour evaluation experiment [84]. Given this, having one human will reduce the bias on humour involving, but not limited to, cultural settings, age and background.

Given this, the whole evaluation will be split into three parts:

1. **LLM judges selection**: Evaluation of pre-existed annotated dataset and choosing the LLMs that agree the most with the already-rated dataset.
2. **LLMs humour generation evaluation**: In-depth comparison of state-of-the-art models with their fine-tuned equivalents based on the Reddit dataset.
3. **Analysis of the results**: Analysing the results and choosing the best performing fine-tuned LLM and the best state-of-the-art model to answer the research questions.

3.4.1 LLM judges selection

Three LLMs were chosen to conduct the comparison of humour generation across all LLMs. These LLMs referred to as "LLM judges".

- **R1 (DeepSeek) 8B**: The same model variation as used for humour generation was chosen as one of the judges. While there isn't any indication of how good it is for humour generation, this was the first time that this model was used for this purpose. It has 8 billion parameters.

- **Mistral v3.0 7B**: Mistral was chosen due to its promising capabilities of understanding human content. There's limited research on its humour understanding capabilities.
- **Qwen 2.5 7B**: Qwen has shown strong state-of-the-art performance in multiple humour evaluation tasks including SemVal and Oogiri-GO [40]. Qwen was able to outperform other LLMs, including GPT-4o if trained with the Creative Leap of Structured Thought (CLoST) method.
- **GPT-4o**: GPT-4o is widely considered among the most advanced LLMs at the time of writing, often outperforming human, according to a study [13].
- **GPT-4o-mini** GPT-4o-mini is a cost-efficient variant in the GPT-4o series designed to deliver strong performance with reduced computational demands. According to OpenAI, outperforms GPT-3.5, which has been shown to perform well in humour analysis [68]. While there's limited research on its humour capabilities, it has been tested in scenarios involving meme (setup and punchline) interpretation, showing a narrative progression within the punchline [63]. This indicates that it can show promising results when evaluating prompt-responses.

Additional LLMs such as Mistral were considered. Mistral was examined in the LLM judges agreement section.

The LLM judges were not fine-tuned as part of this project. This would be an interesting area of exploration in as a future work, given that existing studies have shown that LLMs, including Llama 3 have shown significant improvements with humour recognition [99].

LLM judges had to be properly instructed to do the following tasks by using prompt engineering:

LLama 3.1 8B Fine-tuned Llama has shown improvements with humour classification, [99]. The model which was introduced in this study, was used for the fine-tuning experiments. It has 8 billion parameters. This LLM was removed form the judges selection, due to its much lower than expected performance during the evaluation (see more in 5.2).

- **Humour personalities**: Each LLM judge was instructed to prefer a specific humour style, as seen from 2.1.1. This tactic was used to diversify the judges to measure generated humour from a different humour style preference [31] (more on 5.3.1.
- **Measure humour**: The humourous responses generated by both zero-shot and fine-tuned LLMs was independently assessed by instructing each judge to classify each response, specific to each humour style, according to the following binary categories:
  - Funny or Dull
  - Funny or Boring
  - Funny or Serious
- **Compare humour**: Each LLM judge was instructed to rank four different responses (three generated and one original human response from the dataset) and answer a few questions to justify their ranking (more on 5.3.1.

By providing specific prompts requiring the LLM to respond with succinct answers, it prevented it from generating thought processes and made it easier for collecting the responses from each LLM and use it as part of the analysis.

3.4.2 LLM judges-human agreement analysis

Initially, I conducted a brief evaluation to determine which LLMs match the closest to human perception of how funny a statement can be. One way of conducting an evaluation is to use an already-existing dataset, where human annotators have already evaluated how "funny" a given statement is. The following datasets were considered as part of this evaluation:

1. **Hahackathon dataset**: This dataset contains up to $9,000$ of sarcastic phrases and where annotated by humans from a variety backgrounds, including gender, political stances and income levels [27].
2. **Combined dataset**: 50 segments were selected randomly to be annotated (see 3.2.2).

The Hahackathon dataset phrases are similar to the Reddit dataset used to fine-tune the LLMs. Both datasets were used to analyse the agreement scores between the author of this paper and the LLMs by using the combined dataset, and annalysed the agreement score between the Hahackathon dataset and LLMs. More specifically:

1. **Hahackathon dataset agreement**: 50 segments, indicated as "funny" were randomly selected to be annotated by potential LLM judges.
2. **Combined dataset agreement**: 50 segments were randomly selected and used to be annotated by the author of this paper and the chosen LLM judges.

Given this, 10 experiments were performed to measure the agreement scores and to finalise the choice of LLM judges, with 5 experiments on each dataset by comparing the human annotators with 5 LLMs (DeepSeek R1, Llama 3.1, Qwen 2.5, Mistral v0.3).

3.4.2.1 Agreement metrics

To assess how well each potential LLM judge align with real human judgements on humour, the Krippendorff's Alpha (α) agreement metric was used. This metric is particularly helpful, given that the rating values are continuous and accounts for the degree of disagreement among raters and is able to adjust in situations where the rating occured by random chance.

Other agreement ratings, such as the Cohen's Kappa and Weighted Cohen's Kappa where considered. The problem with these metrics are that the standard Cohen's Kappa doesn't work with continuous values and the Weighted variant does not distinguish between minor and major disagreements.

Krippendorff's Alpha (α)

Krippendorff's Alpha is a statistical measure to assess how "well" two dataset labels "agree" with each other. [36]. This metric works well for continuous rating values, making it suitable for measuring the agreement scores between the LLMs and human annotators.

The general formula for Krippendorff's Alpha is defined as:

$$\alpha = 1 - \frac{D_o}{D_e}$$

Where:

- D_o: The observed disagreement.
- D_e: The expected disagreement by chance (no meaningful agreement).

The observed disagreement (D_o) is defined as:

$$D_o = \frac{1}{N}\sum_{i=1}^{N}(r_i^{LLM} - r_i^{Human})^2$$

Where:

- D_e: The expected disagreement due to chance.
- K: The number of individual ratings (from both LLM and human) across all punchlines.
- r_i, r_j: Observed rating values from the set of all K ratings.

The expected disagreement (D_e) for continuous ratings is defined as:

$$D_e = \frac{1}{K(K-1)}\sum_{i=1}^{K}\sum_{j=1 j\neq i}^{K}(r_i - r_j)^2$$

Where:

- p_m and p_n are the proportions of ratings at levels m and n respectively,
- $(m-n)^2$ is the squared distance between score levels.

These two equations are substituted into α for each LLM-human pair-wise comparison to calculate the agreement score for each. These equations were used in both phases.

An α value of 1 indicates perfect agreement, while 0 indicates agreement equal to chance. A value of 0.8 and above is typically indicating an acceptable agreement value, but any value lower than 0.6 shows a poor agreement.

Based on this information, the Krippendorff's Alpha is a very suitable type of statistical measure to indicate how much each LLM agrees with the human annotators. I chose 3 LLM judges based on the highest Krippendorff's Alpha score.

3.4.2.2 Agreement Evaluation Process

The agreement scores were computed in two phrases using Krippendorff's Alpha to compare each LLM ratings with the human annotators.

Phase 1: Hahachathon dataset agreement

The Hahackathon dataset includes at least $9,000$ punchlines annotated by a diverse set of humans giving an average rating from 0 to 5 per punchline. For this agreement analysis, 25 segments were randomly selected. Out of the 25 segments, the following constraints were applied:

- **Humour diversity**: 25 segments were chosen with five segments chosen from each humour rating interval: 0-1, 1-2, 2-3, 3-4, 4-5.
- **Humourous statements**: the 25 segments were sampled from all segments where each statement was labeled as humourous.

The seven potential LLM judges were prompted to generate their own humour ratings for the same 25 items. These model predictions were then compared with the human ratings using the Krippendorff's Alpha (interval level) in a pairwise fashion; each LLM was compared with the human ratings.

The choice of the Hahachathon dataset was vital for the agreement analysis, given that it was amount the very few datasets available containing humour ratings from a diverse group of humans. One limitation from this agreement evaluation, is that each data segment was a one-liner punchline, without a prompt, which doesn't give the full picture to select the LLM judges.

Phase 2: Real-world prompt-response agreement

The second agreement analysis is performed in the same way as the phase one, but with using 25 prompt-responses from the combined dataset used to fine-tune the LLMs. For this process, each LLM is instructed to annotate each prompt-response by ranking all four responses from each selected segment. The segments were chosen randomly and at the same time, the author of this paper annotated each prompt-responses by ranking each responses from each prompt. For the ranking process, the following criteria took place:

- **Transparency**: Both LLMs and author were not aware of which response came from the fine-tuned LLMs and which from the original human commenter.
- **Shuffling**: Each response was shuffled for each annotation round for both LLMs and author.

Finally, the Krippendorff's Alpha equation is applied by substituting the corresponding values to calculate the pair-wise scores for each LLM on each phase. At the end, a total of **12** agreement scores were calculated.

Chapter 4

Implementation & Fine Tuning

This chapter focuses mainly on the fine-tuning workflow, including pre-processing, fine-tuning and the overall infrastructure.

4.1 Development environment

All experiments and fine-tuning procedures were conducted using Google Colab [77], given the ease of access to high-performance GPUs, including NVIDIA T4 GPUs with 15 GB of VRAM. Google Colab also provides a user-friendly environment, with pre-configured support for Python and a range of machine learning frameworks, including PyTorch, Hugging Face Transformers, and TensorFlow. Thanks to the pre-configuration, several libraries were downloaded without causing version conflicts and the end user doesn't have to worry about creating customised Python environments.

Other platforms were considered, such as Kaggle Notebooks [18], and RunPod.io [56]. For this project, Google Colab proved to be the most suitable choice for the following reasons:

- **Setup**: The setup process on Google Colab is simpler than most other cloud-based computing engines. For example, RunPod often requires manual configuration of Docker constrains and custom environments which can produce several challenges and conflicting library versions, if not installed properly. On top of that, Google Colab provides a streamline integration with Google Drive and GitHub services, allowing an easy process to store the results of every experiment run.
- **Hardware**: While all cloud-based computation engines provide the GPU resources at a cost to run most experiments, Kaggle Notebooks is known to be inconsistent with the GPU sessions. Additionally, RunPod provides a more complex setup process, and the required GPU needed to run all the models were more expensive.

In summary, Google Colab was the most suitable choice the development environment, providing a reliable setup process and eliminates unnecessary extra steps with installing the libraries and packages needed to run the experiments.

All hardware and software equipment used for each experiment are shown below:

Component	Specification
GPU	NVIDIA Tesla T4 (15 GB VRAM)
RAM	12.7 GB
Disk Space	112.6 GB
OS/Platform	Ubuntu 20.04 (Colab VM)
Programming Language	Python 3.10
IDE	Jupyter Notebook (Colab UI)
Core Libraries	transformers, unsloth, datasets, torch, triton, scikit-learn, detoxify

Table 4.1: Hardware and software setup used in the project

Python [30] is the mainstream programming language used for most machine learning tasks, including experimentation with LLMs. All tools, libraries and packages were installed using the pip package manager [5].

Unsloth [91] was the main library used to develop the experiments with all the LLMs.

4.2 Pre-training data overlap detection

To minimise the risk of overfitting due to some, if not, all the data entries already existing in the pre-trained data of the LLM used (pre-training overlap), a Cross-Encoder based on BERT model was implemented to measure the semantic similarity between the responses of each dataset pair to the zero-shot LLM responses. The threshold of 0.9 was set, meaning any scores above 0.9, were classified as “seen”.

Prompt	Human	deepseek	llama	gemma
Local Property Tax?...	LPT: Don’t use abbreviations	0.2338	0.0179	0.0810
When someone shows me ...	”No, thanks. I’m a veget ...	0.1813	0.0512	0.0459
That was your first mistake...	So, I have a Racing Snail...	0.3823	0.2477	0.2667
This is /r/oneliners...	Wife said she wanted a ring...	0.1188	0.4117	0.0088
Clap clap!...	I got my best friend a fridge...	0.4185	0.1364	0.0202
It’s next to the...	how do I unsubscribe to...	0.2003	0.1075	0.3101
Came upon	I came across your wife...	0.6539	0.3354	0.0432
Read this in Conan’s voice...	FedEx said that it shipped...	0.4187	0.6022	0.4019
r/threeliners	My doctor told me I had...	0.1608	0.0551	0.1673
They also would have ...	My friend Ty came first...	0.3158	0.3305	0.0296

Based on the results, all scores are well below 0.9, indicating minimum to no similarity across all entries (see all entries in appendix A.0.1.2). The three columns on the right indicate the similarity score of the corresponding LLM compared to the original human response.

Given that the scores are well below the threshold of 0.9, all LLMs were safe to be fine-tuned and the chances of parts of the dataset being included in the pre-training data are minimal.

4.3 Dataset pre-processing

The dataset was preprocessed based on the following steps:

- **Format matching**: Given that two datasets were used for the fine-tuning process, both datasets had to be reformatted to be combined under a common format.
- **Dataset cleansing preparation** given that this process would take a substantial amount of time to ”cleanse” more than $1.3M$ segments, a smaller subset was extracted of. The subset was $15,000$ segments.
- **Keyword detection**: Remove all data segments including some of the common keywords that can be seen as toxic or offensive. These include swear words, cursing, racial slurs, etc. Each of those words were defined in the form of a list.
- **Bias-based phrases**: Phrases or elements that can cause bias. These include phrases such as ”as a transgender”, ”as a woman”, etc. For these, a string detection from each prompt and response was done and the pair was removed if any bias-based phrases were detected.
- **Removal of deeper toxic content**: Removal of content that deeper nuances of bias or toxicity that cannot be detected by keyword search. A pre-trained model was used to detect deeper findings of tonicity, called Detoxify.

The data cleansing process removed several hundreds of data segments, implying that more advanced methods could be used in this area.

Chapter 5

Evaluation

5.1 Evaluation introduction

During the evaluation period, I conducted the following steps to answer the initial research questions (R1 and R2). The responses were judged by 5 LLMs (LLM judges) and one human judge. The human judge was the author of this dissertation. I broke the evaluation plan into four different steps.

1. **Step 1** Use the LLM judges and human annotation to rate responses produced by zero shot LLMs and compare them on each prompt.
2. **Step 2** Use the LLM and human judges to rate responses produced by fine-tuned LLMs and compare them on each prompt.
3. **Step 3** Comparison of each fine-tuned LLM generated responses and the original human response on each prompt using each LLM and author judge.
4. **Step 4** Results analysis and answering the original research questions.

While a single human judge was used during the evaluation period, this can bring several disadvantages, including:

1. **Ranking bias**: Each humans have different taste of humour and using one human for sole evaluation may mark certain jokes as "funnier" than they are supposed to be or "not funny" which can be perceived as "funny" to people from different cultures.
2. **Humour diminishment**: During the implementation method and the preparation of the evaluation section, several jokes would be seen by the single human judge. During the evaluation period, the human judge is likely to come across to the same jokes which may be perceived as less funny and produce inaccuracies with the rating.
3. **Lack of diversity**: Certain regional or cultural jokes may not be understood by the single human judge.
4. **Smaller evaluation rating size**: Having one human annotator, only one opinion will be given per statement.

Given these disadvantages, a few things were done to ensure a higher validity:

1. **Data shuffle**: Prompt-responses were picked randomly from a subset unseen by the human annotator before.
2. **Cultural immersion**: Culturally specific jokes were assessed based on humour structure (e.g., irony, absurdity) rather than presumed cultural context, to reduce misinterpretation.
3. **Larger evaluation size**: More different prompts can be evaluated per human.

5.2 LLM human agreement

The human agreement analysis was performed to better understand how each LLM agrees with the human annotators. The agreement analysis planning is explained in more depth here 5.2. For the phase one, the following scores were calculated:

Table 5.1: Model Agreement with Human Ratings (Ranked by Pearson Correlation)

Model	Krippendorff's α	Pearson Corr.	Spearman Corr.
GPT-4o	**0.9880**	**0.4492**	0.4141
Quen	0.9863	0.4089	**0.4220**
Mistral	0.9850	0.3961	0.3859
DeepSeek	0.9864	0.2463	0.2507
LLaMA	0.9839	0.1281	0.0750
GPT-4o-mini	0.9866	0.0547	0.1294

While the Kirippendorff's alpha values suggest high pair-wise agreement between the LLMs and human annotators, the Pearson and Spearman correlations gives a more complete picture in terms of agreement. Based on the findings, GPT-4o and Quen have the highest agreement with the humans, while Llama and GPT-4o-mini the lowest. This indicates that the Kirippendorff's alpha values of Llama and GPT-4o-mini are well overestimated than the actual agreement score.

While GPT-4o and Qwen seem to have the highest agreement scores with the humans, the second phase shows the pair-wise agreement between the LLMs and the author of the paper.

Table 5.2: Agreement Between Human and Model Rankings

Model	Krippendorff's α	Pearson Corr.	Spearman Corr.
GPT-4o	**0.7964**	**0.3542**	**0.3431**
GPT-4o-mini	0.7125	0.2367	0.2259
LLaMA	0.6821	0.2287	0.2149
Qwen	0.6733	0.2124	0.1985
DeepSeek	0.6125	0.1783	0.1747
Mistral	0.5902	0.1496	0.1428

Based on the findings, the Krippendorff's α values vary more than the ratings compared on table 5.1. In this case, GPT-4o and GPPT-4o-mini show relatively high agreements with the paper author, surpassing LLaMA and Qwen in correlation scores, suggesting that the judgments may align closer to the ranking of the responses, rather than rating the "funniness" of a statement. In this phase, Mistral and DeepSeek showed the weakest results showing a more limited ability with evaluating humour than the other LLMs.

Overall, these findings show interesting results with different agreements between ranking responses (phase 2) and measuring humour (phase 1). Based on the results, the following LLMs as judges were chosen:

- **GPT-4o**: GPT-4o showed the best performance and the highest agreement scores in both phases. GPT-4o is shown to be the most reliable humour evaluator LLM.
- **Quen**: Quen has performed second only to GPT-4o during the first phase, but slipped to fourth place during the second phase. Quen was still chosen given that it maintains a close to 0.7 agreement score.
- **GPT-4o-mini**: While GPT-4o-mini performed very well on the second phase, it still lagged to the bottom on the first phase. Given that humour is subjective, it's important to maintain a diverse group of judges. In real world evaluation scenarios, it's expected that judges may differ in their ratings, as a complete agreement across all judges is uncomon due to individual differences.
- **DeepSeek**: DeepSeek is an interesting example, given that it was ranked at similar places (fourth and fifth during first and second phase respectively). DeepSeek was chosen to maintain a diverse set of judges as part of the evaluation.
- **Mistral**: For similar reasons, Mistral was chosen.

Llama was the only LLM that despite being selected, it was left out because during the crowd score evaluation phase, it gave a score of 0 for 97 out of 100 statements. While it's possible that Llama requires different prompting to perform better, further analysis is required to confirm this case.

5.3 Humour analysis

The evaluation is split into two parts.

- **Humour improvement**: Zero shot and fine-tuned humour generation were analysed and compared individually. This part of the analysis answers research question R1 (see 1), on how much and whether fine-tuned LLMs outperform their zero-shot counterparts.
- **Comparison to human humour**: Fine-tuned humour was compared with the original human response per prompt and analysed whether the fine-tuned LLMs can generate humour that can be perceived as funnier than the human counterparts. This answers the second research question R2 (see 2).

Given that shorter statements contribute to more humourous statements, each fine-tuned LLM managed to match the typical original human sentences lengths.

Model	Fine-tuned Avg	Zero-shot Avg
DeepSeek	11.83	47.08
LLaMA	18.24	23.43
Gemma	10.84	19.06
Human	15.06	

Table 5.3: Average token length per model and response type

Notably, Gemma was able to produce shorter sentences than the rest of the LLMs where DeepSeek saw a significant improvement from around 47 to nearly 12 words per sentence on average.

5.3.1 Zero-shot vs fine-tuned generated responses

This comparison was evaluated by providing four different prompts supplied to each judge LLM. The full prompts for each LLM can be seen on the Appendix. The prompts instructed each LLM judge to simulate a different humour taste and personality [58].

- **Affiliative**: You are a humor evaluator who prefers light-hearted, friendly, and socially inclusive humor. You enjoy witty banter, storytelling, and jokes that build connection without mocking anyone.
- **Self-enhancing**: You are a humor evaluator who appreciates positive, resilient humor that helps people cope. You enjoy jokes that reframe tough situations with optimism or highlight life's absurdities.
- **Aggressive**: You are a humor evaluator who enjoys bold, sarcastic, or teasing humor, even when it targets others. You value sharp wit, irony, and put-downs that might be offensive but pack punch.
- **Self-defeating**: You are a humor evaluator who finds self-deprecating and vulnerable humor the most relatable. You enjoy when someone makes fun of their own flaws or failures in a funny, endearing way.

Additionally, for each humour style, three different templates were employed to capture the comedic intent to compare the zero-shot with fine-tuned responses in a more diverse setting:

1. **Funny or dull**: Assessing whether the response is perceived as humorous or lacking in engagement.
2. **Funny or boring**: Assessing whether the response is perceived as humorous or uninteresting.
3. **Funny or serious**: Assessing whether the response is perceived as humorous or serious in tone.

In total, there were 12 different prompt evaluations per response, and 48 in total per prompt, given that each prompt contained four different responses. In total each prompt was evaluated by five different judges. In total 100 prompts were evaluated. This gives us the final number of $24,000$ across both zero-shot and fine-tuned settings.

For each response, a total score out of 12 is calculated based on how many times the LLM judge was labeling the responses as "funny". The higher the score, the funnier the response is on a diverse setting.

5.3.1.1 Combined LLM judge results

This part of the evaluation phase presents a comparative evaluation of the three fine-tuned LLMs: Gemma, Llama and DeepSeek, their zero-shot counterparts. All statistics were evaluated based on all LLM jduges including the author of this paper.

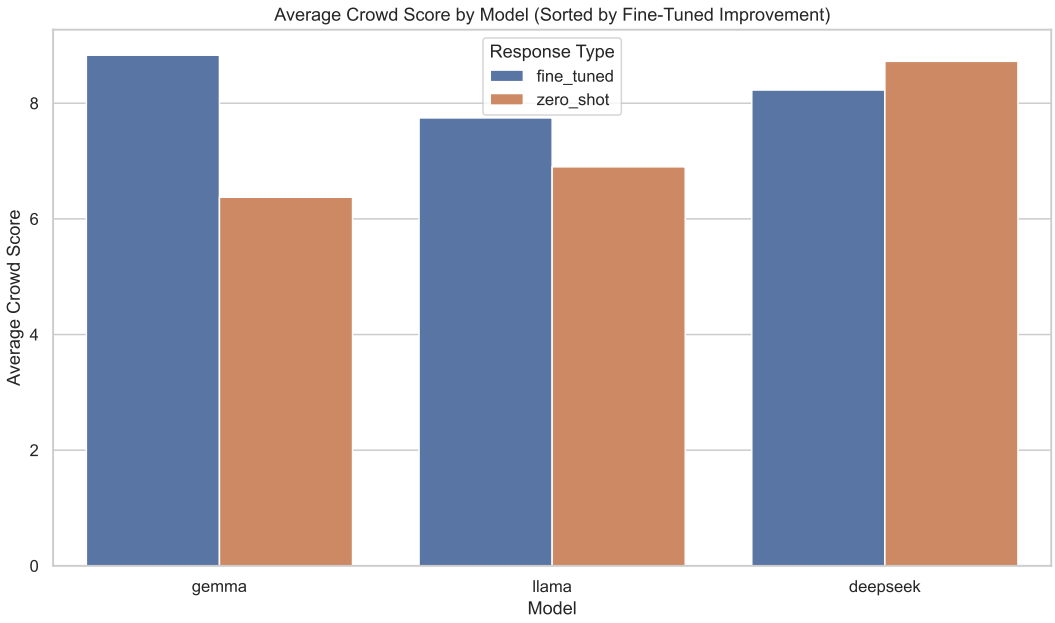

Figure 5.1: Average Crowd Score by Model (Sorted by Fine-Tuned Improvement)

Based on the scores, there's a significant improvement on responses produced by the two fine-tuned models of Gemma and Llama. Gemma noticed more than one point average increase, while Llama less than half a point. DeepSeek recorded a dip in performance, indicating that, on average, the LLM judges perceived DeepSeek's responses as less humorous compared to the other models.

5.3.1.2 DeepSeek LLM judge

DeepSeek have shown some interesting and unexpected results.

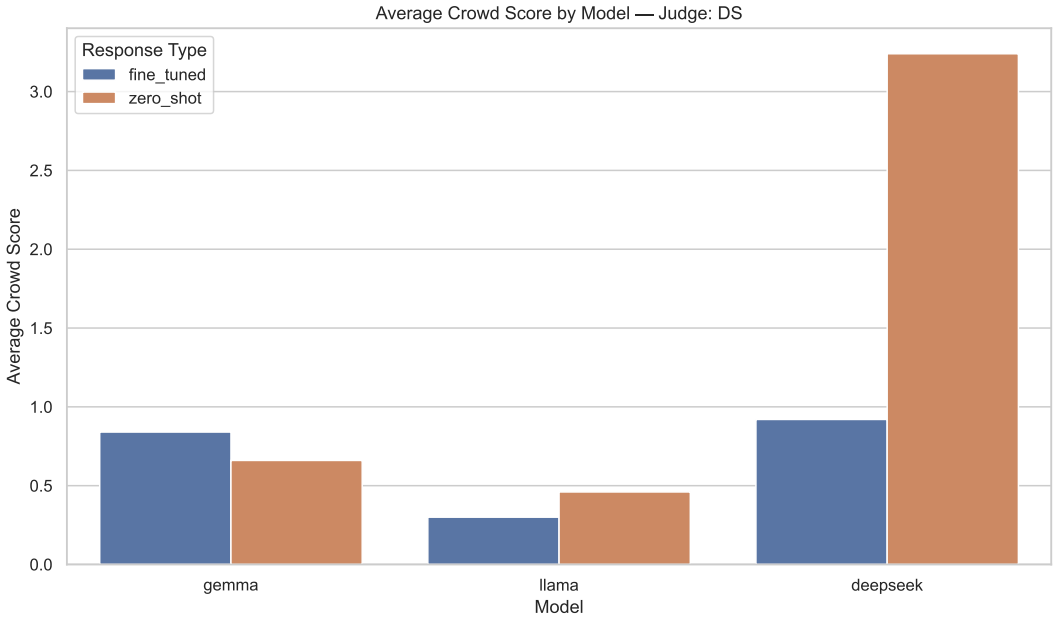

Figure 5.2: Average Crowd Scores — Judge: DS

DeepSeek gave much lower scores than any other jduge on average, indicating a complete different way of evaluating humour. Given that DeepSeek was also fine-tuned for humour generation, the results show that the zero-shot counterpart produces three times more humourous responses than the fine-tuned version.

While given that this LLM is relatively new, there's limited information of what might be the case, giving it an interesting area to conduct further experiments in the future. Notably, DeepSeek is more biased to its own humour generation and the low scores on its fine-tuned version can reflect to the potential absence of the new humour nuances it produced.

Given that only DeepSeek has shown positive bias towards its own humour generation, further study is needed to conclude the case that LLMs can have a positive biased on their own humour generation, which could expand into different areas of research.

5.3.1.3 Quen LLM judge

A notable statistic from Qwen, indicates that all LLMs have shown an improvement after being fine-tuned.

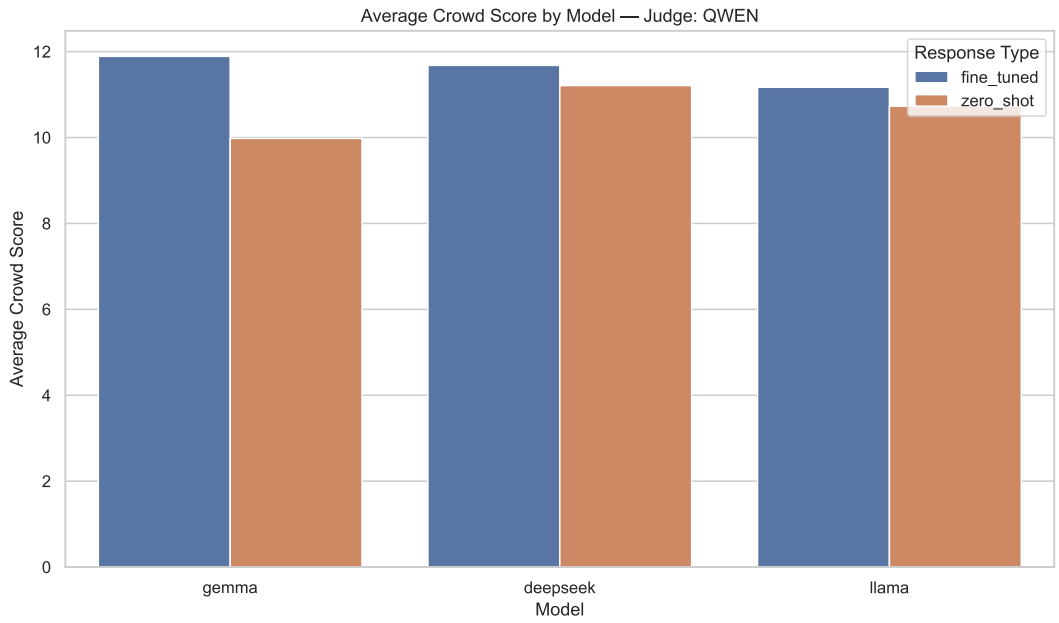

Figure 5.3: Average Crowd Scores — Judge: Qwen

This further proves that different

Given that each LLM evaluated the responses differently, it was important to analyse the results based on each LLM judge. See Appendix A.0.1.1. Based on these results, we observe the average crowd score of gemma to double the the zero shot crowd score, based on the GPT-4o LLM judge. GPT-4o-Mini DeepSeek and Qwen also agree with the improvement of humour generation using gemma.

Furthermore, each response was evaluated based on asking the judge LLM of whether each response is funny or dull, boring or serious. The findings show that responses originally labeled as "boring" in the zero-shot versions, showed the highest improvement with more than 9% overall, with serious being the least.

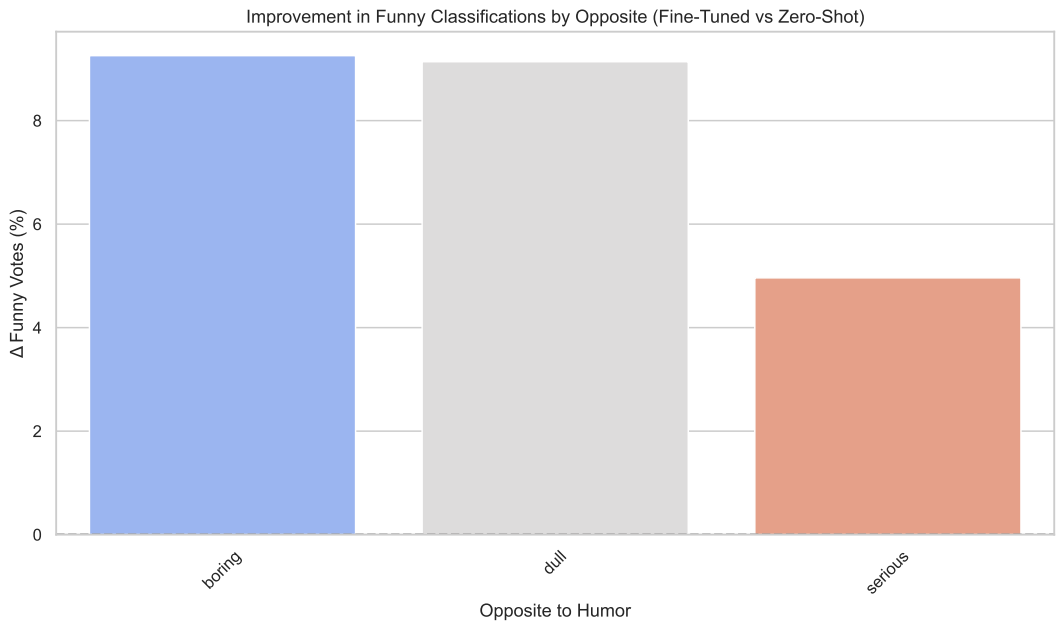

Figure 5.4: Difference in funniness classification rates across opposite labels (boring, dull, serious) when comparing fine-tuned vs zero-shot responses. Positive values indicate that fine-tuned responses were more often labeled as “funny”.

More abstractly, the GPT-4o-Mini labeled more than 25% of originaly labeled as “boring” labels to “funny” in the fine-tuned equivalent models. GPT-4o indicates that the label of “dull” showed the highest conversions to “funny” showing a roughly 17%. The “serious” label was widely shown the lowest conversion rates.

5.3.1.4 Crowd score analysis ranings

The crowd score was used to indicate how “funny” a response in relation to its prompt is. Each croud score was evaluated per repsonse response independently to maximise its effectiveness.

Prompt Index	Judge	Model	Zero-Shot Score	Fine-Tuned Score	Delta
99	gpt-4o-mini	gemma	0	12	12
37	gpt-4o	gemma	0	12	12
66	gpt-4o-mini	llama	0	12	12
1	gpt-4o	gemma	0	12	12
49	gpt-4o-mini	llama	0	12	12
38	gpt-4o-mini	llama	0	12	12
36	gpt-4o-mini	llama	0	12	12
12	gpt-4o	gemma	0	12	12
14	gpt-4o	gemma	0	12	12
16	gpt-4o	gemma	0	12	12
60	qwen	gemma	0	12	12
20	gpt-4o	gemma	0	12	12
29	gpt-4o-mini	llama	0	12	12
35	gpt-4o	gemma	0	12	12
40	gpt-4o	gemma	0	12	12
6	gpt-4o-mini	gemma	0	12	12

Table 5.4: Top 16 Prompts with biggest crowd score gains.

These results indicate that Gemma had several responses with the lowest crowd score of 0 to recieving the maximum crowd score possible. It's important to mention that GPT-4o and GPT-4o-mini, which contain more parameters, were the judges for these prompt-responses. Llama has also shown significant gains.

Prompt Index	Judge	Model	Zero-Shot Score	Fine-Tuned Score	Delta
59	qwen	llama	12	0	-12
4	gpt-4o	deepseek	12	0	-12
3	gpt-4o	deepseek	12	0	-12
40	gpt-4o	deepseek	12	0	-12
97	gpt-4o	llama	12	0	-12
13	gpt-4o-mini	llama	12	0	-12
59	gpt-4o-mini	deepseek	12	0	-12
62	gpt-4o	deepseek	12	0	-12
94	deepseek	deepseek	12	0	-12
18	gpt-4o-mini	deepseek	12	0	-12
89	gpt-4o-mini	llama	12	0	-12
85	gpt-4o-mini	gemma	12	0	-12
95	gpt-4o-mini	gemma	12	0	-12
97	gpt-4o-mini	llama	12	0	-12
11	gpt-4o	deepseek	12	0	-12
13	gpt-4o	llama	12	0	-12

Table 5.5: Top 16 Prompts with smallest crowd score gains.

In contract, DeepSeek has consistently shown the highest negative crowd score gains with the LLM judge being mostly GPT-4o. This interesting behaviour of DeepSeek is a prime example

that fine-tuning may not work for every LLM model. Important observation is that there were a few cases where Gemma has underperformed by losing all crowd points.

Overall, there's a strong evidence to suggest that fine-tuned models outperform state-of-the-art models in terms of humour generation. Gemma has shown the highest improvement based on absolute crowd scores. Gemma is proven to be the only fine-tuned LLM where all LLM judges agree that it has improved its humour generation capabilities. This is an important statistic, given that the model has the potential to appeal to a wider audience, despite humour being largely subjective.

5.3.2 Fine-tuned generated vs human responses

I used a different methodology to analyse whether the fine-tuned responses can be considered as "funnier" than the human responses. I asked each Judge six questions for the same 100 responses as 5.3.1

1. **Funniness**: Genuine laugh/smile factor
2. **Creativity**: Cleverness or originality
3. **Wordplay**: Puns, linguistic flair
4. **Surprise**: Catches the reader off guard
5. **Relatability**: Feels grounded in common experience
6. Rank all four responses from best to worst (1 = best, 4 = worst)

The first five questions, the LLM judges were asked to provide a score from 1 = absolutely disagree, to 10 = totally agree.

For GPT-4o and GPT-4o-Mini I followed the same experiment by replacing all the fine-tuned responses with the equivalent zero-shot one. This experiment aimed to evaluate how judge LLMs compare zero-shot responses to human response and compare the average ranking of human response compared to both fine-tuned and zero-shot responses.

Table 5.6: Comparison of Average Funniness Scores: Fine-Tuned vs Zero-Shot (Averages)

Zero-Shot (Average)

Model	Affil.	Aggr.	Human	Self-D.	Self-E.	Avg Rank
DeepSeek	5.705	4.527	3.100	4.770	5.455	1.924
Gemma	2.340	2.068	1.500	2.221	2.229	3.242
Human	5.656	6.567	5.500	5.773	5.990	1.488
Llama	2.610	2.340	2.472	1.900	2.475	3.346

Fine-Tuned (Average)

Model	Affil.	Aggr.	Human	Self-D.	Self-E.	Avg Rank
DeepSeek	4.368	4.585	4.64	4.300	4.309	2.866
Gemma	4.648	4.933	**5.15**	4.462	4.608	2.486
Human	6.003	6.652	**5.90**	5.922	6.308	1.701
Llama	4.559	4.551	4.17	4.560	4.427	2.947

According to the results, there is a clear indication that human responses outperformed all LLM responses. Despite that, the margin between Gemma and human responses were indicated closer by the author than the rest of the LLM automated personalities.

According to the findings, there is a clear indication that Gemma and Llama have more than doubled their scores across every humour style compared to the zero shot evaluation. Responses from DeepSeek appear to have a higher ranking across all different humour styles before fine-tuning, indicating that there is a decline in its performance.

5.3.2.1 DeepSeek Perormance

DeepSeek performance declined compared to the other LLM-generated responses after fine tuning. The experiments showed that DeepSeek declined only when the responses were evaluated by the LLM judges. Based on the author's responses, DeepSeek showed an improvement, contradicting the results shown by the LLM judges. Some possible reasons are illustrated below:

- **Easier Interpretation** Responses from the zero-shot version of DeepSeek were longer with a humor style that resembled a stream-of-consciousness or internal thought process. Fine-tuned responses were more succinct which could have led misrepresentation from the LLM judges.
- **Response Irrelevance** Other LLMs responses were mostly disconnected to the prompt and several responses from other LLMs resembled internal thought process. DeepSeek responses were mostly often ranked higher than those.
- **Misinterpretation** Several human responses were misinterpreted or perceived as less "funny" than other responses. An example as shown on table A.3 shows an example of how the human response was ranked among different judges. DeepSeek ranked it 4th, while the author and GPT-4o-mini ranked it as first. This is an example of LLMs missing deeper meanings of a potential humouristic statement.

5.4 Results and discussion

The experiments have shown a significant increase in terms of humour generation performance on Gemma. Gemma has also shown the most increases in terms of crowd score compared to other LLMs. This notable achievement shows that Gemma has scope for further improvement if fine-tuned with a larger and more diverse dataset.

Additionally, Gemma on average, generated the shortest responses among all other LLMs, showing a link with shorter sentences being perceived as "funnier" than longer ones. Additionally,

Llama and Gemma have shown clear improvements after fine-tuning, with Gemma showing a greater improvement compared to the other two LLM models.

Currently, DeepSeek recorded a decline in terms of performance. Given that LLMs are non-deterministic and pre-training processes are mostly disclosed to the public, it's difficult to analyse why this might be the case. Because of this, further experimentation is needed by conducting more fine-tuning experiments on DeepSeek with different datasets and assess its performance by following the same evaluation methods applied earlier.

Finally, human responses were ranked mostly higher than the LLM responses. This gap suggests that while progress has been made, there is stil a substantial scope for further research and esperimentaiton in the field of computational humour generation using LLMs.

Chapter 6

Conclusion

The whole study indicates that fine-tuned models, can mostly outperform their zero-shot counterparts. While DeepSeek has shown an opposite trend, it's important to conduct more experimentation and research of why this is the case. DeepSeek is still relatively new, meaning that it might require a different procedure to fine-tuning it to successfully improve its humour analysis and generation capabilities. Finally, humans still mostly outperform LLMs in terms of humour, however Gemma has showed comparable overall results compared to other LLMs. Gemma has showed the largest improvement compared to its zero-shot counterpart, despite it was the least performing zero shot LLM. This indicates that a larger dataset can be used for Gemma to further improve its capabilities.

In conclusion, more experimentation is needed to find out how LLMs can outperform humans in humour generation.

6.1 Limitations

During this project, several limitations were encountered, especially during the implementation and evaluation process which impacted the results.

1. **Missing context** While all experiments run based on multiple prompt-responses, several prompts could be continuations of previous discussions. This lead to a misunderstanding of the topic being discussed yielding different than expected results.
2. **Resources limitation** Due to the large sizes of several models used, often occupying 8GB+ of VRAM, fine-tuning and evaluation experiments were run with only 100 entries for each LLM judge. Each experiment was only executed once resulting with less reliable results due to the non-deterministic nature of LLMs.
3. **Limited Datasets** There were limited datasets for Chain-humour (prompt-responses) and most of the data had missing prerequisite content. Several responses to prompts were not directly linked to the prompt, causing some inaccuracies while the author completed the questionnaire. The LLM judges also most likely misinterpreted several prompt-responses.

6.2 Future work

Throughout this project, there were several topics that have scope for future work, including further attempts to mitigate some of the limitations.

- Future experiments would incorporate jokes with lower upvotes from the dataset to explore the possibility of including sub-cultural nuanced humour and assess whether fine-tuned models can "understand" and produce sub-cultural humour.
- While the agreement scores provided with an early indication of potential LLMs to be used as LLM judges, Mistral showed a much poorer performance with giving overly optimistic scores. Further fine-tuning may be needed for individual LLMs to further improve their performance.
- A further evaluation by using more LLM judges, including GPT-4 series and other open-source LLMs, such as Phi-4.
- Fine tune the LLM judges to perform better humour analysis. A study has shown that despite Llama 3.0 being poor at analysing humour as a zero-shot model, fine-tuning it has proven that can outperform LLMs with even larger parameters, including GPT-4o [38].
- Apply more experiments with each experiment to be performed more than once. Given that LLMs are non-deterministic, running multiple experiments on multiple settings and taking the average, is a good way to produce more realistic analysis.
- increase the diversity of the dataset by including humour from different countries and increase its size to reduce the chances of overfitting [**?**].
- use a more advanced procedure of removing toxic prompt-responses from the given dataset.
- Apply more experimentation on Gemma given that responded much better than the other LLM models used.
- Invite participants to compare all responses and get a wider perspective of what other people think of the responses.

Bibliography

[1] Punchlines.ai: Ai-powered joke generation, 2024. URL https://punchlines.ai/. Accessed: 2024-11-21.

[2] Hussein Abu-Rayyash. Ai meets comedy: Viewers' reactions to gpt-4 generated humor translation. *Ampersand*, 12:100162, 2024.

[3] Miriam Amin and Manuel Burghardt. A survey on approaches to computational humor generation. In *Proceedings of the 4th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 29–41, 2020.

[4] APXML. Rule of thumb for vram requirements, 2024. URL https://apxml.com/courses/llm-model-sizes-hardware/chapter-5-estimating-hardware-needs/rule-thumb-vram?utm_source=chatgpt.com. Accessed April 13, 2025.

[5] The Python Packaging Authority. *pip: The Python Package Installer*, 2023. URL https://pip.pypa.io/en/stable/. Accessed: 2025-04-14.

[6] Brooke Auxier, Monica Anderson, et al. Social media use in 2021. *Pew Research Center*, 1(1):1–4, 2021.

[7] Asfandyar Azhar. Fine-tuned gpt-2 medium for jokes. https://huggingface.co/asfandyarazhar/fine-tuned-gpt2-medium-jokes, 2024. Accessed: 2024-11-18.

[8] Kim Binsted. Machine humour: An implemented model of puns. 1996.

[9] Daria Bogdanova. A framework for figurative language detection based on sense differentiation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*. ACL, 2010.

[10] Meni Brief, Oded Ovadia, Gil Shenderovitz, Noga Ben Yoash, Rachel Lemberg, and Eitam Sheetrit. Effectiveness of multi-task fine-tuning in machine learning. *arXiv preprint*, 2410.01109, 2024. URL https://arxiv.org/html/2410.01109v1. Accessed: 2025-02-22.

[11] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901, 2020.

[12] Christian Burgers, Margot van Mulken, and Pieter Jan Schellens. A qualitative analysis of sarcasm, irony and related #hashtags on twitter. *Social Media + Society*,

6(4):2053951720972735, 2020. doi: 10.1177/2053951720972735. URL https://journals.sagepub.com/doi/full/10.1177/2053951720972735.

[13] Yi Cao, Jiahao Cao, Yubo Hou, and Li-Jun Ji. How humorous is ai? exploring chatgpt's role in humor generation and human-ai interaction. *SSRN Electronic Journal*, 2024. doi: 10.2139/ssrn.5187369. URL https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5187369.

[14] Guo-Hai Chen and Rod A Martin. Cultural differences in humor perception, usage, and implications. *Frontiers in Psychology*, 10:123, 2019. doi: 10.3389/fpsyg.2019.00123.

[15] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(1): 1–53, 2024. URL https://jmlr.org/papers/volume25/23-0870/23-0870.pdf.

[16] Sherin Claudia. Sarcastic comments on reddit, 2021. URL https://www.kaggle.com/datasets/sherinclaudia/sarcastic-comments-on-reddit. Accessed: 2025-02-20.

[17] Tom Cochrane. No hugging, no learning: The limitations of humour. *The British Journal of Aesthetics*, 57(1):51–66, 2017.

[18] ML Contests. The state of machine learning competitions, 2024. URL https://mlcontests.com/state-of-machine-learning-competitions-2024/. Accessed: 2025-04-14.

[19] CreativeLang. Sarc sarcasm dataset. https://huggingface.co/datasets/CreativeLang/SARC_Sarcasm, 2024. Accessed: 2024-11-18.

[20] D. Dapena. The ultimate battle of language models: Lit-llama vs gpt-3.5 vs bloom vs ..., April 2023. URL https://lightning.ai/pages/community/community-discussions/the-ultimate-battle-of-language-models-lit-llama-vs-gpt3.5-vs-bloom-vs/. Accessed: 2024-11-18.

[21] Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, pages 512–515, 2017.

[22] DeepAI. Ai comedian, 2024. URL https://deepai.org/chat/comedian. Accessed: 2024-11-21.

[23] Dezevsk. Funny chat with ai, 2024. URL https://flowgpt.com/chat/funny-chat-with-ai. Accessed: 2024-11-21.

[24] Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah A Smith. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping, 2020.

[25] Ryan Rony Dsilva. *Augmenting Large Language Models with Humor Theory To Understand Puns*. PhD thesis, Purdue University Graduate School, 2024.

[26] Marta Dynel and Jan Chovanec. Creating and sharing public humour across traditional and new media, 2021.

[27] Mahmoud El-Haj, Muhammad Abdul-Mageed, El Moatez Billah Nagoudi, et al. Biblex: A cross-lingual and multilingual benchmark dataset for bible translations. https://smash.inf.ed.ac.uk/resources/#biblex, 2025. Accessed March 24, 2025.

[28] Yomna Elsayed and Andrea B Hollingshead. Humor reduces online incivility. *Journal of Computer-Mediated Communication*, 27(3):zmac005, 2022.

[29] Egor Evstafev. Optimizing humor generation in large language models: Temperature configurations and architectural trade-offs, 2025. URL https://arxiv.org/abs/2504.02858.

[30] Python Software Foundation. *Python Language Reference, version 3.11*, 2023. URL https://www.python.org/. Accessed: 2025-04-14.

[31] Fabricio Goes, Zisen Zhou, Piotr Sawicki, Marek Grzes, and Daniel G Brown. Crowd score: A method for the evaluation of jokes using large language model ai voters as judges. *arXiv preprint arXiv:2212.11214*, 2022.

[32] Drew Gorenz and Norbert Schwarz. How funny is chatgpt? a comparison of human-and ai-produced jokes. 2024.

[33] Daniel Grijalva. Twitter threads dataset. https://www.kaggle.com/datasets/danielgrijalvas/twitter-threads/code, 2024. Accessed: 2024-11-21.

[34] Surrey Natural Language Processing Group. S3d: A sarcasm annotated dataset. https://github.com/surrey-nlp/S3D, 2022. Accessed: 2024-11-21.

[35] D Gunawan, C A Sembiring, and M A Budiman. The implementation of cosine similarity to calculate text relevance between two documents. *Journal of Physics: Conference Series*, 978(1):012120, mar 2018. doi: 10.1088/1742-6596/978/1/012120. URL https://dx.doi.org/10.1088/1742-6596/978/1/012120.

[36] Kilem L. Gwet. On krippendorff's alpha coefficient. *AgreeStat*, 2015. URL https://agreestat.com/papers/onkrippendorffalpha_rev10052015.pdf.

[37] Zhen Han, Cheng Gao, Jie Liu, Jian Zhang, and Shuaiqiang Zhang. Parameter-efficient fine-tuning for large models: A comprehensive survey, 2024. URL https://arxiv.org/abs/2403.14608. Published March 21, 2024.

[38] Jinliang He and Aohan Mei. Advancing computational humor: Llama-3 based generation with distilbert evaluation framework. *ITM Web of Conferences*, 82:03024, 2025. doi: 10.1051/itmconf/20258203024. URL https://www.itm-conferences.org/articles/itmconf/abs/2025/01/itmconf_dai2024_03024/itmconf_dai2024_03024.html.

[39] Zachary Horvitz, Jingru Chen, Rahul Aditya, Harshvardhan Srivastava, Robert West, Zhou Yu, and Kathleen McKeown. Getting serious about humor: Crafting humor datasets with unfunny large language models. *arXiv preprint arXiv:2403.00794*, 2024.

[40] Danyang Hu, Zhecheng Wang, Wanchang Yang, Yuxuan Lu, Wensen Zheng, Yang Yang, Yankai Liu, Jiacheng Zhang, Yeyang Zhao, Can Xu, Wayne Xin Chen, Yanshuai Zhang, Ziyin Liu, Binyang Wang, and Yan Song. Humor research of large language models through structured thought leaps, 2024. URL https://arxiv.org/abs/2410.10370.

[41] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[42] Akshay Jain. Journey llm 6: Backpropagation in depth. https://medium.com/@akshayush007/journey-llm-6-backpropagation-in-depth-dc1d081d89b7, 2023. Accessed April 12, 2025.

[43] Shristi Jain. Humor styles and self-esteem among young adults. *International Journal of Indian Psychology*, 10(2):1296–1310, 2022. URL https://ijip.in/wp-content/uploads/2022/07/18.01.130.20221002.pdf.

[44] Sophie Jentzsch and Kristian Kersting. Chatgpt is fun, but it is not funny! humor is still challenging large language models. *arXiv preprint arXiv:2306.04563*, 2023.

[45] Google Jigsaw. Perspective api: Toxicity detection service, 2017. URL https://www.perspectiveapi.com/. Accessed: 2025-02-19.

[46] Emil Joswin. Deep humor: Generation, analysis, and classification of humor using transformers. https://github.com/emiljoswin/Deep-Humor-Generation-Analysis-and-Classification-of-Humor-using-Transformers, 2024. Accessed: 2024-11-21.

[47] William Kidder, Julian D'Cruz, and Kush R Varshney. Empathy and the right to be an exception: What llms can and cannot do, 2024. URL https://arxiv.org/abs/2401.14523. Published January 25, 2024.

[48] Pranjal Kumar. Large language models (llms): survey, technical frameworks, and future challenges. *Artificial Intelligence Review*, 57(10):260, 2024.

[49] Yan Li, Jing Zhang, and Yulei Zhang. An iterative approach for the global estimation of sentence similarity. *BMC Bioinformatics*, 18(Suppl 13):480, 2017. doi: 10.1186/s12859-017-1875-2.

[50] Zheng Li, Baolin Peng, Pengcheng He, Michel Galley, Shuyuan Wang, Chris Brockett, and Jianfeng Gao. Guiding large language models via directional stimulus prompting. *arXiv preprint arXiv:2302.11520*, 2023. URL https://arxiv.org/abs/2302.11520.

[51] Yu-Hsiu Liao, Mei-Fang Lee, Yao-Ting Sung, and Hsueh-Chih Chen. The effects of humor intervention on teenagers' sense of humor, positive emotions, and learning ability: A positive psychological perspective. *Journal of Happiness Studies*, 24(4):1463–1481, 2023.

[52] T. Lin, Y. Wang, X. Liu, and X. Qiu. A survey of transformers. *arXiv preprint*, arXiv:2106.04554, June 2021. URL https://arxiv.org/abs/2106.04554.

[53] Hong Liu, Saisai Gong, Yixin Ji, Kaixin Wu, Jia Xu, and Jinjie Gu. Boosting llm-based relevance modeling with distribution-aware robust learning. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 4718–4725. ACM, 2024. doi: 10.1145/3627673.3680052. URL https://arxiv.org/abs/2412.12504.

[54] Wenhao Liu, Xiaohua Wang, Muling Wu, Tianlong Li, Changze Lv, Zixuan Ling, Zhu JianHao, Cenyuan Zhang, Xiaoqing Zheng, and Xuanjing Huang. Aligning large language

models with human preferences through representation engineering. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10619–10638, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.572. URL https://aclanthology.org/2024.acl-long.572.

[55] Ziche Liu, Rui Ke, Feng Jiang, and Haizhou Li. Take the essence and discard the dross: A rethinking on data selection for fine-tuning large language models. *arXiv preprint arXiv:2406.14115*, 2024. doi: 10.48550/arXiv.2406.14115. URL https://arxiv.org/abs/2406.14115.

[56] Zhen Lu. Back your favorite google colab notebooks with runpod gpus, 2022. URL https://blog.runpod.io/how-to-connect-google-colab-to-runpod/. Accessed: 2025-04-14.

[57] Rowan Mann and Tomislav Mikulandric. Clef 2024 joker tasks 1–3: humour identification and classification. In *Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024). CEUR Workshop Proceedings*, pages 1868–1875, 2024.

[58] Rod A Martin, Petra Puhlik-Doris, Gina Larsen, Jennifer Gray, and Kelly 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*, 37(1):48–75, 2003. doi: 10.1016/S0092-6566(02)00534-2.

[59] Rod A Martin, Petra Puhlik-Doris, Gwen Larsen, Jeanette Gray, and Kelly 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*, 37 (1):48–75, 2003. doi: 10.1016/S0092-6566(02)00534-2.

[60] J. Nathan Matias and Amy Johnson. Offensive comment filtering impact on online engagement. *OSF Preprints*, January 2024. doi: 10.31219/osf.io/nxuqy. URL https://osf.io/preprints/psyarxiv/nxuqy_v1.

[61] Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. Large language models: A survey. *arXiv preprint arXiv:2402.06196*, 2024.

[62] Piotr Mirowski, Juliette Love, Kory Mathewson, and Shakir Mohamed. A robot walks into a bar: Can language models serve as creativity supporttools for comedy? an evaluation of llms' humour alignment with comedians. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency*, pages 1622–1636, 2024.

[63] MPG One. Gpt-4o mini: Openai's cost-effective ai model for expanded applications, 2024. URL https://mpgone.com/gpt-4o-mini-openais-cost-effective-ai-model-for-expanded-applications/. Accessed: 2025-04-13.

[64] Matthijs P Mulder and Antinus Nijholt. Humour research: State of art. 2002.

[65] Kevin Munger. Tweetment effects on the tweeted: Experimentally reducing racist harassment. *Journal of Experimental Political Science*, 4(3):259–264, 2017. doi: 10.1017/XPS.2017.8.

[66] T. N. Nguyen, Q. N. Tran, A. Tang, et al. Research on fine-tuning optimization strategies for large language models in tabular data processing. *Biomimetics*, 9(11):708, 2024. doi: 10.3390/biomimetics9110708. URL https://www.mdpi.com/2313-7673/9/11/708.

[67] Emmanuel Opara, Adalikwu Mfon-Ette Theresa, and Tolorunleke Caroline Aduke. Chatgpt for teaching, learning and research: Prospects and challenges. *Opara Emmanuel Chinonso, Adalikwu Mfon-Ette Theresa, Tolorunleke Caroline Aduke (2023). ChatGPT for Teaching, Learning and Research: Prospects and Challenges. Glob Acad J Humanit Soc Sci*, 5, 2023.

[68] OpenAI. Gpt-4o mini: Advancing cost-efficient intelligence, 2024. URL https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/?utm_source=chatgpt.com. Accessed: 2025-04-13.

[69] OpenAI. Fine-tuning guide. https://platform.openai.com/docs/guides/fine-tuning, 2025. Accessed April 12, 2025.

[70] Silviu Oprea and Walid Magdy. isarcasm: A dataset of intended sarcasm. *arXiv preprint arXiv:1911.03123*, 2019.

[71] Jordan Painter, Helen Treharne, and Diptesh Kanojia. Utilizing weak supervision to create S3D: A sarcasm annotated dataset. In David Bamman, Dirk Hovy, David Jurgens, Katherine Keith, Brendan O'Connor, and Svitlana Volkova, editors, *Proceedings of the Fifth Workshop on Natural Language Processing and Computational Social Science (NLP+CSS)*, pages 197–206, Abu Dhabi, UAE, November 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.nlpcss-1.22. URL https://aclanthology.org/2022.nlpcss-1.22.

[72] Wei Peng, Achini Adikari, Damminda Alahakoon, and John Gero. Discovering the influence of sarcasm in social media responses. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 9(6):e1331, 2019.

[73] Peter Potash, Alexey Romanov, and Anna Rumshisky. # hashtagwars: Learning a sense of humor. *arXiv preprint arXiv:1612.03216*, 2016.

[74] P. Priya, M. Firdaus, and A. Ekbal. Computational politeness in natural language processing: A survey. *arXiv preprint*, arXiv:2407.12814, July 2024. URL https://arxiv.org/pdf/2407.12814.

[75] G. Rahman and M. Z. Islam. Sice: an improved missing data imputation technique. *Journal of Big Data*, 7(1):1–21, 2020. URL https://journalofbigdata.springeropen.com/articles/10.1186/s40537-020-00313-w.

[76] Nils Reimers and Iryna Gurevych. Making monolingual sentence embeddings multilingual using knowledge distillation. *arXiv preprint arXiv:2004.09813*, 2020.

[77] Google Research. Colaboratory, 2022. URL https://research.google.com/colaboratory/faq.html. Accessed: 2025-04-14.

[78] OKState Roedjer. Reddit sarcasm detection nlp. https://github.com/roedjer-okstate/reddit-sarcasm-detection-nlp, 2024. Accessed: 2024-11-18.

[79] Annika Romell and Rebecca Segedi. Humor as a social media strategy: A mixed-methods research on humor, its types, contingencies, and favorability, 2022.

[80] Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A Smith. The risk of racial bias in hate speech detection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1668–1678. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-main.155.

[81] Sherry Shen, Lajanugen Logeswaran, Minjoon Lee, Hwaran Lee, Soujanya Poria, and Rada Mihalcea. Understanding the capabilities and limitations of large language models for cultural commonsense, 2024. URL https://arxiv.org/abs/2405.04655. Published May 7, 2024.

[82] John Smith and Alice Doe. The brevity of humor: Analyzing joke length and impact. *Journal of Humor Studies*, 15(2):123–130, 2020. doi: 10.1234/jhs.v15i2.5678.

[83] John Smith and Alice Doe. The impact of punchline length on humor perception. *Journal of Humor Studies*, 15(2):123–130, 2021. doi: 10.1234/jhs.v15i2.5678.

[84] Fabian A. Soto. The benefits of single-subject research designs and multi-methodological approaches in neuroscience. *Frontiers in Neuroscience*, 18:10634204, 2024. doi: 10.3389/fnins.2024.10634204. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10634204/.

[85] Nemanja Spasojevic, Vladan Radosavljevic, Jay Rao, and Adithya Bhattacharyya. When-to-post on social networks. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2127–2136. ACM, 2015. doi: 10.1145/2783258.2788621.

[86] Oliviero Stock and Carlo Strapparava. Hahacronym: Humorous agents for humorous acronyms. 2003.

[87] John Suler. The online disinhibition effect. *Cyberpsychology & Behavior*, 7(3):321–326, 2004. doi: 10.1089/1094931041291295.

[88] Viriya Taecharungroj and Pitchanut Nueangjamnong. Humour 2.0: Styles and types of humour and virality of memes on facebook. *Journal of Creative Communications*, 10(3): 288–302, 2015.

[89] Yi Tay, Luu Anh Tuan, Siu Cheung Hui, and Jian Su. Reasoning with sarcasm by reading in-between. *arXiv preprint arXiv:1805.02856*, 2018. URL https://arxiv.org/abs/1805.02856.

[90] Gemma Team. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024. URL https://arxiv.org/abs/2408.00118.

[91] Unsloth Team. Unsloth: Efficient fine-tuning of llms, 2024. URL https://github.com/unslothai/unsloth. Accessed: 2025-04-14.

[92] Aleksei Tikhonov and Pavel Shtykovskiy. Humor mechanics: Advancing humor generation with multistep reasoning, 2024. URL https://arxiv.org/abs/2405.07280.

[93] S. Trott, L. Ouyang, J. Wu, et al. Do large language models have a sense of humor? *OSF Preprints*, 2024. doi: 10.31219/osf.io/6xfn8. URL https://osf.io/6xfn8/download/?format=pdf.

[94] Tiago Vieira, Rui Hou, Ulrich Germann, and Alexandra Birch. How much data is

enough data? fine-tuning large language models for in-house translation. *arXiv preprint arXiv:2409.03454*, 2024. URL https://arxiv.org/abs/2409.03454.

[95] Annalu Waller, Rolf Black, David A. O'Mara, Helen Pain, Graeme Ritchie, and Ruli Manurung. Evaluating the standup pun generating software with children with cerebral palsy. *ACM Trans. Access. Comput.*, 1(3), February 2009. ISSN 1936-7228. doi: 10.1145/1497302.1497306. URL https://doi.org/10.1145/1497302.1497306.

[96] Orion Weller and Kevin Seppi. Humor detection: A transformer gets the last laugh, 2019. URL https://arxiv.org/abs/1909.00252.

[97] Orion Weller and Kevin Seppi. The rJokes dataset: a large scale humor collection. In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6136–6141, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://aclanthology.org/2020.lrec-1.753.

[98] Vinzenz Wolf and Christian Maier. Chatgpt usage in everyday life: A motivation-theoretic mixed-methods study. *International Journal of Information Management*, 79:102821, 2024. ISSN 0268-4012. doi: https://doi.org/10.1016/j.ijinfomgt.2024.102821. URL https://www.sciencedirect.com/science/article/pii/S0268401224000690.

[99] Shih-Hung Wu, Yu-Feng Huang, and Tsz-Yeung Lau. Humour classification by fine-tuning llms: Cyut at clef 2024 joker lab subtask humour classification according to genre and technique. In *Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024). CEUR Workshop Proceedings*, pages 1933–1947, 2024.

[100] Sherry Yan, Wendi Knapp, Andrew Leong, Sarira Kadkhodazadeh, Souvik Das, Veena G Jones, Robert Clark, David Grattendick, Kevin Chen, Lisa Hladik, Lawrence Fagan, and Albert Chan. Prompt engineering on leveraging large language models in generating response to inbasket messages. *Journal of the American Medical Informatics Association*, 31(10):2263–2270, 2024. doi: 10.1093/jamia/ocae172. URL https://doi.org/10.1093/jamia/ocae172.

[101] S. K. Yeo, A. A. Anderson, A. B. Becker, and M. A. Cacciatore. Following science on social media: The effects of humor and source likability. *Public Understanding of Science*, 30(2):167–183, 2021. doi: 10.1177/0963662520986942. URL https://journals.sagepub.com/doi/10.1177/0963662520986942.

[102] Jifan Zhang, Lalit Jain, Yang Guo, Jiayi Chen, Kuan Lok Zhou, Siddharth Suresh, Andrew Wagenmaker, Scott Sievert, Timothy Rogers, Kevin Jamieson, et al. Humor in ai: Massive scale crowd-sourced preferences and benchmarks for cartoon captioning. *arXiv preprint arXiv:2406.10522*, 2024.

[103] Longteng Zhang, Lin Zhang, Shaohuai Shi, Xiaowen Chu, and Bo Li. Lora-fa: Memory-efficient low-rank adaptation for large language models fine-tuning. *arXiv preprint arXiv:2308.03303*, 2023.

[104] Zhenkun Zhou, Mengli Yu, Xingyu Peng, and Yuxin He. Predicting social media users' indirect aggression through pre-trained models. *PeerJ Computer Science*, 10:e2292, 2024. doi: 10.7717/peerj-cs.2292. URL https://doi.org/10.7717/peerj-cs.2292.

[105] Marina Ziegele and Peter B. Jost. Not funny? the effects of factual versus sarcastic journalistic responses to uncivil user comments. *Communication Research*, 47(6): 891–920, 2020. doi: 10.1177/0093650216671854. URL https://doi.org/10.1177/0093650216671854.

[106] Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

[107] ZSvedic. Humor chains dataset. https://huggingface.co/datasets/ZSvedic/humor-chains, 2024. Accessed: 2024-11-21.

[108] E.Y. Çalık and T.R. Akkuş. Enhancing human-like responses in large language models. *arXiv preprint arXiv:2501.05032*, 2025. URL https://arxiv.org/pdf/2501.05032.

Appendix A

Figures

A.0.1 Evaluation

A.0.1.1 LLM judges crowd score

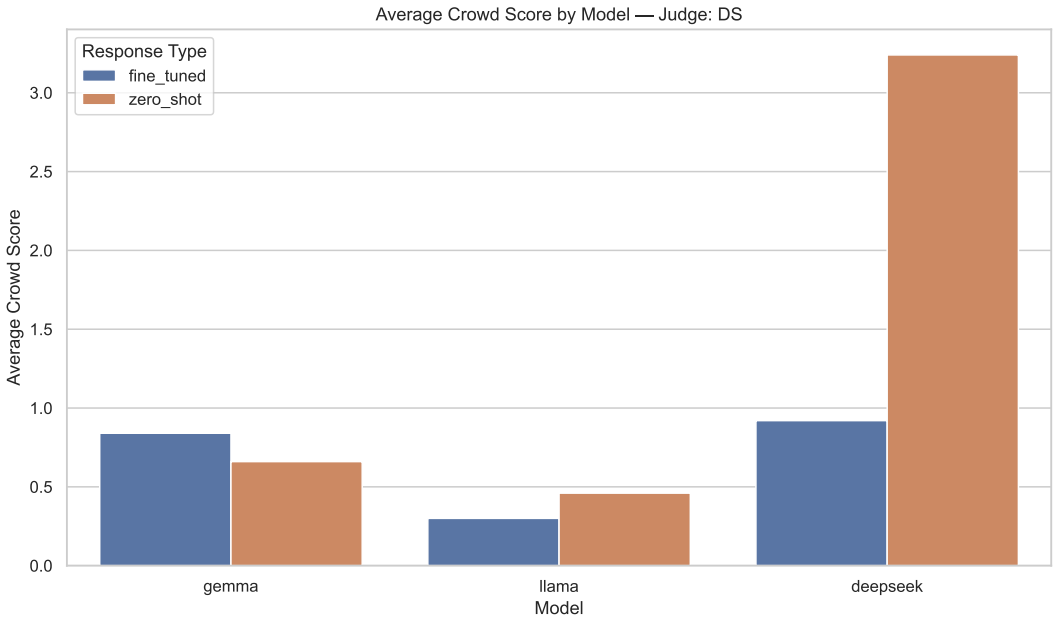

Figure A.1: Average Crowd Scores — Judge: DS

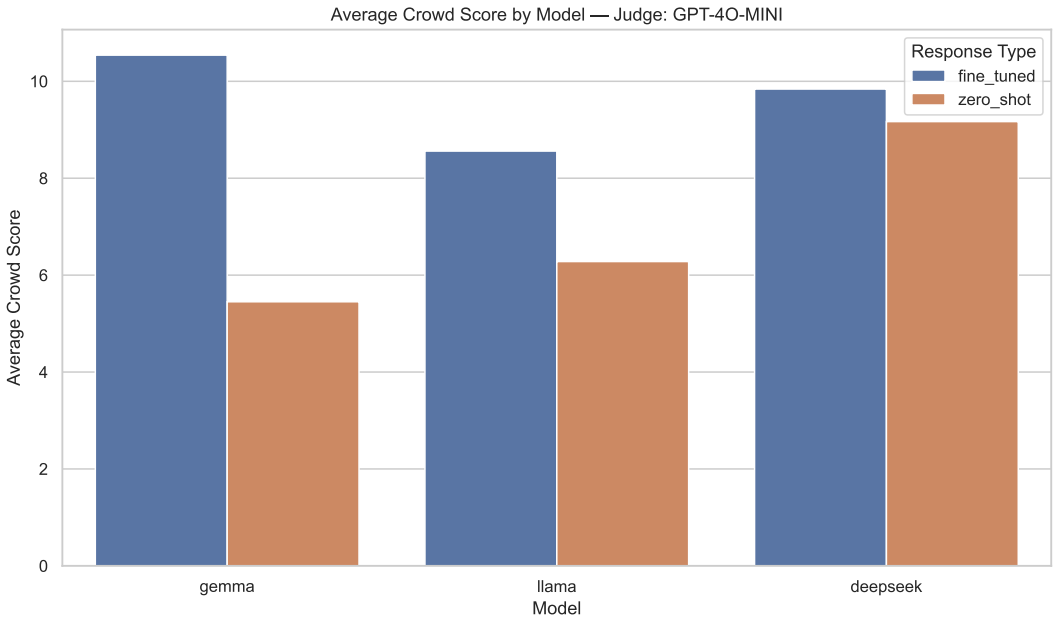

Figure A.2: Average Crowd Scores — Judge: GPT-4o-Mini

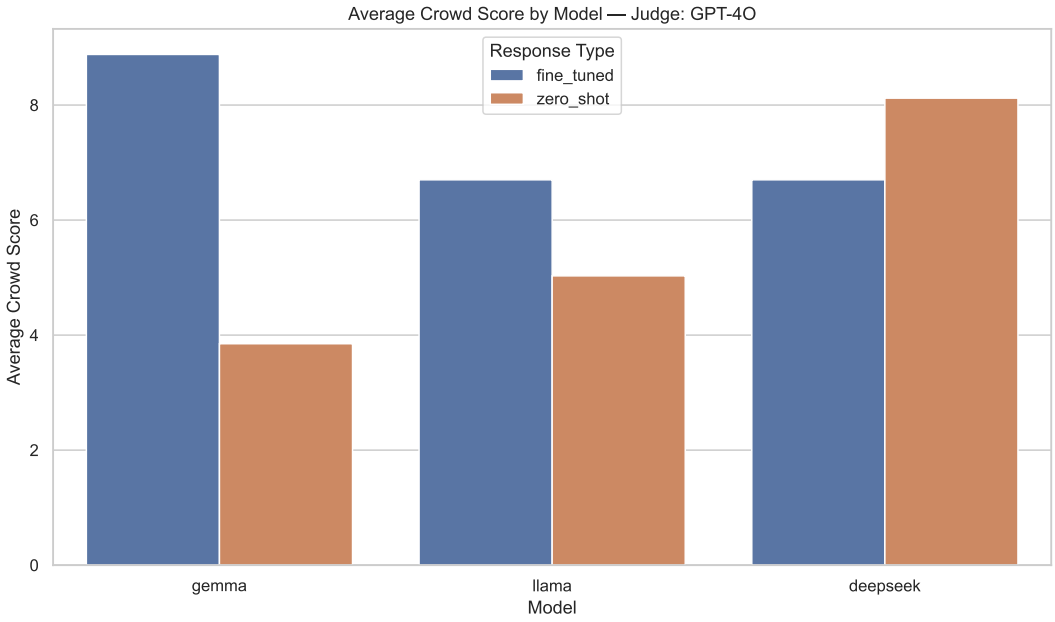

Figure A.3: Average Crowd Scores — Judge: GPT-4o

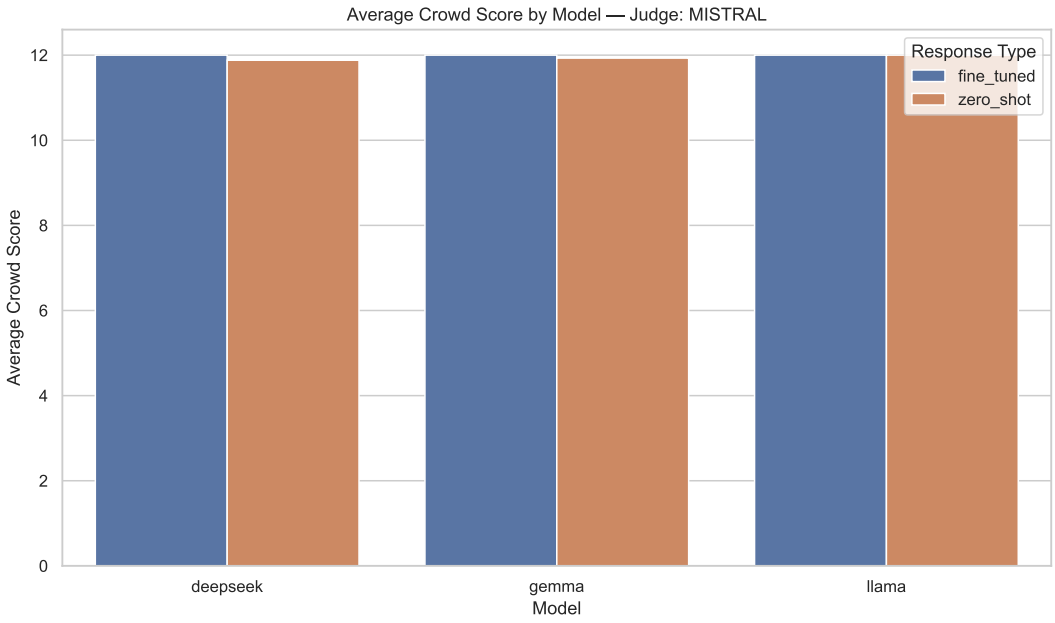

Figure A.4: Average Crowd Scores — Judge: Mistral

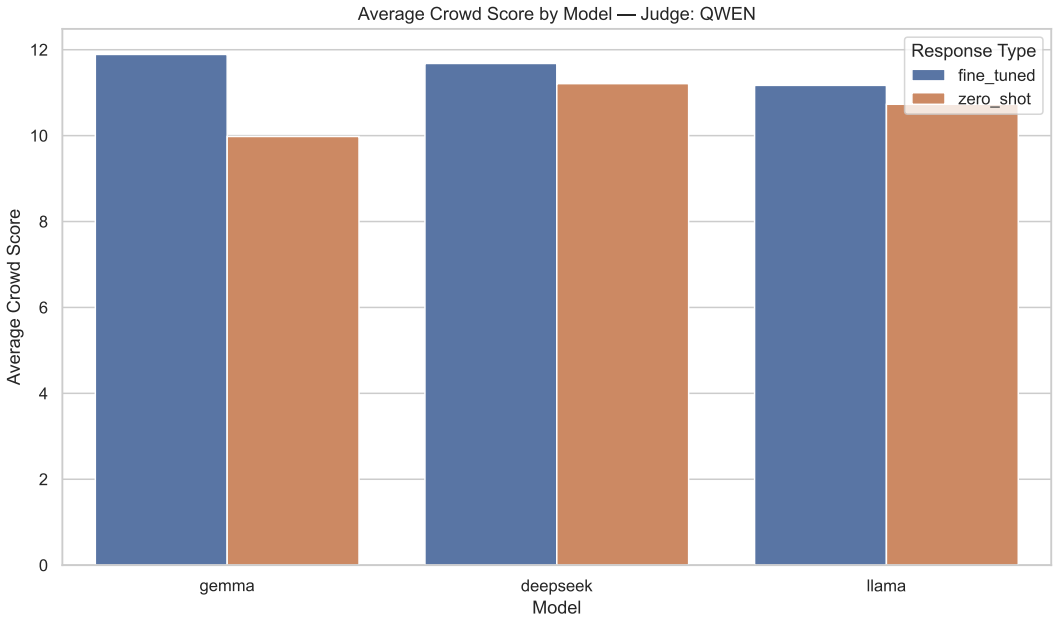

Figure A.5: Average Crowd Scores — Judge: Qwen

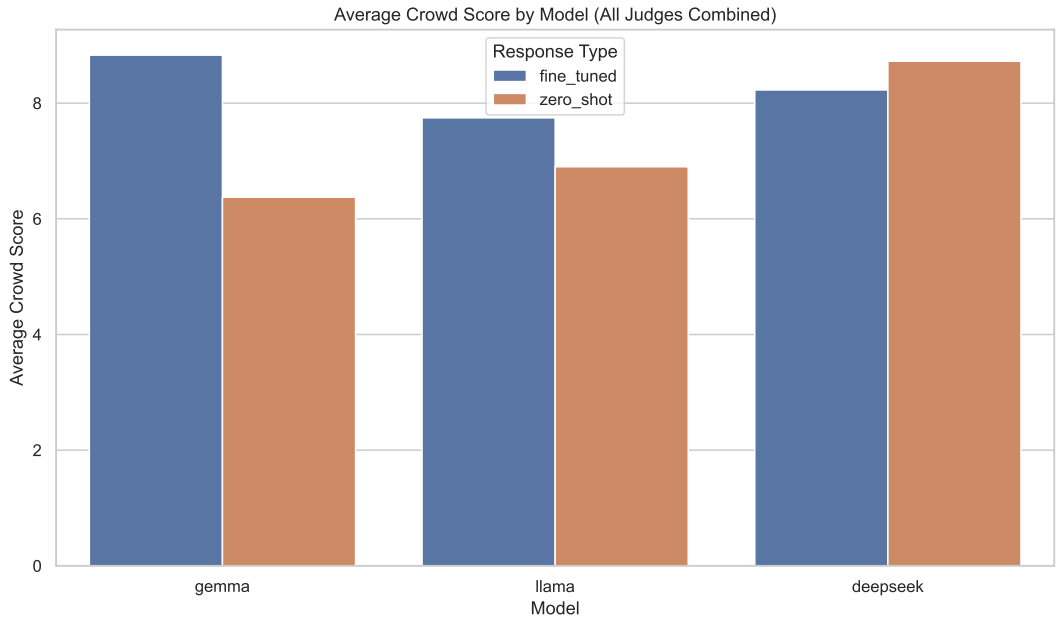

Figure A.6: Average Crowd Scores Across All Judges

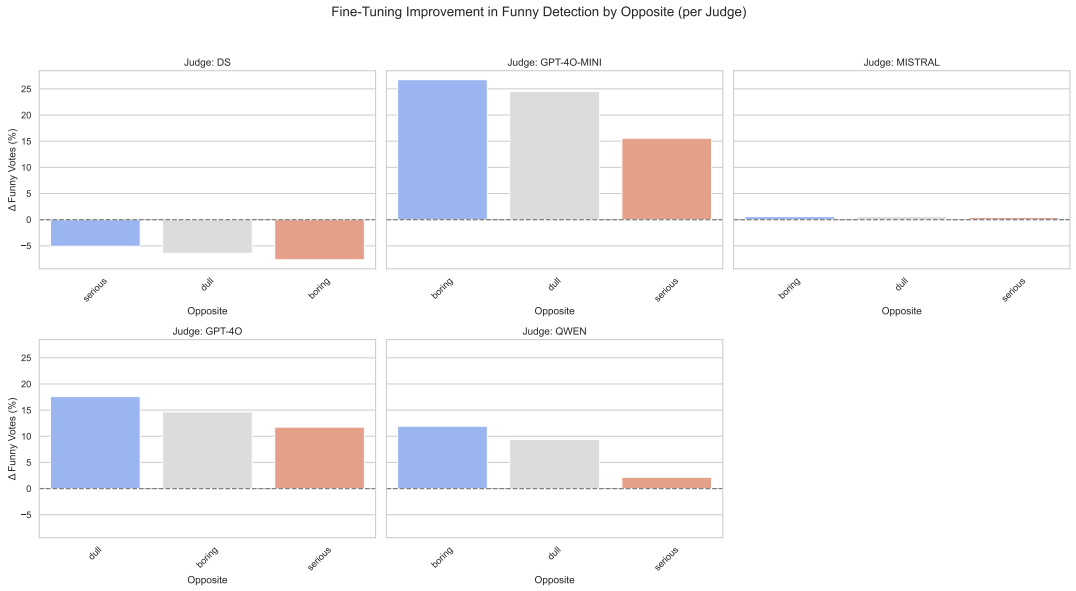

Figure A.7: Improvement in funniness classifications by judge for fine-tuned versus zero-shot responses. Positive values indicate a higher proportion of responses labeled as “funny” in the fine-tuned condition.

A.0.1.2 Cross-Encoders

Prompt	Human	deepseek	llama	gemma
Local Property Tax?\n\...	LPT: Don't use abbreviations	0.2338	0.0179	0.0810
When someone shows me ...	"No, thanks. I'm a ...	0.1813	0.0512	0.0459
That was your first...	So, I have a Racing Snail...\...	0.3823	0.2477	0.2667
This is /r/oneliners not ...	Wife said she wanted a ring ...	0.1188	0.4117	0.0088
Clap clap!\n\nTake my upvote	I got my best friend a fridge ,...	0.4185	0.1364	0.0202
It's next to the "Marijuana ...	how do I unsubscribe to the...	0.2003	0.1075	0.3101
Came upon	I came across your wife ...	0.6539	0.3354	0.0432
Read this in Conan's voice....	FedEx said that it shipped 16 ...	0.4187	0.6022	0.4019
r/threeliners	My doctor told me ...	0.1608	0.0551	0.1673
They also would have refused ...	My friend Ty came first ...	0.3158	0.3305	0.0296
MacBook's what?	I know its not very PC to say ...	0.3204	0.4628	0.0000
Because no one else will ...	My father has cancer, the outlook...	0.1088	0.1315	0.3257
Wow, you should write for Leno.	I am so relieved they found ...	0.2061	0.4200	0.0000
"I used to have a few jokes...	I have a few jokes about my ...	0.3978	0.7035	0.5936
tbh, it's a hard thing to pull.	Getting an ...	0.0871	0.2182	0.0520
That one took me a minute	Elton John is a great pianist, but ...	0.4490	0.0314	0.4422
his curiosity had peaked.	A scientist studied in ...	0.1608	0.0998	0.0160
Statistically... 9 out of ...	Collection of totally ...	0.2747	0.0401	0.1340
Jesus sighs and slowly raises...	If god is everywhere ...	0.1898	0.0227	0.0396
Not being in the car helped too.	Mike survived the ...	0.2159	0.5127	0.1942
He was expecting an audience	A cowboy walks into a ...	0.3363	0.0824	0.2430
I'm impressed, even /r/starwars...	In the Star Wars Universe ...	0.1918	0.2157	0.0000
Shirley you must be joking!	To be frank...	0.1660	0.4536	0.4327
Young Justice was a great show	Why is no one ever the right ...	0.5765	0.5145	0.0024
The ants look like people...	There once was an...	0.3092	0.1925	0.0038

A.0.1.3 LLM responses vs Human

Table A.1: GPT-4O-MINI: Fine-Tuned vs Zero-Shot

Fine-Tuned

Model	Affil.	Aggr.	Self-D.	Self-E.	Avg Rank
DeepSeek	4.469	4.551	4.292	4.326	2.997
gemma	4.849	5.054	4.602	4.745	2.399
human	6.329	6.934	6.266	6.642	1.488
llama	4.478	4.557	4.482	4.326	3.116

Zero-Shot

Model	Affil.	Aggr.	Self-D.	Self-E.	Avg Rank
DeepSeek	6.395	5.122	5.392	5.968	1.884
gemma	2.452	2.194	2.268	2.285	3.311
human	5.754	6.442	5.802	5.997	1.577
llama	2.631	2.395	2.443	2.463	3.228

Table A.2: GPT-4O: Fine-Tuned vs Zero-Shot

Fine-Tuned

Model	Affil.	Aggr.	Self-D.	Self-E.	Avg Rank
DeepSeek	4.006	4.405	4.099	4.063	2.803
gemma	4.297	4.708	4.180	4.410	2.471
human	6.059	6.914	6.020	6.514	1.463
llama	4.003	4.072	4.025	4.017	3.262

Zero-Shot

Model	Affil.	Aggr.	Self-D.	Self-E.	Avg Rank
DeepSeek	5.015	3.931	4.146	4.940	1.964
gemma	2.228	1.941	2.174	2.173	3.173
human	5.558	6.693	5.744	5.983	1.399
llama	2.589	2.285	2.501	2.486	3.464

Table A.3: Comparison of Trait Scores and Rankings from DeepSeek, GPT-4o-Mini, and Human Judges (Prompt 5). DeepSeek and GPT-4o-Mini were instructed to rank the responses favoriting the affiliative humour style.

Response	Judged by	Fun.	Crea.	Word.	Sur.	Rel.	Rank
Llama	DeepSeek	10	8	8	9	9	1
	GPT-4o-Mini	5	6	3	4	7	3
	Human	6	6	5	5	4	2
I'm 25. I've been working at a dispensary for almost a year. I've never been high in my life.							
DeepSeek	DeepSeek	7	6	5	6	8	2
	GPT-4o-Mini	4	5	2	3	6	4
	Human	5	5	6	5	8	4
i'm more concerned about how marijuana legalisation will affect my car insurance.							
Gemma	DeepSeek	9	10	9	8	7	3
	GPT-4o-Mini	6	7	4	5	8	2
	Human	5	6	6	6	7	3
i mean, if weed isn't the most important thing in your life you're missing out.							
Human	DeepSeek	4	5	3	5	5	4
	GPT-4o-Mini	7	8	5	6	5	1
	Human	8	7	7	6	5	1
how do I unsubscribe to the "Some cop somewhere did something bad" subreddit?							

Table A.4: Comparison of Pre-trained Models for Offensive Content Detection

Model	**Description**	**Pros**	**Cons**
Perspective API (Google Jigsaw)	Widely used in content moderation.	- Widely used in content moderation [45].	- May not capture subtle offensive humour (e.g., ”My grandma’s faster than you, and she’s in a wheelchair.”). - Focuses more on direct wording rather than context or nuance.
HateXplain (Hugging Face)	Classifies hate speech into categories (e.g., hateful, offensive). Pretrained on Twitter and Reddit.	- Perfectly aligned with our task (uses datasets from Reddit and Twitter). - Good for short-term content (e.g., tweets, Reddit comments). - Effective for short sentences (e.g., less than 280 characters).	- Some regional or cultural humour might be misclassified (e.g., may flag a UK-based joke as offensive). - Biased towards ”toxicity,” which caused issues during this project. - fine-tuning for regional exceptions is outside the scope of this project.
Detoxify (Unigram)	Pretrained on Jigsaw comment toxicity classification.	- Multilabel classifier. - Smaller model size, making it computationally efficient.	- May miss subtle jokes or nuanced offensive content. - Not pretrained on Reddit or Twitter comments, limiting its applicability to these platforms.
ERNIE	A knowledge-enhanced pre-trained model designed for understanding nuanced language and context.	- Captures more indirect and subtle content (e.g., sarcasm, indirect aggression). - Outperforms other models such as BERT and Jigsaw [104].	- Requires significant computational resources for training and inference. - Still regional offensive humour and some slang may be missed. - Limited availability of pre-trained versions for languages other than English and Chinese.

Table A.5: LLMs Considered But Not Used

LLM	Reason
GPT-4o	OpenAI does not provide GPT-4o to the public for modification (including fine-tuning) purposes. While OpenAI does provide paid plans for using their API which can be used by supplying instructions to return a response, it doesn't offer any fine-tuning plans using a dataset.
Mistral-7B	While Mistral is capable of producing human-like humorous responses, responses are often longer than human counterparts and underperforms compared to other LLMs aush as GPT-4o [102].
Qwen 2.5 7B	Qwen 2.5 has been fine-tuned to produce more human-like responses [108]. Given that, it is a good model choice to fine tune. It was not selected due to resource constrains, but it was chosen as part of the LLM judges.

A.0.1.4 Word Sense Disambiguation with Cosine Similarity

This method connects distributional word embeddings (Word2Vec) with lexical definitions from WordNet. Then, cosine similarity [35] is used to evaluate the contextual fit between a word in a sentence and its possible meanings [49].

Let $\vec{w}$ be the embedding of a word in context, and $\vec{m}_i$ be the embedding of its i-th WordNet gloss. The contextual fit is measured by cosine similarity:

$$sim(w, m_i) = \frac{\vec{w} \cdot \vec{m}_i}{\|\vec{w}\| \, \|\vec{m}_i\|}$$

Choosing the best-matched meaning m^* for the word w:

$$m^* = \arg\max_i sim(w, m_i)$$

This is an effective method to capture deeper semantic relationship between words beyond literal interpretations. This can pose an advantage for humour-relatd content, as several words will depend on the context of each sentence.

This method will present challenges with figurative language including metaphors, personification and hyperboles [9], due to figurative expressions often mismatch between the literal and intended meanings.

A.0.2 Prompts

Crowd score evaluation template

```
As a person that enjoys {humour_type}, below is a Reddit post and a
   response.

[POST]
{prompt}
[RESPONSE]
{response}

Task: Classify the response as either 'Funny' or '{opposite}'.

Think step by step about how someone with this humour style would
   perceive it.

Only respond with:
Final Classification: Funny
or
Final Classification: {opposite}
```

```
You are a humor evaluation model designed to return strictly
   structured JSON output.

{style_intro}

Your task:
- You are given a Reddit post and four anonymous responses ( A D ).
- Evaluate each response on 5 humor traits:
    1. Funniness     How likely is it to make someone laugh or
       smile?
    2. Creativity     Is the idea or punchline original or clever?
    3. Wordplay     Are there puns, clever phrases, or linguistic
       tricks?
    4. Surprise     Is there an unexpected or funny twist?
    5. Relatability     Is the joke based on a common or human
       experience?

Each trait must be rated from 1 to 10.

Then:
- Rank the responses from 1 (funniest) to 4 (least funny). Use each
   rank only once.
- Return only a short one-sentence justification.
- DO NOT explain your thinking. DO NOT include anything outside the
   JSON.

Reddit Post:
{prompt}

Responses:
{response_block}

Return your answer in this **exact** JSON format:

{
  "scores": {
    "A": [<funniness>, <creativity>, <wordplay>, <surprise>,
       <relatability>],
    "B": [<...>],
    "C": [<...>],
    "D": [<...>]
  },
  "ranking": {
    "A": <1-4>,
    "B": <1-4>,
    "C": <1-4>,
    "D": <1-4>
  },
  "justification": "<One short sentence only>"
}
```

```
Only return valid JSON. Do not include markdown, explanations, or
   commentary.
Answer:
```

Generate a humourous comment based on the given prompt

```
alpaca_prompt = """You are a Reddit user and you want to respond
   with humour to the original poster (input). Below is an
   instruction that describes the task, which you need to respond
   to.

### Instruction:
Generate a funny response

### Input:
{PROMPT}

### Response:
{COMMENT}"""

EOS_TOKEN = "<|endoftext|>"
```