The Role of Role Design in In-Context Learning for Large Language Models

Anonymous ACL submission

Abstract

In-context learning (ICL) enables Large Language Models (LLMs) to generate predictions based on prompts without additional fine-tuning. While prompt engineering has been widely studied, the impact of role design within prompts remains underexplored. This study examines the influence of role configurations in zero-shot and few-shot learning scenarios using GPT-3.5 and GPT-4o from OpenAI and Llama2-7b Llama2-13b from Meta. We evaluate the models' performance across datasets, focusing on tasks like sentiment analysis, text classification, and question answering. F1 scores are used to measure the effectiveness of different role designs. Our findings highlight the potential of role-based prompt structuring to enhance LLM performance, offering new insights for optimizing prompt design strategies in natural language processing tasks.

1 Introduction

In recent years, the field of Large Language Models (LLMs) has seen remarkable advancements. Models such as GPT-3 (Brown et al., 2020) and Llama (Touvron et al., 2023) have showcased impressive capabilities in various natural language tasks, including question answering (Rouzegar and Makrehchi, 2024b), sentiment analysis (Zhang et al., 2023), and text classification (Abburi et al., 2023).

A new paradigm known as ICL has emerged in natural language processing (NLP) (Dong et al., 2022). In ICL, LLMs generate predictions based on provided prompts, which usually include a few training examples. This approach has become a trend for evaluating and extending the abilities of LLMs, allowing them to generalize to new, unseen cases without the need for additional fine-tuning (Dong et al., 2022).

Although several studies have discussed various ICL and prompt engineering methods, the impact of role design in ICL has not been thoroughly explored. Role design involves structuring prompts with distinct roles, such as system instructions, user inputs, and assistant responses. Understanding how these roles influence model performance can provide valuable insights for optimizing prompt engineering strategies.

In this paper, we examine the impact of role design in zero-shot and few-shot experiments using two prominent instruction-tuned LLMs: GPT-3.5 and GPT-4o from OpenAI and Llama2-7b and Llama2-13b from Meta. We systematically evaluate the models' performance across different natural language tasks, including sentiment analysis, text classification, and question answering. By employing F1 scores as our primary evaluation metric to measure how closely the outputs match the labels and Structural Accuracy as a secondary metric to measure how closely the outputs match the instructions in the prompt, we aim to quantify the effectiveness of various role configurations in enhancing the models' predictive accuracy and structural reliability.

To the best of our knowledge, this is the first study to systematically evaluate the effectiveness of role designs in prompts across multiple datasets and LLMs. Our findings provide valuable insights into optimal prompt design strategies, potentially enhancing the performance of LLMs in various natural language processing tasks. Notably, we observed that designing roles without altering the main prompts contributed to better overall performance. The code provide in anonymous GitHub.[1]

2 Related Works

Significant research has been conducted in prompt engineering and ICL. Methods such as KATE (Knn-Augmented in-conText Example selection) (Liu et al., 2021) and EPR (Efficient Prompt Retrieval) (Rubin et al., 2021) enhance example selection and retrieval for LLMs using k-nearest neigh-

[1] GitHub Code

bors and efficient interaction methods, respectively. Self-generated In-context Learning (SG-ICL) (Kim et al., 2022) reduces reliance on external demonstrations by generating examples internally, while Mutual Information (MI) (Sorensen et al., 2022) and Perplexity Estimation (Gonen et al., 2022) leverage information theory and language familiarity to improve prompt design.

Active Example Selection using reinforcement learning (RL) (Zhang et al., 2022a) involves strategies like Markov Decision Processes (MDP) and Q-learning to optimize example choices. Methods such as LENS (fiLter-thEN-Search) (Li and Qiu, 2023a) refine this process through dataset filtering and iterative search for optimal combinations. Other approaches inspired by topic models select optimal demonstrations from annotated data using smaller LLMs, generalizing these to larger models for ICL (Wang et al., 2024), with frameworks like Unified Demonstration Retriever (UDR) (Li et al., 2023) employing multi-task ranking and iterative mining for high-quality candidates.

Instruction Induction (Honovich et al., 2022) and Automatic Prompt Engineer (APE) (Zhou et al., 2022b) enhance model instruction-following by generating instructions from input-output pairs and optimizing them via candidate searches. The SELF-INSTRUCT framework (Wang et al., 2022b) further improves these capabilities by utilizing self-generated examples. Chain-of-Thought (CoT) prompting (Wei et al., 2022) and its variants like complex CoT (Fu et al., 2022) and Auto-CoT (Zhang et al., 2022b) provide reasoning demonstrations, enhancing models' reasoning abilities for complex tasks.

The Self-Ask method (Press et al., 2022) enhances performance by decomposing complex questions into simpler sub-questions, which the model answers sequentially. The Memory-of-Thought (MoT) (Li and Qiu, 2023b) approach involves pre-thinking on an unlabeled dataset and storing high-confidence thoughts as external memory, which the model can recall during testing to aid reasoning.

Super In-Context Learning (SuperICL) (Xu et al., 2023) enhances supervised task performance by combining LLMs with locally fine-tuned smaller models that act as plug-ins, providing specific task knowledge. Iterative Context-Aware Prompter (iCAP) (Wang et al., 2022a) employs an iterative prompting framework for multi-step inference, while the LEAST-TO-MOST Prompting method (Zhou et al., 2022a) addresses complex problems by breaking them into simpler subproblems. Additionally, the Task-Agnostic Prefix Prompt (TAPP) (Ye et al., 2024) facilitates zero-shot generalization by prepending a fixed prompt to every input.

3 Methodology and Experimental Setup

3.1 Datasets

Following (Min et al., 2022) and (Rouzegar and Makrehchi, 2024a), our experiments were conducted using samples from a diverse range of datasets to evaluate the performance of different prompt designs across various natural language tasks:

- **commonsense_qa**: (Talmor et al., 2019) A dataset designed for common-sense question answering.
- **ai2_arc**: (Sheng and Uthus, 2020) A benchmark dataset used for evaluating question-answering capabilities.
- **wiki_movie_plots**: A dataset containing movie plots used for genre classification.
- **IMDB_reviews**: A sentiment analysis dataset comprising movie reviews with corresponding sentiment labels.

These datasets cover a wide spectrum of tasks, including question answering, sentiment analysis, and text classification, ensuring a comprehensive evaluation of the role designs in prompts.

3.2 Prompt Designs

We utilized four state-of-the-art LLMs for our experiments: GPT-3.5-turbo-0125 and GPT-4o from OpenAI, and Llama2-7b-chat and Llama2-13b-chat from Meta. These models were selected due to their robust performance and widespread applicability in various NLP tasks. (Brown et al., 2020) (Touvron et al., 2023)

To investigate the impact of role design, we experimented with the following prompt configurations. Importantly, the main content of the prompt remained the same across all configurations; the key difference lay in how the prompt was split into distinct roles. This approach allowed us to isolate the effect of role-based structuring on model performance, which is the primary focus and contribution of our study.

Dataset	LLM	ZeroU		ZeroSU		FewU		FewSU		FewSUA	
		Str. Acc.	F1 Score	Str. Acc.	F1 Score	Str. Acc.	F1 Score	Str. Acc.	F1 Score	Str. Acc.	F1 Score
commonsense_qa	GPT-3.5	24	68	60	68	20	69	55	68	100	**73**
	GPT-4o	100	77	100	80	99	79	100	82	100	**83**
	Llama2-7b	0	**19**	0	**19**	0	18	0	**19**	67	9
	Llama2-13b	0	33	0	**36**	0	**36**	0	**36**	99	28
ai2_arc	GPT-3.5	73	76	39	80	60	80	87	78	99	**85**
	GPT-4o	99	96	100	96	100	95	100	96	100	**97**
	Llama2-7b	0	36	0	39	0	26	0	25	9	**40**
	Llama2-13b	0	**50**	0	52	0	39	0	37	77	48
wiki_movie_plots	GPT-3.5	99	76	99	**79**	99	77	99	76	99	77
	GPT-4o	100	80	100	81	100	81	100	82	100	**84**
	Llama2-7b	0	**75**	0	**75**	0	68	0	70	25	74
	Llama2-13b	0	73	0	75	0	72	0	75	64	**85**
IMDB_reviews	GPT-3.5	100	**94**	100	93	100	93	100	67	100	**94**
	GPT-4o	100	95	100	95	100	96	100	92	100	**97**
	Llama2-7b	1	**87**	1	87	0	62	0	60	18	85
	Llama2-13b	3	91	3	90	0	64	0	67	82	**93**

Table 1: Performance results of various language models using different prompt designs on various datasets. The table includes Structural Accuracy (Str. Acc.) and F1 scores for each method: ZeroU (Zero-shot User-only), ZeroSU (Zero-shot System and User), FewU (Few-shot User-only), FewSU (Few-shot System and User), and FewSUA (Few-shot System, User, and Assistant).

In role design, the "system" provides high-level instructions that guide the overall task and set the context for the interaction. This includes specifying the format, the rules for the responses, or any additional context necessary for the task. The "user" presents specific queries or prompts that the model needs to respond to, effectively driving the interaction and simulating real-world usage scenarios. The "assistant" is the model's response to the user's queries, which should adhere to the system's instructions and accurately address the user's prompts. (Figure1)

Zero-shot User Prompt (ZeroU): The entire prompt is provided as a user instruction without any role distinctions.

Zero-shot System and User Prompt (ZeroSU): The initial part of the prompt is provided as a system instruction, followed by the user input.

Few-shot User Prompt (FewU): Multiple examples of questions and answers are provided within the user prompt.

Few-shot System and User Prompt (FewSU): Examples are provided with clear distinctions between system and user roles.

Few-shot System, User, and Assistant Prompt (fewSUA): Includes examples with system instructions, user inputs, and assistant responses.

3.3 Experimental Setup

Data Preparation: Each dataset was divided into training and test sets, ensuring a balanced distribution of categories and labels. We used a balanced subset from each dataset to ensure a comprehensive evaluation. For the few-shot scenario, we randomly selected three samples from each dataset, ensuring these examples were not included in the test set.

Prompt Construction: We maintained the same format of role design for both models, as they are both instruction-tuned LLMs. This consistency allowed us to isolate the effect of role-based structuring on model performance.

Model Configuration: All GPT and Llama models were configured with a temperature of '0.0'. This setting minimizes the randomness in the model's output, ensuring more deterministic responses. Other parameters, such as maximum sequence length and batch size, were optimized for each task to ensure optimal performance.

Evaluation Metrics: The primary evaluation metric was the F1 score, which checks if the output of the prompt matches with the class for sentiment analysis and movie genres or the correct answer in QA tasks. The second metric is structural accuracy, which measures how closely the output matches the requested structure in the prompt (Figure 2). For example, the movie genre should be a single word among the specified genres, the sentiment should be a single word, either "positive or "negative," and QA answers should be a single capital letter among the choices (A, B, C, or D). (Figure 3 and Figure4)

We examined structural accuracy regardless of the correctness of the answer. For instance, if the output was a single letter (A to D) for ai2_arc, it was considered structured. However, outputs like "the answer to this question is D: state park" or simply "D: state park" were considered unstructured because they did not match the desired format. Additionally, for measuring the F1 score, we used post-processing to extract the label from the output and determine if it matched the true label of the dataset.

4 Results

The results of our experiments are summarized in Table 1, where we present the F1 scores and structural accuracy (Str. Acc.) for various prompt designs across four datasets. The experiments reveal several key insights regarding the impact of role design in prompts for LLMs. In most cases, the few-shot system, user, and assistant prompt (FewSUA) configuration exhibited a superior F1 score across the experiments, indicating that incorporating clear role distinctions and examples significantly enhances model performance.

For the Llama models, using few-shot user (FewU) and few-shot system and user (FewSU) prompts led to a noticeable decrease in F1 scores. This suggests that embedding few-shot examples within user prompts may cause these models to produce more incorrect answers and hallucinations.

The structural accuracy of GPT models was generally high, with both models adhering to the desired output structure in most cases. However, the Llama models frequently generated more than one word, failing to follow the instructions, which is understandable given their smaller size. A standout finding is that the FewSUA prompt configuration helped the Llama models generate outputs in the desired structure, making it potentially useful for applications like chatbots where maintaining a specific structure is crucial. Also, Llama2-13b outperformed Llama2-7b in structural adherence.

Additionally, in tasks such as movie genre classification and sentiment analysis, the Llama2-13b model showed results comparable to the GPT models, demonstrating its capability in certain NLP tasks despite its smaller size. However, in question-answering tasks, the performance of the GPT models was significantly better and not comparable to that of the Llama models.

Overall, our experiments indicate that larger models generally achieve better F1 scores. GPT-4o exhibited superior performance compared to GPT-3.5, and Llama2-13b outperformed Llama2-7b. This suggests that model size plays a crucial role in both F1 score and structural adherence. The FewSUA prompt configuration consistently improved both accuracy and structural performance, highlighting its potential as a robust prompt design for enhancing the capabilities of LLMs.

5 Limitations

This study's limitations include the use of a limited range of datasets and models, which may affect generalizability. It's important to note that the role designs are only applicable to instruction-tuned LLMs, such as Llama and GPT chat models, as these roles are not understandable for all models. Additionally, there may be potential improvements in modifying the original prompt structure to achieve better results. Future research should explore a broader range of models and datasets.

Conclusion

This study systematically evaluated the impact of role design in prompts on the performance of large language models (LLMs) in zero-shot and few-shot learning scenarios. Using GPT-3.5, GPT-4o, Llama2-7b, and Llama2-13b-chat from Meta, we demonstrated that incorporating distinct role configurations significantly enhances model performance across various natural language processing tasks. Our experiments showed that the FewSUA prompt configuration, which includes system instructions, user inputs, and assistant responses, consistently improved both predictive accuracy and structural adherence. These findings highlight the potential of role-based prompt structuring to optimize LLM capabilities, providing valuable insights for future prompt design strategies.

References

Harika Abburi, Michael Suesserman, Nirmala Pudota, Balaji Veeramani, Edward Bowen, and Sanmitra Bhattacharya. 2023. Generative ai text classification using ensemble llm approaches. *arXiv preprint arXiv:2309.07755*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. 2022. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*.

Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. 2022. Complexity-based prompting for multi-step reasoning. In *The Eleventh International Conference on Learning Representations*.

Hila Gonen, Srini Iyer, Terra Blevins, Noah A Smith, and Luke Zettlemoyer. 2022. Demystifying prompts in language models via perplexity estimation. *arXiv preprint arXiv:2212.04037*.

Or Honovich, Uri Shaham, Samuel R Bowman, and Omer Levy. 2022. Instruction induction: From few examples to natural language task descriptions. *arXiv preprint arXiv:2205.10782*.

Hyuhng Joon Kim, Hyunsoo Cho, Junyeob Kim, Taeuk Kim, Kang Min Yoo, and Sang-goo Lee. 2022. Self-generated in-context learning: Leveraging autoregressive language models as a demonstration generator. *arXiv preprint arXiv:2206.08082*.

Xiaonan Li, Kai Lv, Hang Yan, Tianyang Lin, Wei Zhu, Yuan Ni, Guotong Xie, Xiaoling Wang, and Xipeng Qiu. 2023. Unified demonstration retriever for in-context learning. *arXiv preprint arXiv:2305.04320*.

Xiaonan Li and Xipeng Qiu. 2023a. Finding support examples for in-context learning. *arXiv preprint arXiv:2302.13539*.

Xiaonan Li and Xipeng Qiu. 2023b. Mot: Memory-of-thought enables chatgpt to self-improve. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6354–6374.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2021. What makes good in-context examples for gpt-3? *arXiv preprint arXiv:2101.06804*.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. 2022. Measuring and narrowing the compositionality gap in language models. *arXiv preprint arXiv:2210.03350*.

Hamidreza Rouzegar and Masoud Makrehchi. 2024a. Enhancing text classification through llm-driven active learning and human annotation. In *Proceedings of The 18th Linguistic Annotation Workshop (LAW-XVIII)*, pages 98–111.

Hamidreza Rouzegar and Masoud Makrehchi. 2024b. Generative AI for Enhancing Active Learning in Education:a Comparative Study of GPT-3.5 and GPT-4 in CraftingCustomized Test Questions. *Proceedings of the Canadian Conference on Artificial Intelligence*.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2021. Learning to retrieve prompts for in-context learning. *arXiv preprint arXiv:2112.08633*.

Emily Sheng and David Uthus. 2020. Investigating societal biases in a poetry composition system. *Preprint*, arXiv:2011.02686.

Taylor Sorensen, Joshua Robinson, Christopher Michael Rytting, Alexander Glenn Shaw, Kyle Jeffrey Rogers, Alexia Pauline Delorey, Mahmoud Khalil, Nancy Fulda, and David Wingate. 2022. An information-theoretic approach to prompt engineering without ground truth labels. *arXiv preprint arXiv:2203.11364*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Boshi Wang, Xiang Deng, and Huan Sun. 2022a. Iteratively prompt pre-trained language models for chain of thought. *arXiv preprint arXiv:2203.08383*.

Xinyi Wang, Wanrong Zhu, Michael Saxon, Mark Steyvers, and William Yang Wang. 2024. Large language models are latent variable models: Explaining and finding good demonstrations for in-context learning. *Advances in Neural Information Processing Systems*, 36.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022b. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Canwen Xu, Yichong Xu, Shuohang Wang, Yang Liu, Chenguang Zhu, and Julian McAuley. 2023. Small models are valuable plug-ins for large language models. *arXiv preprint arXiv:2305.08848*.

Seonghyeon Ye, Hyeonbin Hwang, Sohee Yang, Hyeongu Yun, Yireun Kim, and Minjoon Seo. 2024. Investigating the effectiveness of task-agnostic prefix prompt for instruction following. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19386–19394.

Wenxuan Zhang, Yue Deng, Bing Liu, Sinno Jialin Pan, and Lidong Bing. 2023. Sentiment analysis in the era of large language models: A reality check. *arXiv preprint arXiv:2305.15005*.

Yiming Zhang, Shi Feng, and Chenhao Tan. 2022a. Active example selection for in-context learning. *arXiv preprint arXiv:2211.04486*.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2022b. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, et al. 2022a. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*.

Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2022b. Large language models are human-level prompt engineers. *arXiv preprint arXiv:2211.01910*.

A Appendix

The appendix provides detailed descriptions and discussions of the examples of prompts and the outputs from different models, specifically for the wiki_movie_plots dataset. The prompt designs include ZeroU (Zero-shot User-only), ZeroSU (Zero-shot System and User), FewU (Few-shot User-only), FewSU (Few-shot System and User), and FewSUA (Few-shot System, User, and Assistant). Each design isolates the effect of role-based structuring on model performance.

While the specific example in Figure 1 is for the wiki_movie_plots dataset, similar prompts are used for other datasets with minor changes in the desired output, such as sentiment for IMDB reviews or answers for commonsense_qa and ai2_arc. By splitting the prompts into distinct roles, the study observes the impact on the models' ability to generate accurate and structured responses.

Figures 2, 3, and 4 present examples of structured and unstructured responses across different datasets, including sentiment analysis, genre classification, and question-answering tasks. Structured responses are critical for consistency in evaluation, requiring specific formats such as single words or letters.

The post-processing technique mentioned in the text is essential for accurately extracting these labels from the model outputs, highlighted by the blue-colored text in the examples. However, there are instances where the extraction process makes mistakes, as shown in Figure 3. For example, in the GPT-4o (commonsense_qa) example, the post-processing incorrectly extracted 'A' as the answer, although the output indicated 'C'. This highlights the challenges in achieving precise output structures and the importance of improving post-processing techniques to avoid such errors.

The appendix emphasizes the importance of clear role distinctions in prompt design, particularly the FewSUA configuration, which consistently improves both F1 scores and structural adherence. The detailed examples and discussions provide valuable insights for developing effective prompt designs, highlighting the crucial role of post-processing in maintaining output accuracy.

A.1 Resource Utilization

For the experiments involving GPT models, we used the OpenAI API, with the total cost amounting to approximately 120 USD. For the Llama models, we utilized a single GPU with 64GB memory for the 7b model and two parallel GPUs, each with 64GB memory, for the 13b model. This setup ensured that we had sufficient computational resources to conduct the experiments efficiently and obtain reliable results.

ZeroU prompt:

role: **"user"**, content: Determine the genre of the movie based on the provided plot:
For the plot provided, classify its genre as a single word (without other marks or words like 'genre:'), either "comedy", "action", "drama", or "horror".
Plot[i]

ZeroSU prompt:

role: **"system"**, content: Determine the genre of the movie based on the provided plot.
For the plot provided, classify its genre as a single word (without other marks or words like 'genre:'), either "comedy", "action", "drama", or "horror".
role: **"user"**, content: Plot[i]

FewU prompt:

role: **"user"**, content: Determine the genre of the movie based on the provided plot:
For the plot provided, classify its genre as a single word (without other marks or words like 'genre:'), either "comedy", "action", "drama", or "horror".
Examples:
Plot[1] - Genre[1]
Plot[2] - Genre[2]
Plot[3] - Genre[3]
...
Plot[i]

FewSU prompt:

role: **"system"**, content: Determine the genre of the movie based on the provided plot.
For the plot provided, classify its genre as a single word (without other marks or words like 'genre:'), either "comedy", "action", "drama", or "horror".
role: **"user"**, content: Examples:
Plot[1] - Genre[1]
Plot[2] - Genre[2]
Plot[3] - Genre[3]
...
Plot[i]

FewSUA prompt:

role: **"system"**, content: Determine the genre of the movie based on the provided plot.
For the plot provided, classify its genre as a single word (without other marks or words like 'genre:'), either "comedy", "action", "drama", or "horror".
role: **"user"**, content: Plot[1]
role: **"assistant"**, content: Genre[1]
role: **"user"**, content: Plot[2]
role: **"assistant"**, content: Genre[2]
role: **"user"**, content: Plot[3]
role: **"assistant"**, content: Genre[3]
role: **"user"**, content: Plot[i]

Figure 1: This figure illustrates the various prompt designs utilized to feed all models. Each prompt design is defined as follows: ZeroU (Zero-shot User-only), ZeroSU (Zero-shot System and User), FewU (Few-shot User-only), FewSU (Few-shot System and User), and FewSUA (Few-shot System, User, and Assistant).

Sentiment (IMDB reviews):
positive / negative (single word)

Genre (wiki_movie_plots):
comedy, drama, action, horror (single word)

QA (ai2_arc):
A to E (single letter)

QA (commonsense_qa):
A to D (single letter)

Figure 2: This figure illustrates the various structured responses for different datasets used to evaluate the models.

GPT-4o (IMDB reviews):
neutral

GPT-4o (commonsense_qa):
Greg's performance was mediocre. What might his habits be like? A: distinguished B: unique C: peculiar D: exceptional E: outstanding
The answer is: C

GPT-3.5 (IMDB reviews):
science fiction

GPT-3.5 (commonsense_qa):
D: to remove toxic material from the stomach before it is absorbed

Figure 3: This figure shows examples of unstructured responses from GPT models. The blue-colored text highlights the extracted labels in the examples.

Llama2-7b (commonsense_qa):

Based on the question provided, the answer is:

A: halving the distance between them

Therefore, the answer is "A".

Llama2-7b (IMDB reviews):

The sentiment of the review is negative. The reviewer expresses disappointment with the sequel compared to the original, citing a lack of fun and too much time spent on plot development. They also mention that Tim Thomerson's wisecracking is toned down, and the performances are adequate but the script lets them down. The action is described as routine and the plot is only mildly interesting, which suggests that the reviewer was not impressed with the film's action sequences or story. The reviewer also states that they need lots of silly laughs to stay entertained during a Trancers movie, but the film fails to deliver on this front. Overall, the review is negative and suggests that the reviewer did not enjoy the sequel as much as the original.

Llama2-7b (wiki_movie_plots):

Based on the provided plot, I would classify the movie as a comedy. The plot involves a group of irresponsible and wealthy individuals who die in an accident and become ghosts, with the goal of freeing their friend from a regimented lifestyle in order to enter heaven. The movie appears to have a lighthearted and comedic tone, with elements of slapstick humor, witty dialogue, and farcical situations. The characters' antics and misadventures are likely to provide most of the comedic elements in the movie.

Llama2-13b (wiki_movie_plots):

Sure! Based on the plot provided, the genre of the movie would be:

Comedy.

The plot involves a man trying to relive his youth and pursue a beautiful woman, with his friends, in a series of comedic and awkward situations. The tone of the plot is lighthearted and humorous, indicating that the movie would be a comedy.

Llama2-13b (IMDB reviews):

Based on the review, I would rate the movie 3 out of 10. Here's a breakdown of the review:

* The reviewer can't knock the movie too hard because it's clear that the filmmakers were trying to make it bad or campy.

* Some of the parts that were intended to be funny actually are, but only for their cheesiness.

* The reviewer notes that the movie goes over the top near the end with a time warp and sexual innuendoes.

* The reviewer is surprised that the movie got a PG rating, implying that it's not suitable for all audiences.

* The reviewer concludes that it's a fun way to waste a couple of hours, but it's not on the same level as Star Wars or Trek.

Llama2-13b (ai2_arc):

Sure! Based on the question provided, the answer is:

D : wildfire

Figure 4: This figure shows examples of unstructured responses from Llama2 models. The blue-colored text highlights the extracted labels in the examples.