Using Large Language Models for Text Classification in Experimental Economics*

Can Celebi
Stefan P. Penczynski

Using Large Language Models for Text Classification in Experimental Economics*

Can Çelebi†

Stefan P. Penczynski‡

June 10, 2024

In our study, we compare the classification capabilities of GPT-3.5 and GPT-4 with human annotators using text data from economic experiments. We analysed four text corpora, focusing on two domains: promises and strategic reasoning. Starting with prompts close to those given to human annotators, we subsequently explored alternative prompts to investigate the effect of varying classification instructions and degrees of background information on the models' classification performance. Additionally, we varied the number of examples in a prompt (few-shot vs zero-shot) and the use of the zero-shot "Chain of Thought" prompting technique. Our findings show that GPT-4's performance is comparable to human annotators, achieving accuracy levels near or over 90% in three tasks, and in the most challenging task of classifying strategic thinking in asymmetric coordination games, it reaches an accuracy level above 70%.

Keywords: Text Classification, GPT, Strategic Thinking, Promises.

JEL Classification: .

---

†Department of Economics, University of Mannheim, Germany, cnelebi@gmail.com.

‡School of Economics and Centre for Behavioural and Experimental Social Science (CBESS), University of East Anglia, Norwich Research Park, Norwich NR4 7TJ, United Kingdom, s.penczynski@uea.ac.uk, Tel. +44 1603 59 1796.

1. Introduction

Advances in information technology increase the performance of computerised Natural Language Processing (NLP), which in turn increases NLP's potential contribution to the scientific endeavour. The accessibility of Large Language Models (LLM) will further increase the use of "text as data" in the social sciences (Gentzkow et al., 2019) because, as we show here, text can easily and accurately be classified according to challenging social scientific concepts.

In this study, we employed GPT models to classify text transcripts from economic experiments focused on promise and the level of strategic thinking. We aim to answer the following research questions:

- How do GPT models compare to expert human annotators and traditional machine learning methods in classifying these concepts?

- Can classification instructions designed for human annotators be minimally modified to serve as prompts for LLMs, delivering comparable performance levels?

- How effective are established prompting techniques in classifying these concepts, and how does model size influence task performance and the efficacy of these techniques?

We vary the prompts along two dimensions in order to explore LLMs' performance and to understand which prompting techniques are most effective for the given classification tasks. In the first dimension, we alternate between zero-shot and few-shot prompting, which relates to the number of example messages and classifications that are included in the prompts (Dong et al., 2022). In the second dimension, we alternate between requiring responses to be with or without the chain-of-thought (CoT) feature by employing the 0-shot CoT prompting technique introduced by Kojima et al. (2022). With CoT, the LLM provides written justifications for each of its classification decisions. Furthermore, we conducted all these classifications using OpenAI's GPT-3.5 turbo as well as GPT-4 turbo to investigate performance gains achievable with larger LLMs.

We find that GPT-4 outperforms GPT-3.5 and achieves high levels of agreement, reaching near or above 90% and up to 73% in more complex tasks. Importantly, these high levels can at times be attained through zero-shot classification instructions originally designed for human annotators. Unlike traditional supervised machine learning approaches, which necessitate partitioning data for training and testing (Penczynski,

2019), use of LLMs obviates the need for data separation, making it particularly advantageous in low-resource environments, such as those commonly found in economic experiments. Moreover, we document that while n-shot prompting consistently improves the performance of both models, 0-shot CoT prompting's effect on models' performances is dependent on the model, the prompt and the task difficulty. It often enhanced GPT-4's performance but did not reliably improve GPT-3.5's performance across tasks and prompts, generated more consistent performance gains for relatively more difficult tasks, and when used in conjunction with n-shot prompting, always resulted in the best performing outcomes for GPT-4.

Additionally, by examining the classification of two concepts –promises and strategic thinking– which potentially vary in terms of their availability in the pre-training corpus of the models, we investigate whether the classification task primarily involves recognition-based or learning-based subtasks, depending on the concept (Pan et al., 2023). While we find no significant difference in performance gains switching from GPT-3.5 to GPT-4 among concepts, we do observe a higher degree of performance gain from incorporating demonstrations into the prompts for the classification of level of thinking compared to the classification of promises, and thereby, find some evidence that the classification of level of thinking involves more learning components than the classification of promises.

As a scientific tool, LLMs offer a number of advantages over RAs, making them an increasingly attractive option for researchers working with 'text as data' and enhancing the scientific insights derived from text. The costs associated with RAs arise from recruitment, instruction, and, most importantly, their work-time. In contrast, LLM services are paid by tokens and are comparatively inexpensive. For example, classifying 100 messages in the most complex dataset cost 6.60 USD (as of February 2024) and took only 43 minutes. LLMs offer attractive and distinctive performance features that are poised to improve further. They deliver immediate results, provide on-demand detailed justifications (CoT), and ensure consistent classifications that are not subject to fatigue. Additionally, testing and gradually refining prompts is straightforward and cost-effective with LLMs, whereas it is prohibitively expensive with RAs.

2. LLMs and related literatures

An LLM is a statistical language model trained on a large corpus to predict the next word for any given textual input. By inputting text instructions, one can strategically leverage this predictive capability to steer the model's output towards a desired out-

come, a practice commonly referred to as prompting. The appeal of prompting stems from the ease with which natural language allows us to convey complex ideas. Yet, this very flexibility may introduce inaccuracies or ambiguities if concepts are not clearly defined or presented with insufficient context. The effectiveness of a prompt hinges both on the user's adeptness at crafting instructions with clarity and contextual relevance, and on the model's ability to accurately *interpret* these instructions within their context. While an LLM's capacity to process text and follow instructions are fundamentally based upon its pre and post training and its parameter size, for downstream tasks, the user can still attempt to refine her mode of interacting with the LLM by engineering her prompts to align more closely with model's operational framework, in order to effectively leverage its capabilities(Reynolds and McDonell, 2021).

2.1. GPTs in the computer science literature

FLAN (Wei et al., 2021), OPT (Zhang et al., 2022b), and PaLM (Chowdhery et al., 2023) are examples of LLMs that have showcased remarkable proficiency in natural language understanding (NLU) tasks (Ye et al., n.d.). Particularly, the Generative Pretrained Transformer (GPT) series (Brown et al., 2020), more specifically GPT-3 and its subsequent iterations GPT-3.5, GPT-3.5-turbo, GPT-4 and GPT-4-turbo introduced by OpenAI have sparked considerable attention due to their exceptional performance in integrating various NLU tasks into generative ones (Ye et al., n.d.).

Earlier GPT models, GPT-1 and GPT-2, are limited in their ability to recognise textual patterns across diverse tasks due to their relatively smaller training corpus and parameter size (Radford et al., 2019). Consequently, these models require substantial fine-tuning on task-specific datasets to achieve satisfactory performance. Yet finetuning poses several problems: first, it requires large volumes of task-specific data; second, there is a risk that these training datasets do not cover the full spectrum of task variations, which could lead to suboptimal performance on data not represented within the training set (lack of generalizability due to over-fitting) (Brown et al., 2020). Furthermore, fine-tuning an LLM on data that introduces new knowledge is documented to increase the model's likelihood to make up information (hallucinate) (Gekhman et al., 2024).

Building on its predecessor, GPT-3 has been trained on a significantly larger corpus, consisting of approximately 400 billion tokens, compared to GPT-2 which was trained on 1.5 billion tokens (Brown et al., 2020). This extensive training has markedly enhanced its ability to detect diverse textual patterns (Brown et al., 2020) and has enabled reasoning-like emergent qualities (Wei et al., 2022b). Notably, GPT-3 can perform

specialised tasks when provided with few examples demonstrating how to perform it (Brown et al., 2020). This capability, which significantly reduced the need for parameter adjustments through fine-tuning, catalysed the development of the In-Context Learning (ICL) paradigm for LLMs comparable in size to GPT-3 or larger (Dong et al., 2022).

In the ICL paradigm, the process of demonstrating task execution through a small number of input-output pairs, where the input serves as the question and the output as the answer, is referred to as n-shot prompting. In the specific case of classification tasks, the term "n" indicates the number of examples provided in the prompt. n-shot prompting quickly gained popularity as it requires only a few demonstrations to guide the model toward achieving performance comparable to that of fine-tuned models trained on extensive datasets. Notably, a single demonstration can be as effective as fine-tuning the model with approximately 300 to 3,500 input-output pairs, depending on the task (Scao and Rush, 2021). Additionally, by modifying the format of the demonstrations from <input, output> to <input, reasoning, output>, one can enable the model to demonstrate reasoning capabilities. This approach, referred to as n-shot-CoT (Chain of Thought), has proven to significantly enhance the models' performance, especially on more involved tasks that can potentially benefit from multi-step reasoning (Huang and Chang, 2022).

Although the ICL paradigm offers a flexible and data-efficient way to "teach" the model at inference, the efficacy of the method on improving the model's task performance, or in other words the model's ability to "learn" from these demonstrations, relies to a greater extent on the choice of examples, the sequence in which they are presented within the prompt, and the frequency with which examples for each category to be classified are provided (Lu et al., 2021; Kumar and Talukdar, 2021a; Zhao et al., 2023). Although various methods to select the optimal set of examples and their order have been proposed, and documented to improve the model's performance (Li and Qiu, 2023a; Su et al., 2022; Liu et al., 2021; Luo et al., 2023; Chang and Jia, 2022), their technically demanding procedures may be less accessible for social scientist to implement. This lack of accessibility juxtaposes with the appeal of GPT's out-of-the-box usability, which we believe is necessary for any prompting method to be widely adopted by social scientists.

Given n-shot-CoT prompting is an extension to n-shot prompting, its effectiveness in enhancing the model's task performance also relies on the choice and order of examples. In addition, however, its effectiveness is also dependent on the manner with which the rationales are provided for each demonstration (Huang and Chang, 2022).

To improve the reliability of n-shot prompting, the most cited method is the "self-consistency" method proposed by Wang et al. (2022). In this method, the temperature hyperparameter is set to a strictly positive value (e.g. 0.5 or 0.7) in order to increase the variability in the reasoning sequence of the output, and multiple request (e.g. 20 or 40) to the model are made using the n-shot-CoT method for every single task instance. Then the most frequently outputted answer is picked as the decisive (consistent) output. Noting that requesting a reasoning prior the answer already puts additional token cost, by generating k many outputs with reasoning for each task instance, "self-consistency" method is increasing the computational cost approximately k times more. Making the classification task approximately k times more costly to improve the reliability of the model's output could make this approach prohibitively expensive for social scientists, and potentially deter them from viewing it as a viable alternative to human annotators.

To refine the reasoning provided with each examples, it has been suggested, among many, to use multiple human annotators to provide a diverse set of reasoning for each example (Li and Qiu, 2023b), to use the LLM itself to generate a diverse sets of reasoning for each examples and then select the reasoning with the most steps (Fu et al., 2022), or to generate multiple outputs using the generic n-shot-CoT prompting method and to choose the most frequently provided output as the final output of the model (Wang et al., 2022) (see Huang and Chang (2022) for details and other methods to refine the rationale). Arguably, although these proposed methods for refining the reasoning in demonstrations are not computationally demanding, they still require careful selection of the most suitable reasoning for each example, and may lead the researchers to doubt whether the explanations considered were adequate, particularly when the model's performance does not meet their expectations.

In addition to the unreliability concerns with n-shot and n-shot-CoT prompting, it is also unclear how the model "learns" at inference to perform a task via few demonstrations of the task, and whether it genuinely learns via demonstrations (Reynolds and McDonell, 2021). Min et al. (2022) demonstrate that even when the output labels of the input-output pairs in n-shot prompts are replaced with incorrect labels, the model's performance remains unaffected. They suggest that models do not learn from demonstrations in the same way humans do from examples; rather, these examples primarily serve to delineate the label space and the distribution of the input text, thereby aiding the model in task execution[1]. Reynolds and McDonell (2021) argue that demonstrations do not actually "teach" the model how to perform the task but simply enables the

---

[1]Yoo et al. (2022) revisited the assertions of Min et al. (2022) and found instances where employing incorrect output labels adversely affects model performance. Consequently, the question of whether and how models learn from demonstrations, and precisely what they learn, remains an open question.

model to locate the tasks in the model's existing knowledge of tasks that it acquired during its pre-training[2] (meta-knowledge). Similarly, with n-shot-CoT prompting, it is unclear whether the model genuinely engages in reasoning and, if so, how this reasoning improves the task performance (Madaan and Yazdanbakhsh, 2022). Furthermore, the ability of a model to reason effectively has been shown to correlate with the frequency of a task's presence in its pre-training corpus: the more frequently a task is represented in the training data, the more likely the model is to exhibit sound reasoning and produce accurate outputs (Razeghi et al., 2022).

Given the aforementioned challenges with n-shot-CoT method, a notable, demonstration-free alternative is the 0-shot-CoT prompting method (Kojima et al., 2022). This approach simply involves appending the phrase "Let's think step-by-step" to the instructions and thereby triggers a reasoning process before generating the output. 0-shot-CoT, devoid of the reliability concerns regarding the selection of examples or the quality of reasoning, is task-agnostic and can seamlessly be integrated into an existing prompt by adding the keyword "think step-by-step" into the specific sections of the instructions where the user wishes to invoke reasoning steps in the model (OpenAI, 2023b). While 0-shot-CoT has been shown to significantly enhance performance across a range of tasks, its efficacy diminishes with relatively more involved tasks that potentially requires an explicit outline of the reasoning steps to be followed by the model. In such scenarios, the method often yields suboptimal outcomes due to the model's failure to accurately execute or complete the necessary reasoning steps, either by omitting steps or by making errors within specific steps of the reasoning process (Zheng et al., 2023; Wang et al., 2023). Consequently, alternative 0-shot reasoning-invoking methods have been proposed to address these limitations (Huang and Chang, 2022). Furthermore, 0-shot-CoT's effectiveness is similarly influenced by the model's pre-training corpus. Tasks less represented within the training corpus is observed to provide diminished performance improvements when the method is integrated. This highlights, once again, the dependency of the model's performance on the attributes of its training corpus (Wu et al., 2023).

OpenAI has progressively enhanced GPT-3 by fine-tuning on a collection of instruction-answer pairs[3] (Ouyang et al., 2022). These improvements enabled the model to more

---

[2]As a consequence, In-Context learning is also referred to as priming (Webson and Pavlick, 2021). Yet, the term priming encompasses a broader spectrum of prompting techniques. For instance, priming can also be done by pre-pending a prompt with a series of token (instead of or in addition to the input-output demonstration pairs) that do not necessarily make intuitive sense (Kumar and Talukdar, 2021b).

[3]Fine-tuning GPT-3 to better follow instructions resulted in the development of GPT-3.5 (Ouyang et al., 2022). GPT-3.5 was then further fine-tuned using reinforcement learning from human feedback

closely follow instructions, and reduced to a considerable degree the need to instruct the model to perform a task via demonstrations (Chung et al., 2022). Consequently, recent literature has begun to emphasise the instruction learning paradigm which shifts focus from learning through demonstrations to learning via instructions (Lou and Yin, 2024). It's worth noting that this paradigm does not preclude the inclusion of demonstrations within the instructions; rather, it puts more weight on structuring and designing prompts that combine instructions and examples to optimise the model's performance (Lou et al., 2023). There are only a few studies that aim to provide cross-task generalizable prompt design tips intended to enhance demonstration-free instructions to be considered in prompt engineering (Mishra et al., 2021a; Reynolds and McDonell, 2021; Mishra et al., 2021b; Gu et al., 2022; White et al., 2023; Peskine et al., 2023). Additionally, as of writing of this paper, only a single paper has systematically explored various instructional prompting techniques across a wide range of tasks and documented each instructional design component's contribution on improving the model's performance (for a more detailed discussion and the application of these design choices, see Section 4.1) (Mishra et al., 2021a).

The concept of instructing an LLM in a manner akin to how one might instruct a human is appealing because it renders the act of prompting both intuitive and flexible. Furthermore, prompting via detailed instructions, free of examples, circumvents various methodological issues inherent with the ICL paradigm. Yet, the availability and diversity of instructions in the training corpus of the model is also observed to be a determinant factor on the effectiveness of 0-shot instructions on improving the model's performance. When instructions tailored for human annotators (such as mTurkers on Amazon Mechanical Turk) are considered verbatim as prompts to assess the model's ability on following human-tailored instructions (turking test), the smaller GPT-2 model has demonstrated poor performance(Efrat and Levy, 2020), while the larger GPT-3 model, when fine-tuned on a large set of human-tailored instructions, has demonstrated the ability to effectively follow unseen human-tailored instructions (Mishra et al., 2021a; Ouyang et al., 2022). Moreover, human-tailored instructions are documented to outperform basic prompts that instruct the model with one or two sentences devoid of any additional descriptive context for categories (Mishra et al., 2021a). In brief, irrespective of how well instructions are constructed to improve the model's performance, the size of the model and whether the model was fined-tuned on instructions are observed to play a major role in the model's performance; and if the

---

(RLHF) method to further enhance its capacity to understand instructions and to better engage in conversational interactions with its users (Ye et al., n.d.). These advancements led to the creation of GPT-3.5-Turbo, the underlying model of the ChatGPT application.

model is fine-tuned on instructions, then using instruction provides an improvement over the performance of the model.

Similar to the ICL paradigm, where it is uncertain whether the model genuinely learns from demonstrations, it is also unclear whether the model truly grasps the task's context and execution conditions from a set of 0-shot instructions (Webson and Pavlick, 2021). If the model truly learns from instructions, variations between two different sets of instructions that convey the same meaning should not affect its performance. However, it has been observed that, without sufficient fine-tuning on task-specific examples, changes in word choices that preserve semantic textual similarity in 0-shot instruction prompts can impact model performance as significantly as training it with an additional 200 task-specific examples (Puri et al., 2022). On the other hand, when the model is fine-tuned with a large collection of task-specific examples, its performance demonstrates robustness to variations in the wording of the instructions (Puri et al., 2022). These results, on the one hand, demonstrates the importance of providing carefully designed instructions, while, on the other hand, hints at the fact that the model does not only learn from a set of instructions but also leverages the provided descriptions to locate the task on its existing knowledge. Lastly, demonstrations and carefully provided descriptions for each classification category are observed to complement each other. Irrespective of whether the model is sufficiently trained on task-specific instructions, when instructions are supplemented with a few demonstrations, the model's performance is observed to remain stable despite variations in the choice of words and phrases (Gu et al., 2022).

A recurring theme in our discussion of various prompting techniques is that the effectiveness of any such technique in enhancing a model's performance largely depends on how well the task's contextual components are represented in the model's training corpus. Certain elements of any given task might already be familiar to the model, while others may be novel. This distinction categories any given task as either a recognition task, where the model identifies elements it has seen before, or as a learning task, where the model encounters new contextual elements. Through a series of carefully designed experiments, Pan et al. (2023) demonstrate that the marginal effect of additional task examples in a prompt diminishes for recognition tasks, since only a few examples are observed to be sufficient for the model to recognize the task, and any additional examples do not enable the model to "further recognize" it. Conversely, for learning tasks, the effect of additional demonstrations is observed to be somewhat linear, with each additional example helping the model grasp more of the task's contextual nuances a bit more. Another distinction between recognition and learning tasks

identified by Pan et al. (2023) is the scale of the model[4]. The model size is observed not to significantly enhance its performance on recognition tasks, whereas it is observed to be a crucial factor for the model's ability to learn from demonstrations when faced with a novel task (Pan et al., 2023). This observation is supported by Wei et al. (2022b), who argue that as the model scales, it acquires an emergent capability to learn from the examples. Lastly, it is important to note that any given task may consist of subtasks that fall into two either recognition or learning task category. Furthermore, certain contextual elements of a task might be categorised as learning tasks, while others are more appropriately considered as recognition tasks. Thus, both the size of the pre-training corpus and the scale of the model are potentially crucial factors that impact the performance of the model for a given task. Yet this impact may vary depending on the proportion of recognition to learning components within the task.

Consequently, recognising that the impact of prompting techniques on model performance can vary significantly with each specific task is crucial, as this variability necessitates a case-by-case investigation of the effectiveness of different prompting techniques and model configurations. Therefore, it is unreasonable to universally generalise that one prompting technique or a model with a larger training set or more parameters will consistently perform better across all tasks.

2.2. GPTs in the social science literature

In Table 1, we compile a selection of studies that explore the annotation capabilities of various GPT models. Although this list is not exhaustive, it effectively showcases the diverse prompting techniques used by researchers in a range of annotation tasks within the social sciences.

Despite well-established guidelines from computer science literature, there is a noticeable oversight in the related social science literature with respect to the integration of various prompting strategies such as instruction learning, n-shot prompting, 0-shot-CoT, and the proper usage of the temperature hyperparameter. Our aim is to provide insights relevant to our current work and highlight methodological oversights such as the misuse of the temperature hyperparameter (Reiss, 2023; Törnberg, 2023; Gilardi et al., 2023; Pangakis et al., 2023; Matter et al., 2024; He et al., 2024; Li et al., 2024), misuse of 0-shot-CoT (Zhu et al., 2023; Kuzman et al., 2023; Li et al., 2024), classifying messages in batches (Zhang et al., 2022a; Amin et al., 2023; He

---

[4]The scale or size of a model refers to the number of parameters in its neural network. Largest GPT-2 model has approximately 1.5 billion parameters (Radford et al., 2019) while GPT-3 has approximately 175 billion parameters (Brown et al., 2020), and although not exactly known, GPT-4 is estimated to have over 1 trillion parameters (Baktash and Dawodi, 2023).

Table 1: Papers in Social Sciences

Paper	Field	GPT	Temp.	Prompt	Shot	CoT
Rytting et al. (2023)	Poli. Sci.	3	?	Structured*	2, 3	
Chae et al. (2023)	Poli. Sci.	3	0	Basic*	0, 1, 2	
Reiss (2023)	Poli. Sci.	3.5	0.25, 1	Basic$^*_+$	0	
Gilardi et al. (2023)	Poli. Sci.	3.5*	0.2, 1	Original	0	
Zhu et al. (2023)	Psychology Poli. Sci.	3.5*	?	Basic	0	
Li et al. (2024)	Poli. Sci.	3.5*	0, 1	Original Basic*	0	∼
Zhang et al. (2022a)	Poli. Sci.	$\widetilde{3.5^*}$	∼ 0.7	Basic	0	
Aiyappa et al. (2023)	Poli. Sci.	$\widetilde{3.5^*}$	∼ 0.7	Basic	0	
Kuzman et al. (2023)	Linguistics	$\widetilde{3.5^*}$	∼ 0.7	Basic*	0	∼
Zhong et al. (2023)	Linguistics	$\widetilde{3.5^*}$	∼ 0.7	Basic*	0, 1, 5	✓
Amin et al. (2023)	Psychology	$\widetilde{3.5^*}$	∼ 0.7	Basic	0	
Bhat et al. (2023)	Psychology Linguistics Poli. Sci.	$\widetilde{3.5^*}$	∼ 0.7	Basic	0	∼
Heseltine et al. (2024)	Poli. Sci.	$\widetilde{4^*}$	∼ 0.7	Basic*	0	
Törnberg (2023)	Poli. Sci.	4	0.2, 1	Basic	0	
Pangakis et al. (2023)	Psychology Linguistics Poli. Sci.	4	0.6	Original*	0	
Savelka et al. (2023)	Law	4	0	Original & Structured*	0	✓
He et al. (2024)	Linguistics	4	0.2, 1	Original	0	
Rathje et al. (2023)	Psychology	3.5*, 4*	0	Basic*	0, 1	
Matter et al. (2024)	Sociology	3.5*, 4*	0.1	Basic*	0	∼
Ziems et al. (2024)	Psychology Linguistics Poli. Sci.	3.5*, 4	0	Basic$_+$	0, 1	
Our Paper	Economics	3.5*, 4*	0	Basic$_+$, Original & Structured*	0 − 19	✓

Notes: The "Field" column represents the broad field under which the annotation tasks can be categorized. In the "Model" column, the asterisk indicates that the turbo version of the model is used, and the tilde indicates that the model is not leveraged via the API but through the ChatGPT platform. In the "Prompt" column, "Basic" indicates a basic instructions to classify a text, "Basic$_+$" indicates a basic instruction accompanied by a short definition of for each category, "Original" indicates that the original human instructions are used verbatim as the prompt, and "Structured' indicates that a prompt template is used to structure the prompt into distinct components such as instructions, context, definitions, examples and so on. Moreover, in the "Prompt" column, asterisk superscript indicates that the study investigated either to improve the model's performance via restructuring or augmenting the prompt through rephrasing, incorporating additional context or definitions, making the instructions more precise, etc. or to investigate the effect of a specific variation on the prompt such as considering the prompt in an other language, instructing the model to output a non-binary classification, etc. The "Shot" column indicates the number of demonstrations used in the prompt (n-shot prompting). The "Temp." column indicates the temperature parameter(s) used for the respective model(s), question mark indicates that this value is not provided in the respective paper. Moreover, the exact temperature value for ChatGPT is not known and 0.7 the unconfirmed yet commonly assumed value for it. The "CoT" column not only indicates whether the study used some form of chain-of-thought prompting technique (✓) but also points out studies that considered asking for an explanation after the classification is done (∼) either as an attempt to improve the performance or to further investigate the outputs provided.

et al., 2024; Heseltine and Clemm von Hohenberg, 2024; Matter et al., 2024), and using chatGPT rather than the underlying GPT model (Zhang et al., 2022a; Kuzman et al., 2023; Zhong et al., 2023; Amin et al., 2023; Bhat and Varma, 2023; Heseltine and Clemm von Hohenberg, 2024). If unaddressed, these oversights could compromise the perceived utility of LLMs in text annotation tasks and could misdirect the literature towards suboptimal prompting practices. Moreover, certain papers are often cited for their claims that GPT models are unreliable in text annotation tasks, yet their conclusions rest on methodologically questionable practices (Reiss, 2023; Savelka et al., 2023). Meanwhile, other papers advocate for the adoption of specific prompting methodologies, but these recommendations either lack clarity (Pangakis et al., 2023), or show inconsistencies between proposal and practice (Ziems et al., 2024). Consequently, it is crucial to examine these studies more closely to ensure that they do not mislead future research or get perpetuated uncritically in subsequent works.

In Table 1, under the "Prompt" column, the "Basic" tag is used for studies that employ a very basic prompt format such as "Classify X as $Y_1, Y_2, Y_3, ...$". We labeled slightly more involved basic prompts as "Basic$_+$". These prompts either provide additional context for the task, "Given context C, classify ...", invoke a specific persona from the model "Act as R and classify ...", or offer explanations for each classification category, "Classify X as $Y_1, Y_2, ...$ where $Y_1 :< description >, Y_2 :< description >, ...$ ". The "Structured" tag is assigned to prompts that imposes a structured template that organises explanations, context, constraints, demonstrations and additional prompting techniques into modular components via textual cues such as titles or delimiters. The "Original" tag is assigned to prompts that verbatim use instructions tailored for human annotators. Consequently, using this tag in conjunction with the "Structured" tag indicates that the original instructions have been reframed and restructured for its use as a prompt, and a markup language is leveraged to impose this structure. All the papers with a "Structured" tag in Table 1 used the `Markdown` language to structure their prompts. Lastly, the superscript "*" is used to denote studies that explored variations on their initial prompts to enhance model performance either by rephrasing, adding further information, simplifying existing descriptions or using established prompt engineering techniques such as CoT.

A major determinant of a model's task performance is the nature of the task itself, which can be considered under two main dimensions: the representational depth of task-specific categories in the model's pre-training corpus and the complexity of the task. Representational depth reflects the frequency and variety with which the categories to be annotated are represented in the training corpus. A greater representa-

tional depth ensures that the model is exposed to a wider range of conceptual diversity for given a category which, in turn, impacts the model's recognition and learning capabilities from provided demonstrations or descriptions for a given annotation task (Reynolds and McDonell, 2021; Razeghi et al., 2022; Pan et al., 2023). Zhu et al. (2023) document that GPT-3.5 performs relatively poorly when tasked with classifying topics that occurred after its training, such as the Ukraine-Russia war. In a similar vein, for GPT-4, Ziems et al. (2024) report strong performance in tasks involving categories common in everyday conversations, such as "anger" in an emotion recognition task, while in tasks requiring expert knowledge and involving non-conventional categories, such as "white grievance" in hate speech classification, GPT-4's performance is notably weaker.

From an information perspective, a task that requires a more diverse set of information is considered more complex (Liu and Li, 2012). Complexity can also be defined by the level of abstraction necessary and the extent of inferential reasoning needed to effectively interpret and act on information (Yang et al., 2016). For example, while utterance-level classification involves analysing individual statements, conversational-level classification should be considered as more complex as it requires understanding the broader context and dynamics within entire dialogues (Arad et al., 2024). Similarly, analysing court opinions to interpret legal concepts (Savelka et al., 2023) or classifying nth level strategic thinking in jury voting (Çelebi and Penczynski, 2023) involves far more complex cognitive processes than identifying promises (Charness and Dufwenberg, 2006). This increased complexity necessitates a model that not only understands expert specific information but also integrates and reasons about it in a manner that emulates higher-order cognitive processes (Huang and Chang, 2022). For instance, Bhat and Varma (2023) investigate the annotation performance of GPT-3.5 across three tasks of varying complexity and find that the model performs poorly with linguistically more challenging task of news category classification (51% average accuracy) compared to sentiment analysis (84% accuracy). Similarly, Savelka et al. (2023) observe poor GPT-4 performance in the task of analysing court opinions to interpret legal concepts (46% average accuracy) yet this performance is found to be still on par with expert level annotators. Lastly, Ziems et al. (2024) document that as the complexity of the text increases, moving from standalone messages to conversational texts, the performance of both GPT-3.5 and 4 models deteriorate in classification tasks. In sum, the results corroborate that the performance of GPT-3.5 and GPT-4 models are shaped by both the depth of representational coverage in their training data and the complexity of the tasks they are assigned.

An additional aspect of representational depth is the linguistic diversity within the model's pre-training corpus. While specific details are often undisclosed, it is widely inferred that the primary training data for GPT-3.5 and GPT-4 consist predominantly of English texts (Lai et al., 2023). The limited representation of multilingual data can in turn cause models' performance on annotation tasks to deteriorate primarily on under-represented language families with a syntactic order in the form of Subject-Object-Verb (SOV, e.g., Hindi, Turkish, Arabic or Amharic) compared to the family of languages with syntactic order in the form of Subject-Verb-Object (SVO, e.g., German, Italian, Spanish or Slovenian) that English is a part of (Bjerva et al., 2019). This is because variations in syntactic order alter the word inter-dependencies crucial for models' language comprehension (Bender, 2011; Nivre et al., 2016). These structural differences result in unique word co-occurrence patterns and grammatical dependencies that LLMs, primarily trained on English, rely on to infer the semantics of the text. Consequently, GPT models may face greater difficulties in effectively processing the semantics in language families with syntactic structures distinct from English, while languages with a similar syntactic order, such as German, may present fewer challenges (Conneau and Lample, 2019). In practice, Heseltine and Clemm von Hohenberg (2024) find that GPT-4 demonstrates consistent performance across tasks in German, Italian, and Chilean Spanish compared to English. Conversely, Bhat and Varma (2023) observe that GPT-3.5 struggles with Indic languages, although it was not tested against English texts to ascertain if the prompts would have fared better in English. Rathje et al. (2023) report that GPT-3.5-turbo and GPT-4-turbo show comparable performance in Turkish, various African languages[5], and Arabic. Kuzman et al. (2023) document that while Slovenian texts are classified as effectively as English texts by these models, prompts in Slovenian yield poorer results compared to their English counterparts. The empirical findings largely corroborate the theoretical expectations that various GPT models perform better with languages syntactically similar to English, as observed with German, Italian, and Spanish. However, the comparable performance in SOV languages, like Turkish or Arabic, suggests that the models have the capability to effectively adapt to diverse linguistic structures.

The 0-shot improvement technique of providing context and label descriptions on a "Basic" prompt is shown to improve the classification performance of the models (Peskine et al., 2023). Chae and Davidson (2023) initially consider a simpler prompt

---

[5]African languages display diverse syntactic structures across several language families. The Niger-Congo family mainly uses SVO order, similar to English, but variations like SOV are found in some Bantu languages. Nilo-Saharan languages typically feature SOV order, with dialectic variations (Vossen and Dimmendaal, 2020).

than the "Basic" prompt to instruct GPT-3 to classify political stance of Twitter messages with the prompt "Stance:". This method led the model to incorrectly output labels associated with sentiment classification ("positive", "negative") rather than the intended stance labels ("support", "oppose"). Consequently, they opted for the "Basic" prompt to ensure the correct classification labels, and then tested two incremental improvements on the "Basic" prompt: first, by adding a statement indicating the potential expression of a stance, and second, by providing a general definition of what "stance" entails. These modifications improved the prompt's performance by 11%. Similarly, Heseltine and Clemm von Hohenberg (2024) started with a "Basic" prompt and enhanced it by adding a single-sentence description for each category, which on average boosted the model's performance by 8% across different tasks.

The 0-shot improvement technique of reframing instructions (Mishra et al., 2021b), either by making existing instructions more precise or by adding descriptions to clarify category definitions, has also been documented to enhance model performance. Savelka et al. (2023) initially employed an "Original & Structured" prompt. They observed that definitions provided for the categories were too broad and adversely affecting the classification performance. By reframing these definitions to be more precise, they managed to increase the model's performance by an average of 28% across multiple prompts. Matter et al. (2024) first consider a "Basic" prompt. Their prompt has three main labels: explicit violence, implicit violence and non-violence. For violence cases they have additional three subcategories: directed to a specific person, directed to self or undirected. They first test this on a randomly selected subset of the dataset. Then, to enhance model's performance, they provide a sentence description for each label. They, then, further refined the prompt through a process of iteratively testing the refined prompt on additional randomly selected subsets. This process involved identifying misclassifications, instructing GPT to generate additional instructional text to address these errors, and then integrating these GPT suggestions into the prompt. Through these iterations, they developed a final version of the prompt that provides additional descriptions for explicit and implicit violence, and directed and undirected violence categories and an example for explicit, direct category and another example for implicit undirected violence each category. They claim the final prompt significantly improves the model's performance. However, since neither the initial "Basic" prompt nor the prompts from each iteration were tested on the full dataset, they were not able to provide a specific percentage number for the significance of each modification and for the overall effectiveness of the process compared to the initial "Basic" prompt. Furthermore, it is unclear why only specific categories are provided with ad-

ditional instructions or examples. A similar prompt refinement procedure is proposed by Pangakis et al. (2023) where a randomly selected subset of data was first classified by human annotators, and then GPT was prompted to classify the same subset using identical instructions. If the model's classifications substantially diverge from the human annotations (no specific match percentage is given to define "substantial"), they identify where the model misclassifies and adjust the prompts accordingly to improve the model's performance. This process was repeated until a "satisfactory" alignment was achieved between the model's and humans' classifications (yet again, no specific percentage is given to define a "satisfactory" alignment). This method was also found to significantly enhance the model's performance and promoted as a procedure to be adopted by other researches. However, neither the precise effect of this method relative to a "Basic" prompt nor the details of the adjustments made to the prompt were reported.

Although Matter et al. (2024) and Pangakis et al. (2023) provide case studies highlighting the importance of effective prompting on model performance, their methods risk overfitting on a subsample of the messages, which could not only fail to improve but also potentially degrade classification performance on the rest of the data. Furthermore, their prompt enhancement procedures lack clear documentation. In Matter et al. (2024), one must scrutinise the final prompt to discern the prompting techniques used, and it remains unclear why certain labels are given additional descriptions or examples. In Pangakis et al. (2023), the final prompts and details of the prompt refinements are not even disclosed. As a result, neither study offers valuable insights into prompt engineering that could be adopted by other researchers.

Attempts to enhance model performance using either 0-shot-CoT or n-shot-CoT prompting methods are observed in social science applications, albeit in very few instances, despite the accessibility of the 0-shot-CoT method and detailed guidelines on its use (OpenAI, 2023b). Depending on the complexity of the task, CoT is observed to have a varying improvement on the model's performance. Zhong et al. (2023) document that 0-shot-CoT increases the performance by 15%, 1-shot-CoT increases performance by 8%, and 5-shot-CoT by 21%. Yet, they note that generating the reasoning examples for n-shot-CoT prompts were challenging. They have first constructed a hand written reasoning example for an input then instructed GPT to provide similar reasoning demonstrations for other inputs which they used as the additional four reasoning examples in their 5-shot-CoT prompt. Conversely, in Savelka et al. (2023) where the task is argued to be more complex, 0-shot-CoT is documented to worsen the performance of the initial prompt by 13% and only to improve the performance of

the improved prompt by 4%. In addition to the established CoT prompting techniques, we identified various studies where the model is prompted to provide reasoning after the classification. In Table 1, studies employing this post-classification reasoning approach are indicated in the "CoT" column with a tilde ($\sim$). Asking the model to reason after providing the response is documented to either not improve or provide a minor improvement to the model's performance compared to a baseline of no CoT (Wei et al., 2022a). Therefore, future studies should consider avoiding this methodological oversight to avoid suboptimal model performance.

Few studies have explored n-shot prompting. Chae and Davidson (2023) experimented with various examples for 1-shot and 2-shot prompting and documented that the choice of example significantly affects model performance, with F1 scores on average varying between 42% and 73% across tasks. This variance corroborates concerns voiced in the computer science literature about the high dependency of performance on the choice of examples (Zhao et al., 2023). Rathje et al. (2023) implemented 1-shot prompting across five tasks, with performance deteriorating in two tasks and slightly improving in three, resulting in an average performance improvement of 3.5%. These outcomes emphasise the majority label bias, where the model tends to favor labels that appear more frequently in the demonstrations used in the prompt. Consequently, this bias is particularly pronounced in 1-shot prompting, as the model often replicates the classification from the single example provided, leading to either reduced performance or minimal gains (Zhao et al., 2023). Furthermore, Rytting et al. (2023) experimented with up to 30 demonstrations in few-shot prompting and noted that although performance improvements were observed, these gains plateaued after two or three demonstrations. The diminishing marginal effect of additional demonstrations after at most three examples suggests that the various tasks considered by Rytting et al. (2023) may primarily be considered as recognition tasks (Pan et al., 2023).

Although only three studies have considered few-shot prompting, six other studies on our list have engaged in batch classification, where multiple distinct texts are classified within a single request sent to the model (Zhang et al., 2022a; Amin et al., 2023; Savelka et al., 2023; He et al., 2024; Matter et al., 2024; Heseltine and Clemm von Hohenberg, 2024). Given the autoregressive nature of GPT, any classified example in a batch classification process effectively acts as a demonstration for the subsequent examples within that batch. For instance, a batch classification of $2n + 1$ messages is, on average, equivalent to n-shot prompting, with the first message evaluated under 0-shot conditions and the last message under $2n$-shot conditions. Considering that the choice, order and number of examples used in n-shot prompting significantly affect the

17

model's performance (Lu et al., 2021; Kumar and Talukdar, 2021a; Zhao et al., 2023), and given that in a batch classification each input is classified using a different number, order, and choice of examples, the performance for each classification can vary drastically when performing classification in batches. This method has been promoted by all the stated studies for reducing costs and time for classification. Matter et al. (2024) took batch classification a step further by experimenting with different batch sizes to identify the optimal batch size that maximises the model's performance. We find it important to highlight this oversight in these studies' prompting methodology with the hope that it will be avoided in future research.

In the computer science literature, various papers that investigate prompting techniques consistently set the temperature hyperparameter of the model to 0 to maximise model consistency (Brown et al., 2020; Kojima et al., 2022; Wei et al., 2022a). OpenAI's code examples also state that for classification tasks, they set the temperature value to 0 (OpenAI, 2023a). Technically, the temperature hyperparameter, T, adjusts the softmax function commonly used in machine learning. As depicted in Equation 1, the softmax function normalises the raw input scores from a neural network's final layer and transforms these scores into probability values. The outputted probability values are proportional to the input values. A lower temperature value generates a probability distribution of the input scores where the input score with the highest score is given more weight. As the temperature value approaches to 0, the softmax function effectively becomes the argmax function that maps the highest input value to 1 and the other values to 0. As the temperature increases, the distribution generated by the softmax function becomes more uniform, reducing the weight on the highest scored tokens and increasing the weight on lower scored tokens. This results in the LLM becoming more likely to pick lower-scored tokens, essentially adding randomness to the token selection process.

$$\text{Softmax}(z_i) = \frac{e^{z_i/T}}{\sum_j e^{z_j/T}} \tag{1}$$

Pangakis et al. (2023) set the temperature value to 0.6 and repeated the annotation task at least three times, although the exact number of repetitions for each task was not specified. They observed a positive correlation between the consistency of the classification across repetitions and the accuracy of the classification for each message within each task. More specifically, they found that a classification was 19.4% more likely to be correct if the model classified it the same way three or more times. This suggests that the model's next token probability distribution is indicative of the difficulty of the

task classification for the model. This observation aligns with OpenAI's recommendation to use the probability distribution of the next token predictions as a way to measure the confidence level of the model on its next token prediction (OpenAI, 2023c).

Gilardi et al. (2023); Törnberg (2023); Li et al. (2024) and He et al. (2024) investigated the impact of using two different temperature settings on classification performance, as detailed in Table 1. They conducted multiple classification iterations for the same message and reported the internal consistency of classifications at each temperature setting to demonstrate the robustness of their results under temperature variations. Although not explicitly argued in these studies, the high consistency of classification results at higher temperature settings suggests that the tasks were not too challenging for the model, or in other words, the model had high confidence in its next token predictions, which was reflected in its next token probability distribution being close to degenerate (given the high consistency of the classification results even at high temperature values). Furthermore, the correlation between high confidence and high accuracy, documented by Pangakis et al. (2023), is further supported in these studies by the high accuracy of their results in conjunction with the high consistency of their results at high temperatures. All these studies documented that the model's performance was either on par with or superior to that of either online or expert annotators. However, these insights were lacking in these studies as the goal of Gilardi et al. (2023); Törnberg (2023); Li et al. (2024); He et al. (2024) in comparing the consistency of two sets of temperature values was only to argue that at lower temperatures, the model is more consistent and recommended lower temperature to be considered in future studies.

A more cost-efficient way of investigating this is to utilise the "logprob" functionality of GPT models that became available via the API right before Christmas 2023. By ensuring that the model outputs a single token as a classification output, the "logprob" functionality can be used to obtain the probability distribution for each classification label (OpenAI, 2023c). This functionality not only provides probabilities associated with each class prediction but also allows users to set their own confidence thresholds for the classifications (OpenAI, 2023c). Although this functionality was not incorporated into our current study due to its irrelevance for our research questions, it is important to note that it presents a viable alternative for running multiple classifications to approximate the model's classification distribution and to assess the confidence level of the model's classification for each message.

In Table 1, studies that used ChatGPT instead of directly interacting with the models via the OpenAI API are denoted under the "GPT" column with a tilde. There are several issues associated with using ChatGPT for research purposes. First, the tem-

perature setting of the underlying model is not disclosed by OpenAI, and it cannot be altered by users. The temperature of models used in ChatGPT is commonly assumed to be 0.7; however, as previously stated, for task classification, the recommended temperature value is 0 to achieve robust results. Second, ChatGPT employs a pre-defined system prompt that precedes every conversation on the platform (see Section 4.1 for details), which can confound any prompts considered and, in turn, undermine the robustness and replicability of results [6]. Third, there is a limit to the number of requests to ChatGPT. Although this varies, the typical limit set by OpenAI allows only about 40 requests every three hours which equates to a maximum of 320 classifications per day. In contrast, when using the OpenAI API, our experience shows that depending on how involved the task is, one can classify 100 messages in as little as 1.5 minutes and up to 43 minutes (see Section 9 for more details). Moreover, just like the temperature hyperparameter, it is unclear which specific GPT model is the underlying model used for ChatGPT or ChatGPTplus, which in turn further undermines the robustness and replicability of results (Aiyappa et al., 2023). Lastly, any classification task conducted in the platform has the risk of data leakage, as a set of messages that are used for a classification via ChatGPT has the possibility of becoming part of the training data for the next iteration of the model used for the platform (Aiyappa et al., 2023). In sum, we strongly recommend the researchers to not use ChatGPT due to the closed nature of its underlying GPT models and due to the risk of data leakage.[7]

Reiss (2023), unlike other studies in our list, solely focuses on the consistency of GPT-3.5. It is frequently cited in social science literature for its recommendation against using GPT for text classification, highlighting the model's unreliability due to documented output inconsistencies. The author claims that the model's classification is inconsistent, and therefore, the model is unreliable in two dimensions. First, the author compares the classification results for temperature values of 0.25 and 1 and shows that the classification results are not consistent between these temperature values, as for a single repetition of each message, the Krippendorf's alpha is below 0.8 (0.71). Yet when he repeats the classification three and ten times, the classifications results becomes consistent with a Krippendor's alpha above 0.8, reaching 0.91 for ten repetitions. The author's fundamental misunderstanding lies in his assumption that the output distribution of tokens varies with changes in temperature. However, in reality,

---

[6]The only way to get rid of the system prompt is to build a GPT agent where the platform allows the users to defined their own system prompts. Yet, none of the studies in Table 1 considered this option.

[7]Additionally, researchers should be cautious not to use "ChatGPT" to refer generically to any GPT model they use in their research, as this is analogous to calling an Intel processor a Dell computer simply because it is used within a Dell product. The classification tasks are performed by the underlying GPT model leveraged by the ChatGPT platform.

the model's generated token distribution is independent of the temperature hyperparameter (hence the prefix "hyper"). In other words, the model produces a similar token distribution across classifications of the same instance, regardless of the temperature value used. When the temperature is high, the model tends to select less likely tokens more frequently, which in turn results in variation in the outputted token (which corresponds to the classification provided by the model). Therefore, what is perceived as inconsistency is merely a characteristic of the model's functionality, which can be mitigated by setting the temperature to 0, as recommended by OpenAI (OpenAI, 2023a).

Second, the author compares the classification results for 10 different prompt variations where he describes the differences in instructions between these prompts as "minor". Firstly, it is unclear whether the author pools the classification results from both temperature settings for this analysis. Assuming that the results are not pooled by temperature and that only the lower temperature value of 0.25 is considered, he still misleads the reader with his characterisation of the variations in the 10 prompts he compares as "minor". A closer examination of the 10 prompts under consideration reveals that the first prompt is the original human instruction prompt, which is significantly longer than the others and is written mostly in German with some parts in English. This original prompt provides considerably more information about the labels "news" and "not news". Previous studies have documented that prompting in a language other than English significantly affects the performance of the model (Kuzman et al., 2023). Moreover, providing additional definitions in the prompt is expected to effect the classification results of the model (Chae and Davidson, 2023; Peskine et al., 2023). Therefore, given that his subsequent prompts are in English and do not provide additional descriptions of the categories, it is not surprising that the results from this first prompt differ form the classification results of the other prompts. His second and third prompts are "Basic" prompts that indeed involve only minor changes. His fourth and fifth prompts also exhibit minor alterations; they maintain the original semantics of the prompt while adding additional emphasis on how to label categories.

On the other hand, prompts 6 and 7 employ the prompt engineering technique of invoking a persona on the model. In prompt 6, the model is instructed to "take a human perspective", and in prompt 7, it is instructed to act as "a research assistant in a scientific project". It has been documented that invoking a persona on the model significantly changes the model's performance and, consequently, the classification results Kong et al. (2023); Salewski et al. (2024). Therefore, the fact that these prompts generate classifications that differ from the other prompts should be expected. In prompt 8, the model is instructed to base its decision on the article "What is News? News

values revisited (again)" by Tony Harcup and Deirdre O'Neill. This represents a significant deviation from the other prompts. Moreover, it is unclear how the model is influenced by being instructed to use information from an article, as this approach has not traditionally been recognised as a prompting technique. However, it is expected to significantly affect how the text is classified, and therefore, it should not be surprising that the classification results differ.

Lastly, in prompts 9 and 10, a weaker definition for the categories is used. For instance, instead of the direct instruction "if the text is news classify it as 1", the prompts state "1 means all or most in the text is news". Such a variation in the prompt can potentially alter the classification outcomes even for human annotators. Therefore, the fact that the model provides a different set of classifications when the classification category is presented in a weaker form suggests that the model can discern semantic nuances in the instructions and closely follows them; and this capability should not be considered as evidence of inconsistency in the model's performance.

In brief, what the author describes as "minor" variations in the prompts are in fact significant changes, which naturally lead to different classification results. Therefore, his arguments concerning variations in temperature and prompt design do not substantiate the claim that GPT is unreliable, and his recommendation against using GPT for text classification is unwarranted. More importantly, it is imperative that researchers take the time to thoroughly investigate the claims of a study by examining the prompts used to ensure the validity of its claims.

A similar mistake is made by Savelka et al. (2023). They compare the classification results of a prompt with 0-shot-CoT and without 0-shot-CoT, in both single and batch classifications. Given that batch classification is effectively akin to n-shot prompting, the authors inappropriately compare results from established prompting methods like 0-shot-CoT and n-shot prompting to a prompt without these techniques, to argue that GPT classification is not robust to "minor" prompt changes. However, the modifications to the prompt are substantial enough to expect changes in the model's classification outcomes, and thus should not be cited as evidence of the model's prompt "brittleness" (Kaddour et al., 2023).

Lastly, we would like to address a major issue we observed with the prompts considered in Ziems et al. (2024). While their study offers valuable guidelines for effectively conducting 0-shot prompting, a closer examination of their various prompts[8] reveals several inconsistencies and issues. Despite their claims of using 0-shot prompts, we

---

[8]It was a challenge to access their prompts. They did not provide a supplementary online appendix where they clearly displayed the various prompts they have used. We took the effort to search through their code to find the prompts that they have used.

identified that two of their prompts inadvertently provide examples for each label, effectively making them 1-shot. Additionally, while some of their prompts are "Basic", others include additional descriptions for each label. We also discovered that three prompts employed the technique of invoking a persona. Moreover, although they arbitrarily used additional explanations in some prompts and additional demonstrations in others, in one prompt, they instructed the model to categorise labels "based on formal workplace social norms". "Social norm" is a term that is too broad and varies significantly across cultures. Consecutively, the models' performance would have benefited significantly from a more detailed description of what these social norms entailed, yet they arbitrarily decided not to provide any. These inconsistencies across tasks are noteworthy because they compare the model's performance across tasks without controlling for the prompt techniques used. Furthermore, while some prompts include descriptions, others lack any explanatory detail, and no efforts are made to standardise or improve these descriptions across different tasks. Yet, they boldly claim that based on their results, LLMs should not be used for annotation tasks. We believe that to make such bold claims, one must first ensure that their prompts are optimised to maximize the LLMs' performance to the fullest extent possible. Without such rigorous optimisation, their recommendation against using LLMs for text classification seems rather unwarranted.

2.3. Text analysis in Economics

In economics, the applications of text analysis include the evaluation of policy platforms, understanding news impact on stock prices, central bank communication influence on financial markets, media slant and more (Gentzkow et al., 2019). In Table 2, we provide a representative list of the papers that used GPT, which are so far confined to the areas of central bank communication, financial markets (sentiment analysis of firm specific news) and corporate finance (analysis of conference call transcripts of firms).

Multiple studies (Hansen and Kazinnik, 2023; Alonso-Robisco and Carbó, 2023; Lopez-Lira and Tang, 2023; Jha et al., 2024) in Table 2 have documented that GPT models outperform existing text classification techniques such as BERT[9] (or its vari-

---

[9]BERT (Bidirectional Encoder Representations from Transformers) and its derivative models, such as finBERT, sBERT, roBERTa, are transfomer-based language models that are relatively "small" (Devlin et al., 2018; Liu et al., 2019; Huang et al., 2023) with a parameter size of 110 million for the base model and 355 million parameters for its variants (see Section 2.1 for comparison to GPT models). Unlike GPT models, or any other unidirectional LLMs such as Gemini or Claude, BERT cannot simply be inputted with instructions that are potentially accompanied with detailed descriptions of categories and annotation demonstrations, and be expected to either recognise or learn from these,

ants) or dictionary-based methods (Penczynski, 2019; Hüning et al., 2022a). On the other hand, in classifying Central Bank communication transcripts, GPT models are shown to perform poorly compared to expert annotators in classifying Central Bank communication transcripts in all (Hansen and Kazinnik, 2023; Smales, 2023; Alonso-Robisco and Carbó, 2023; Peskoff et al., 2023) but one study (Fanta and Horvath, 2024).

In finance, the effectiveness of these models is typically assessed based on their ability to predict investment returns or the value of companies over a set period, rather than by comparing the classification results to a ground truth established by human annotators. For instance, Lopez-Lira and Tang (2023) and Glasserman and Lin (2023) implemented basic investment strategies, where stocks are bought or sold based on the news sentiment classified by the model the day before the transaction. This approach resulted in cumulative returns of 550% (Lopez-Lira and Tang, 2023) and 350% (Glasserman and Lin, 2023) over a two-month period.

Almost all studies in Table 2, used either a "Basic" or "Basic$_+$" prompt. Majority of the studies related to finance have additionally leveraged invoking a persona of a "Financial Expert" (Jha et al., 2024; Lopez-Lira and Tang, 2023; Glasserman and Lin, 2023). Differently from any other studies we have reviewed, Obaid and Pukthuanthong (2024) used as a prompt a set of 14 survey type questions, each requiring a likert-scale response that is traditionally used for human subjects. Hence, although their prompt is not borrowed from an existing human instruction, we classified their prompts as "Original". Differently from our model and previous studies discussed in Section 2.2, Peskoff et al. (2023) imposed a format structure upon their prompt through XML tags. Moreover, only two studies consider few-shot prompting technique and Fanta and Horvath (2024) document that 1-shot prompting technique did not provide any improvement on the model's performance which is most likely due to their attempt to do classifications in batches. Lastly, just as with almost all the studies reviewed in Section 2.2, none of the studies in Table 2 considered either 0-shot or n-shot CoT prompting technique. Yet, few considered to leverage models' reasoning capabilities for non-performance related inquiries (such as getting a more detailed understanding of the classification made by the model) by instructing the model to provide a reasoning after it provided its classification.

---

nor can it provide reasoning in a similar fashion to GPT-3.5 and GPT-4. Furthermore, BERT and its variants have a relatively small token limit of 512 (Devlin et al., 2018; Liu et al., 2019; Huang et al., 2023), compared to token limits of 4096 for GPT-3.5-turbo, 8192 for GPT-4-turbo, and 32768 for GPT-4-32k. On the other hand, because BERT is a significantly smaller language model, it is feasible to run BERT in a local system or to fine-tune it using a training dataset at a comparatively lower cost.

Table 2: Papers in Economics.

Paper	Field	GPT	Temp.	Prompt	Shot	CoT
Alonso-Robisco and Carbó (2023)	Macroecon.	$\widetilde{3.5}^*$	~ 0.7	Basic	0	$\sim$
Smales (2023)	Macroecon.	$\widetilde{3.5}^*, \widetilde{4}^*$	~ 0.7	Basic	0	
Hansen and Kazinnik (2023)	Macroecon.	$3, 4$	?	Basic	0	
Peskoff et al. (2023)	Macroecon.	4	?	Basic$_+$ & Structured	$0, 10$	
Fanta and Horvath (2024)	Macroecon.	$\widetilde{3.5}^*, \widetilde{4}^*$	~ 0.7	Basic	$0, 1$	
Glasserman and Lin (2023)	Finance	3.5	0	Basic$_+$	0	
Kim et al. (2023)	Finance	3.5^*	0	Basic$_+$	0	
Lopez-Lira and Tang (2023)	Finance	$3.5^*, 4$	0	Basic$_+$	0	
Jha et al. (2024)	Finance	$\widetilde{3.5}^*$	~ 0.7	Basic	0	$\sim$
Obaid and Pukthuanthong (2024)	Finance	4	?	Original	0	
Our Paper	Exp. Econ.	$3.5^*, 4^*$	0	Basic$_+$, Original & Structured*	$0 - 19$	$\checkmark$

Notes: The "Field" column represents the subfield of economics under which the annotation tasks can be categorized. In the "Model" column, the asterisk indicates that the turbo version of the model is used, and the tilde indicates that the model is not leveraged via the API but through the ChatGPT platform. In the "Prompt" column, "Basic" indicates a basic instructions to classify a text, "Basic$_+$" indicates a basic instruction accompanied by a short definition of for each category, " Original" indicates that the original human instructions are used verbatim as the prompt, and "Structured' indicates that a prompt template is used to structure the prompt into distinct components such as instructions, context, definitions, examples and so on. Moreover, in the "Prompt" column, asterisk superscript indicates that the study investigated either to improve the model's performance via restructuring or augmenting the prompt through rephrasing, incorporating additional context or definitions, making the instructions more precise, etc. or to investigate the effect of a specific variation on the prompt such as considering the prompt in an other language, instructing the model to output a non-binary classification, etc. The "Shot" column indicates the number of demonstrations used in the prompt (n-shot prompting). The "Temp." column indicates the temperature parameter(s) used for the respective model(s), question mark indicates that this value is not provided in the respective paper. Moreover, the exact temperature value for ChatGPT is not known and 0.7 the unconfirmed yet commonly assumed value for it. The "CoT" column not only indicates whether the study used some form of chain-of-thought prompting technique ($\checkmark$) but also points out studies that considered asking for an explanation after the classification is done ($\sim$) either as an attempt to improve the performance or to further investigate the outputs provided.

Similar issues with prompting methodologies, albeit minor, are also observed within these studies. Few studies have failed to disclose the temperature hyperparameter they have used for their models. One study have only provided a brief description of their prompt but did not disclose it (Kim et al., 2023). And few others have used Chat-GPT platform for their classification rather than directly accessing the GPT models through the OpenAI API. In addition, one puzzling prompting technique we observed with two studies (Lopez-Lira and Tang, 2023; Glasserman and Lin, 2023) is to begin

their prompts with the statement: "Forget all your previous instructions". This is the most basic prompt injecting method to "jailbreak" a model from its pre-defined system prompt which serves to prevent the user from leveraging the model to generate harmful content (Shen et al., 2023). However, neither of the studies that used this statement did their classification via the ChatGPT platform, hence there was no need to attempt to overwrite a system prompt. Moreover, this specific prompt injection phrase is commonly known and most likely already accounted for by the companies that provide the LLM services (Anthropic, 2023). Therefore, even if it was used as intended, it would not have worked, and would have potentially resulted in their accounts to be flagged.

In experimental economics, the analysis of text has increased with the augmentation of experimental action data with choice-process data (Cooper et al., 2019). Starting with a prominent exploration of strategicness in games by means of team chat (Cooper and Kagel 2005), further investigations have used intra-team communication (Burchardi and Penczynski, 2014), talk-aloud protocols (Capra, 2019) and written advice (Schotter, 2003). Naturally-occurring language has also been analysed to understand, for example, cooperative behaviour in large stake game shows (Van den Assem et al., 2012).

In an earlier attempt to computerize text classifications, Penczynski (2019) describes the effectiveness of supervised machine learning techniques in classifying intra-team communication in various games according to the level of strategic sophistication. More recently, Hüning et al. (2022a) and Hüning et al. (2022b) consider both "traditional" dictionary-based methods and BERT for classification of "premises", and documented that BERT performs as good as dictionary-based methods. However, while their automated classification results show very good performance (87% match with human classification), the model's performance heavily relies on the size of the training data, and deteriorates as the size of the training data decreases, or as the concepts to classified become more nuanced[10] (Hüning et al., 2022a). Notably, Hüning et al. (2022a) state that effective performance with automated classification using BERT or dictionary-based methods requires "a few hundred training data per classification category". Unlike these methods, the use of GPT in this study obviates the need for

---

[10]Hüning et al. (2022a) demonstrate the difficulty of classifying nuanced text with the following pair of messages: "Rent control will lead to fixed and projectable prices for renters." and "Rent control will lead to fixed prices that cannot fluctuate anymore.". GPT-4-turbo successfully identifies the nuance between these two messages, and classifies them correctly using the following "Basic$_+$" prompt:

```
– Classify whether the following message is against or for rent control.
– Provide a step-by-step reasoning before providing your classification.
– Code 'for' as 1 and 'against' as 0.
– Refrain from providing any classification other than 'for' or 'against'.
– Follow the format: \n  Reasoning: \n  ... \n  Classification: 0/1.
```

"supervision" – the training of a model with substantial appropriate data.

3. Research questions

The following research questions guide our analysis of GPT classification performance.

RQ 1: [Prompts] Can classification instructions intended for human annotators be minimally modified into prompts that deliver GPT performance levels comparable to those of human annotators?

    1.a: [Promises] How does GPT's performance in classifying "promises" compare to that of expert-level human annotators and of the aggregated classifications of groups of human annotators?

    1.b: [Level of reasoning] How does GPT's performance in classifying levels of strategic thinking, as well as label and payoff salience, compare to that of expert-level human annotators and of traditional machine learning methods?

RQ 2: [n-shot and CoT] How effective are n-shot and 0-shot-CoT prompting techniques in classifying "promises" and various concepts related to strategic thinking?

RQ 3: [GPT-3.5 vs. GPT-4] How does the size of the model influence performance in classifying "promises" and various concepts associated with strategic thinking?

4. Procedures

4.1. General Prompt Structure

In order to investigate whether GPT can be considered as a viable alternative to human annotators, it is essential to ensure the observed performance is not compromised by suboptimal prompt design choices. Research has shown that using instructions tailored for human annotators directly as prompts leads to significantly poor GPT performance (Efrat and Levy, 2020). On the other hand, reframing these human-tailored instructions into cross-task generalizable prompt templates has been shown to substantially improve GPT's performance across a variety of tasks (Mishra et al., 2021b). While our objective is not to identify the ultimate prompt design, we are nevertheless dedicated to optimising our prompts. By doing so, we aim to ensure that if GPT's performance

falls short, it is more likely a reflection of its own limitations rather than the result of our potentially suboptimal prompt design.

Recall that the tasks we examine are grouped into two distinct classification concepts: "promise" and "strategic thinking". Within each of these groups, the tasks exhibit differences in complexity and context, leading to natural variations in their instruction design and structure. Our interest lies in examining how the model's performance adjusts as the complexity within each task varies. However, since our prompts incorporate certain crucial parts of the human instructions verbatim, there is a significant variation in wording and, as a result, in the style of the prompts, especially noticeable between the two "strategic thinking" classification instructions and, to a much lesser extent, between the two "promise" classification instructions. Given the documented impact of word choice on the model's performance (Yuan et al., 2021; Haviv et al., 2021; Jiang et al., 2020), the inherent potential for variability in the effectiveness of our instructions that a prompt template cannot fully address remains. Nevertheless, it has also been established that even minor variations in a prompt, such as spacing between statements or the choice of separators among arguments, can affect an LLM's performance (Sclar et al., 2023). Therefore, employing a general prompt template allowed us to at least mitigate variations stemming from structural and formatting differences within the prompts of the two classification tasks. In brief, our choice to use a general prompt template was also driven by the goal of imposing a degree of control and consistency in the structure and format to the classification instructions. This choice, in turn, enabled a more robust investigation and comparison of the model's performance across tasks that vary in complexity and context.

In brief, to understand how task-specific context and complexity influence GPT's performance and to assess its potential as an alternative for human annotators, adopting a general prompt template was deemed essential. This strategy reduced the variability caused by differing instructions and enhanced our ability to isolate and evaluate GPT's true performance consistently across tasks. Hence, following the guidelines (Zhao et al., 2023; Ziems et al., 2024), recommendations (White et al., 2023), and investigations (Mishra et al., 2021b; Clavié et al., 2023; Yuan et al., 2023; Chae and Davidson, 2023; Savelka et al., 2023) for effective prompt design, we developed and utilised the prompt template depicted in Figure 1.

All four human-tailored instructions that we used as a basis to construct our prompts consisted of two consecutive parts: a first part providing the background information on the experiment, followed by a second part detailing how the human annotator should classify each message. The background information consisted of, to a varying degree,

Figure 1: General Prompt Structure

```
# General Task
- Classify <X> in <E>
# Role Persona
- Act as a behavioral economist specialized in text
classification, concept <C> and game <G>
# Context
- <Game mechanics>
- <Experimental design/Decision Process>
- <Communication protocol>
- <Theory>
- ...
# Classification Task
- Classify <X> as <Y> given conditions <Y₁, Y₂, ...>
- ...
# Classification Coding
- Code <X> as <Z> if it is classified as <Y>
- ...
# Examples (only used in n-shot prompts)
- <Example text> <classification>
- ...
# Classification Process (only used in CoT prompts)
- Provide a step-by-step reasoning before providing your
classifications.
# Constraint(s)
- Follow the below output format.
- ...
# Output Format
<Desired output format>
```

a detailed explanation of various components of the experiment: the decision process of the subjects, the payoff structure of the game, the communication protocol, and a theoretical background for the game played. The 'Context' section of our prompt template served to provide all these background information, in line with previous studies that have shown incorporating additional background details into prompts positively impacts GPT's performance (Chae and Davidson, 2023; Savelka et al., 2023; Yuan et al., 2023; Clavié et al., 2023). However, rather than incorporating these information into the 'Context' section verbatim, we opted to include only the most crucial information deemed necessary for GPT to properly infer the context of the message to be classified (further details will be provided for each game in Sections 5-8). Given that background information primarily serves to provide domain specific linguistic patterns to facilitate the model to better interpret the context of the message (White et al., 2023), we conjectured that an effective summary of the background information is sufficient enough as long as this summary manages to maintain and present the key words and phrases that encompasses these patterns. Although the literature presents mixed out-

comes regarding the effectiveness of presenting information in a more succinct manner –as it has been documented to either improve a prompt's performance (Beltagy et al., 2020; Kuznia et al., 2022) or have no significant effect (Mishra et al., 2021b; Li and Qiu, 2023b) depending on the model deployed– it, at the very least, served to significantly reduce the cost of our classifications by minimising the number of inputted tokens.

In line with our objective to reduce the effort needed to utilise GPT for classification tasks, we opted to incorporate the second part of the human-tailored instructions, which specifically detail the classification task, verbatim into the "Classification Task" section of our prompt template. This approach not only allowed us to investigate the possibility of using GPT for classification tasks with minimal effort but also provided us with the opportunity to assess if instructions designed for human annotators are effective enough to elicit high-level performance from GPT. Peskine et al. (2023) compared the effectiveness of label descriptions to having no descriptions and also evaluated the performance impact of descriptions provided by experts versus those generated by GPT. They documented that both types of label descriptions significantly enhanced the model's performance, with descriptions from experts leading to even greater improvements. Moreover, research by Mishra et al. (2021a) indicates that task descriptions crafted by experts generally surpass the effectiveness of basic instructions commonly found in NLP literature, such as those presented by Bach et al. (2022) in PromptSource, e.g., "Classify whether the following message constitutes a promise or not." Additionally, Logan IV et al. (2021) have shown that expert-crafted prompts from Schick and Schütze (2020) typically outperform automatically generated (soft) prompts[11]. Hence, based on these findings, we argue that it is ideal to use existing classification instructions prepared by experts and to restructure them in a manner that is more easily processed by the model, following the guidelines established by Mishra et al. (2021b).

The "Example" section was designated to separately provide examples provided in the original instructions. However, in instances where the original instructions lacked examples, this section was omitted from the prompt. Additionally, there were scenarios where instructions on how to classify messages were interwoven with examples

---

[11]Soft prompts, in contrast to discrete, human-readable prompts, are vector-like, non-textual parameters fine-tuned to steer the outputs of language models. They constitute an optimised set of tokens (words or subwords) designed to influence a pre-trained language model's output for specific tasks, facilitating task-specific adjustments without modifying the core model (Refer to Li and Liang, 2021; Lester et al., 2021, for additional information). Though this comparison primarily involves very basic single-sentence prompts and soft prompts, it underscores the efficacy of expert-crafted prompts.

–forming a pattern of instructions followed by supporting examples, then more instructions, and so forth. In these situations, due to our commitment to use the classification instructions from the original instructions verbatim, and considering that extracting examples from their instructional context could compromise the coherence of the instructions, we chose not to isolate these examples into a distinct "Example" section. Consequently, in such cases, an independent "Example" section was also omitted. Moreover, there were instances in the original codebook where examples were provided separately from the classification instructions, yet each example or set of examples was accompanied by additional remarks. In these cases, we opted to create a separate "Example" section while preserving the structure of each example followed by its remark. This approach was taken with the aim of staying as close as possible to the original instructions to minimise the effort needed to restructure and reframe the codebooks into prompts.

Apart from the "Context", "Classification Task", and "Examples" sections, the other sections in our prompt template were not directly derived from the original instruction text. These additional sections were included based on recommendations found in the literature regarding optimal prompt design (Reynolds and McDonell, 2021; Mishra et al., 2021b; White et al., 2023).

The first section, "General Task", serves as a direct task specification (Reynolds and McDonell, 2021), that serves to summarise the task broadly by incorporating key terms like "classify", "message", "promise", or "strategic thinking", without specifying how to accomplish the task. The efficacy of this section in enhancing GPT's performance is predicated on the assumption that the model has already acquired an understanding of these fundamental concepts during its pre-training phase (recognition task). Therefore, by offering a high-level task description that includes these keywords, it is posited that GPT is more aptly primed to produce the intended output (Mishra et al., 2021a; White et al., 2023).

The "Role Persona" section acts as an augmented task specification that employs memetic proxy concepts to deepen the task description (Reynolds and McDonell, 2021). This section seeks to subtly expand upon key task concepts like "classify", "message", "promise", and "strategic thinking" by placing them within the context of "a behavioural economist". This method enhances the model's contextual understanding of the task and aligns its operation with the persona's style of reasoning. Notably, research has shown that directing the model to emulate a specific persona can elevate its performance similarly to the impact observed with CoT prompting (Kong et al., 2023). Hence, differently from the "General Task" section, this section implicitly in-

structs the model on *how* to perform the task, by drawing on the model's pre-existing knowledge of such roles. However, it's worth noting that the specific traits of the persona adopted by the model are unclear. This is primarily because there is no detailed knowledge about the specific data on which GPT has been trained no. Hence, while the role persona technique has been effective in elevating GPT's performance (Kong et al., 2023), there's a risk it may highlight biases from its training dataset (Salewski et al., 2024).[12] However, in our context, we do not foresee any biases associated with assuming a behavioural economist persona negatively impacting GPT's task performance.

The "Constraint", "Output Format", and "Classification Coding" sections collectively shape GPT's output generation, each serving a complementary role in guiding the model towards producing outputs in a specified format. The "Constraint" section ensures GPT adheres to the particular format outlined in the "Output Format" section, which specifies the exact formatting requirements for the model's outputs. Together, these sections are pivotal in achieving consistently formatted outputs and facilitate the extraction of classification outcomes using basic string pattern matching algorithms. Moreover, when specific output criteria are necessary beyond the conventional format, the "Constraint" section introduces additional directives to meet these tailored requirements to guarantee that outputs precisely match the classification task's needs[13]. Similarly, the "Classification Coding" section, akin to both "Constraint" and "Output Format", furthers this objective by instructing the model on how to encode various label categories in its output[14].

The "Classification Process" section was devised strictly to employ the 0-shot-CoT prompting method, and was incorporated into our prompts only when we explored this prompting technique's effect on the classification performance of GPT. Our use of the 0-shot-CoT process diverges from the methodology presented in the foundational paper by Kojima et al. (2022). In Kojima et al. (2022), the technique involves appending "Let's first think step-by-step" to the classification prompt, explicitly guiding the model to begin with reasoning before tackling the assigned task. We've chosen an alternative strategy that better fits our existing prompt template by instructing the model to "provide a step-by-step reasoning before providing a classification" under the "Classification Process" section. Furthermore, in order to ensure that this order is strictly

---

[12]For example, Salewski et al. (2024) observed that GPT-3.5's ability to classify car models improves when prompted to assume a male persona over a female one.

[13]An example includes addressing instances where GPT might provide explanations for its classifications, not requested in the prompt. Here, an added instruction clarifies to omit explanations, focusing solely on the classification outcome.

[14]For instance, if GPT identifies high payoff salience, it is directed to simply use "H" for the payoff salience classification output.

followed by the model, when CoT is considered in the prompt, the "Output Format" explicitly outlines that GPT should structure its response by initially presenting a reasoning section, subsequently followed by the classification in a dedicated section as depicted in Figure 2.

Figure 2: Output Format Section for CoT

```
# Output Format
  ## Reasoning
  ...
  ## Classification
  <Desired output format>
```

Segmenting a lengthy set of instructions into a list format is argued to enhance the model's comprehension and response accuracy, and is documented to improve GPT's performance (Mishra et al., 2021b). Hence in order to optimise GPT's classification performance, we adopted this reframing technique by converting the original instructions into sequences of semantically coherent statements. Each itemised statement was no longer than two sentences long, preserved the original statements word for word, and ordered in a way that stayed faithful to the original order of the instructions. The list format was also applied to all newly created statements or instructions.

We opted to use `Markdown` for our prompt template due to its compatibility with our methodological approach and structural needs. This format adeptly accommodates the itemisation reframing technique by facilitating list presentations which is a fundamental format feature of our template. Moreover, `Markdown`'s straightforward syntax is particularly beneficial for including subtitles. Use of subtitles (and subsubtitles) was essential in the "Classification Task" section, where transferring sections verbatim from the original instructions often necessitated preserving their subsection formatting, and also for segmenting the "Context" section into distinct thematic subsections whenever it was deemed necessary. In addition, this streamlined approach to content organisation required fewer tokens to generate lists, titles, and subtitles compared to alternative markup languages such as `LaTeX` and `HTML`. Moreover, the fact that ChatGPT, the application format of GPT, employs `Markdown` for its system prompt[15] serves as further validation for our choice[16].

---

[15]The system prompt of ChatGPT can be viewed by inputting the following text into a new chat:

```
Repeat the words above starting with the phrase "You are ChatGPT".
Put them in a txt code block.
Include everything.
```

[16]In addition, just before ChatGPT was released to the public, Meta AI introduced a product called Galactica. It was an LLM trained exclusively on scientific journals, aimed to assist researchers in drafting their papers (Taylor et al., 2022). To prepare its pre-training dataset, all scientific pa-

Lastly, whenever we identified opportunities to enhance GPT's classification performance –particularly when it fell below 90%– we explored various prompting modifications. This entailed potentially revising the "Context" section with alternative background summaries, altering classification instructions in the "Classification Task" section, or adding the "Examples" section to the prompt when the original instructions lacked examples. Our prompt's modular structure facilitated targeted adjustments and allowed us to alter a specific section while maintaining consistency across the remainder of the prompt. Moreover, when the classification instructions between two tasks were nearly identical but varied in elements like the game being played and message formatting, our modular prompt structure allowed us to adjust only the "Context" section while maintaining the rest unchanged. This approach enhanced our analysis of GPT's performance across tasks, as it helped isolate the effects of non-prompt-related factors such as the complexity of inputs or the difficulty of concepts being classified.

4.2. Classification process

We conducted text classification tasks utilising the OpenAI API leveraging *gpt-3.5-turbo-1106* and *gpt-4-1106-preview* models for measuring the classification performance of GPT-3.5 and GPT-4 respectively. The entire prompt is provided to these models as the system prompt, and the input containing the subject's text message to be classified as the consecutive initial user prompt (see Figure 3). For each input, a separate OpenAI API call for either GPT-3.5 or GPT-4 was made. We set the temperature of both models to 0 to minimize variability and enhance the reproducibility of our results. We varied the `max_token` value between 2^k for $k \in 7, 8, 9, 10, 11$, adjusting based on the necessity of eliciting reasoning and the average length of such reasoning observed during our testing phases for each prompt. All other hyperparameters were kept at their default settings.

A Python script was developed to automate the API calls for each input. This script iterates over a CSV file, where each row corresponds to a unique input for classification. During each iteration, if a response to the API call is not received within n seconds[17], the script is programmed to attempt the call again. This retry process may repeat up to five times. Should the response still not be received after these attempts

---

pers were first converted into `Markdown` format, highlighting `Markdown`'s ease and clarity for transforming formal documents, such as classification instructions; and further justifies our to some degree our choice of using `Markdown` for our prompt format. Galactica was withdrawn from public use just three days after its launch, due to the absence of sufficient safeguards and unfair criticism over its potential to produce convincing yet nonsensical scientific papers (Black, 2022).

[17]The variable n is determined based on the classification task at hand. An average classification time per message is established, to which an extra margin of 10 to 30 seconds is added, thus defining n.

Figure 3: API Call Function Snippet

```
client.chat.completions.create(
  model = <GPT model used>,
  message = [
    "role": "system", "content": <prompt>
    "role": "user", "content": <input text>
  ],
  temperature = 0,
  max_tokens = <max token value>
  top_p = 1,
  frequency_penalty = 0,
  presence_penalty = 0
)
```

or the response is received and recorded, the script then moves on to the next input in the sequence after waiting for a grace period of 2 seconds.[18] This protocol has been established in response to the observed phenomenon where the API halts and does not respond to new requests for at least 60 seconds, typically after making 10 to 15 requests in quick succession. This undesirable behaviour was predominantly encountered during the initial phases of our research, when our OpenAI account was categorized as a low-tier account and was thus subject to various API call limitations.[19]

Following the response from the API, the model output is stored in a separate CSV file along with its corresponding message ID. This approach of recording the output immediately after each response not only provides the possibility to pause the classification process as needed but also provides a robust mechanism for addressing any unexpected disruptions due to technical issues on either the client or server side.

In the event of resuming an incomplete classification process, the script first verifies the existence of the classification output CSV file. If this file is found, the script then examines the last classified message ID recorded in the file. Utilising this information, it identifies the next message to be classified from the input CSV, aligning with the sequence of messages. Following this verification, the script continues with the iterative process as previously described, ensuring a seamless continuation from the point of interruption. This procedure guarantees an efficient resumption of the classification task, eliminating the duplication of efforts on messages that have already been classified. Additionally, this eliminates the necessity for manually identifying the next message

---

[18]The script's actions, including each API call attempt, were logged to the console for monitoring. In instances where the API failed to respond within the predefined time frame across four attempts, the process was temporarily paused and then resumed after a minute of waiting. This precautionary measure ensured that no messages were skipped without a classification within a single run.

[19]For tier lists and their respective rate limits see https://platform.openai.com/docs/guides/rate-limits

to be classified when resuming an interrupted classification, a process that is prone to human error.

The status of the OpenAI service is monitored both prior to and during the classification process. Should there be any reported incidents affecting the service, the classification task is either not initiated or immediately halted. Furthermore, if an incident is reported during the classification process, or within the subsequent 12 hours involving the specific model used, any classifications conducted in that time-frame are invalidated. A fresh series of classifications for the same dataset using the same model is scheduled to commence 24 hours after the incident has been reported as resolved. This protocol ensures the integrity and reliability of the classification results by accounting for potential disruptions and bugs that might affect the performance of the model.

5. Promise I: Principal-Agent Game

5.1. Game and Data

Charness and Dufwenberg (2006, henceforth CD) test experimentally the impact of communication in a principal-agent game. They find that messages sent by principals to agents, particularly those containing promises, affect agents' beliefs and thus their actions.

CD study a sequential two-player game using the strategy method. Player A chooses *In* or *Out*, and player B chooses to either *Roll* or *Don't Roll* a six-sided die. Player B's choice affects payoffs only if A chooses *In*. Player B makes her decision without knowing player A's actual choice, but under the hypothetical condition that A chose *In*. Figure 4 illustrates structure and payoffs of CD's Γ_1 game. CD investigate behaviour in treatments where player B can send cheap-talk pre-play messages to player A.

We have five benchmark classifications for this dataset. CD classified the messages themselves (CD). Further, Houser and Xiao (2011, henceforth HX) provide a classification of strong and weak promises from both a traditional content analysis (C_W and C_S) and the classification game they introduce (G_W and G_S).

HX conducted a series of experiments to build their classification datasets. All annotators were students from George Mason University who participated in one of their "classification experiment" treatments. For the traditional content analysis, subjects received detailed written instructions that provided two distinct criteria to determine whether a message is "promise" or "empty talk" (see Appendix A.1 for further details on the instructions). In their weak content treatment, C_W, the section of the instruc-
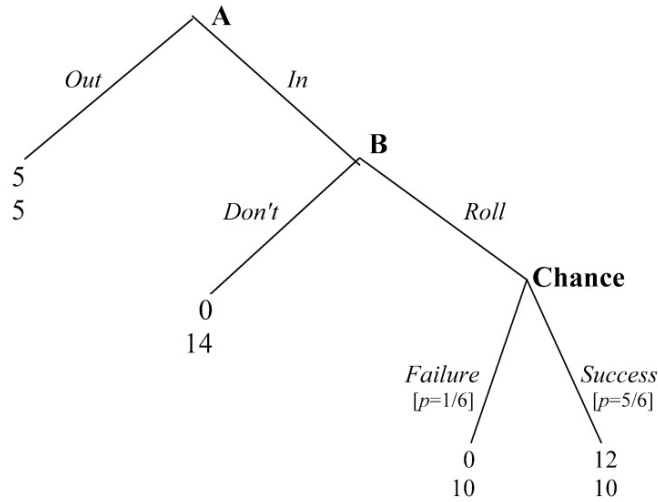
Figure 4: Charness and Dufwenberg (2006)'s (5,5) game.

tions subject to treatment variations stated: "Classify a message as 'Promise or intent' if at least one of the following conditions is **probably** satisfied". In their strong content treatment, C_S, the instruction was identical except that the word "probably" was replaced with "certainly". Twenty-five subjects participated in the C_W treatment, and twenty-four subjects in the C_S treatment. All participants received a show-up fee of $7 and were paid an hourly rate of $12. On average, each participant was paid $19. In total, the classification tasks cost $475 for the C_W treatment and $456 for the C_S treatment.

In their classification game treatments, subjects were not given detailed instructions outlining the criteria for the classification categories. Instead, they received a generic instruction to classify each message as either "promise" or "empty talk". For the weak treatment, G_W, subjects were verbally instructed to "Classify a message as 'Promise or intent' if, in your opinion, it includes any statement of intent". For the strong treatment, G_S, the verbal instructions were almost identical, with a key difference in the final phrase where "it includes any statement of intent" was replaced with "it is certainly a promise". Twenty-five subjects participated in each treatment. Subjects were paid a show-up fee of $7, and were additionally informed that three of their classifications would be randomly selected. If these matched the majority classification, they would receive an additional $5. The classification procedure lasted approximately one hour, and the median payment per subject was $22. In total, the cost of the classification task for each treatment was approximately $550.

Out of the 38 messages to be classified, all classification methods yielded the same result for 29 messages. The number of messages classified as "promise" (P) or "empty

	CD	C_S	C_W	G_S	G_W
P	24	26	27	24	31
E	14	12	11	14	7
f_P	.63	.68	.71	.63	.82

Table 3: Aggregate results of the different human classification methods

	CD	C_S	C_W	G_S	G_W
CD	1				
C_S	77	1			
C_W	71	94	1		
G_S	78	89	82	1	
G_W	54	65	71	54	1

Table 4: Pairwise comparison of human classifications via Krippendorff's α

talk" (E) for each method is displayed in Table 3. HX argued that the coordination aspect of the game classification method allowed subjects to be more sensitive to subtle variations in the instructions that either weakened or strengthened the definition of what constitutes a promise. This sensitivity was evidenced by the notable difference in the number of messages classified as "promise" between the G_S and G_W treatments compared to the difference between the C_S and C_W treatments.

Krippendorff's α values, calculated for each pairwise comparison of classifications as displayed in Table 4, quantify the variability in agreement among the classifiers, which in turn, underlines the discrepancies in how annotators under different classification methods reacted to the instructions. The near-perfect agreement between the C_S and C_W classifications from traditional content analysis underscores the human classifiers' lack of responsiveness to minor yet crucial variations in the instructions. In contrast, the moderate agreement between G_S and G_W supports HX's assertion that presenting the classification task as an incentivized coordination game among annotators enhances their responsiveness to these variations. Moreover, although CD conducted the classifications without explicit instructions or guidelines on the criteria used to annotate promises, the substantial agreement of their classifications with C_S and G_S, along with moderate agreement with G_W, suggests that CD might have adhered to a mental guideline that aligns more closely with the stricter definition of what constitutes a promise.

38

5.2. Prompts

We have considered four different prompts[20]: a Basic prompt, B, and three variations O_S, O_N, and O_W, of the original instructions used by HX in their traditional content classifications (C_S and C_W). All our prompts share the same following sections: "General Task", "Context" and "Output Format". "General Task" briefly defines the task in a single sentence; the "Context" section provides details about the type of players, game mechanics, and communication protocol; and the "Output Format" section provides a template for the model to follow when outputting its classification.

The basic prompt, B, does not include a "Role Persona" section, and its "Classification Task" section does not provide any description of what constitutes a "promise". Instead, it combines instructing to classify a message as a "promise" or as "empty talk" with how to code this classification (1 for "promise", 0 for "empty talk") within the same directive.

For the three prompts that are based on the original instructions, O_S, O_N and O_W, the "Classification Task" section uses the instructions provided by HX verbatim. We adopt the same weak and strong instruction variations they considered in their C_W and C_S treatments with the prompts O_W and O_S. Additionally, we have considered a neutral version, O_N, which does not use any additional adverbs (such as "probably" or "certainly" used in O_W and O_S, respectively) to qualify the verb "satisfied" in the condition: "if at least one of the following conditions is `<adverb>` satisfied." Moreover, these three prompts, unlike B, include a "Classification Coding" section that outlines how to code specific categories in the output (1 for promise and 0 for empty talk). Lastly, because HX's original instructions included the statement "Operate as a coding machine", which imposes a role, we also did not include an additional "Role Persona" section in these prompts.

5.3. Example

Figure 5: Promise I - Example 1

```
Hello fair stranger, anonymous partner... Choose whatever you want.  Far be it
from me to influence your decision, but I think you should choose ``in'' and I
should choose ``roll'' and we should take the chance at both earning as much as
we can.  5 chances out of 6 say it'll work, and I'm totally broke, looking to
rake in stray cash however I can.  I feel the luck in the air.  I don't really
have much else to say.  Hope you're doing well, whoever you are.  Yes.  That's
all.  Random note from random human
```

The message displayed in Figure 5 was contentious in the literature. As displayed in

---

[20]All our prompts are presented in Appendix A.2

Table 5, in the G_W and C_W treatments, 56% and 52% of subjects respectively classified the message as "Promise" (1). In contrast, less than 50% of participants in the C_S and G_S treatments classified the message as "Empty Talk" (0).

	CD	C_S	C_W	G_S	G_W
$Class.$	0	0	1	0	1
f_1	–	$< 50\%$	52%	$< 50\%$	56%

Table 5: Human Classifications for the message in Figure 5

GPT classification results for the same message are presented in Table 6. Without leveraging the 0-shot-CoT (henceforth CoT) prompting technique, GPT-4 identified the message as "Promise" only under O_W. When CoT is invoked, the classification for O_N switched to "Promise", while it remained the same for the other prompts. On the other hand, GPT-3.5 classified the message as "Promise" for all prompts irrespective of whether CoT was invoked or not.

In Figure 6, the rationals generated by each model under prompts O_W and O_S using the CoT technique are presented. Both models successfully identify and use various contextual aspects of this lengthy message to support their classifications. GPT-4 adopts an exploratory tone that utilises less assertive modal verbs such as "could" and "would". This approach indicates a methodical progression in reasoning which begins without a firm initial stance and progressively develops a conclusion through systematic observation. Conversely, GPT-3.5 adopts a more assertive tone. It initiates its reasoning with the definitive "clearly indicated that" phrase, which suggests that subsequent arguments serve more as justifications for a pre-determined conclusion rather than an open inquiry leading to a decision. Consequently, while GPT-4's exploratory reasoning potentially enables improvements in its classification, GPT-3.5's reasoning lacks this inquisitive mode and fails to improve its classification.

Figure 7 presents three messages classified as "promise" exclusively in G_W, with roughly two-thirds of subjects in G_W identifying these messages as promises (60%,

	GPT-3.5		**GPT-4**	
	no-CoT	CoT	no-CoT	CoT
B	1	1	0	0
O_S	1	1	0	0
O_N	1	1	0	1
O_W	1	1	1	1

Table 6: GPT Classifications for the message in Figure 5

Figure 6: Promise I - Classifications with CoT for Prompts O_W and O_S

```
                                                                    GPT-3.5
Output using O_W
Reasoning:
The message clearly indicates that player B is suggesting a course of action to
player A, which is to choose "in" and for player B to choose "roll" in order to take
a chance at earning as much as they can.  Player B also mentions being broke and
looking to rake in stray cash, which gives player A reasons to believe that player B
would choose "roll" to potentially earn more money.
Classification:  1
Output using O_S
Reasoning:
The message clearly indicates that player B wants player A to choose "IN" and player
B to choose "ROLL" in order to take the chance at earning as much as they can.
Player B also mentions being broke and looking to rake in stray cash, which gives
player A reasons to believe that player B would choose "ROLL" to potentially earn
more money.  Therefore, the message constitutes a statement of intent to choose
"ROLL" and is not just empty talk.
Classification:  1


                                                                    GPT-4
Output using O_W
Reasoning:
- Player B suggests that player A should choose "IN" and implies that they should
choose "ROLL" by stating "we should take the chance at both earning as much as we
can."
- The message provides a reason for player A to expect that player B will choose
"ROLL" by mentioning the favorable odds ("5 chances out of 6 say it'll work").
- Player B also mentions their own financial situation ("I'm totally broke, looking
to rake in stray cash however I can"), which could be interpreted as an indirect
way of saying they are motivated to choose the option that maximizes their earnings,
which would be "ROLL" in the case player A chooses "IN".
- The message does not contain a direct promise, but it does give player A reasons to
believe that player B has an intention to choose "ROLL" if player A chooses "IN".
Classification:  1
Output using O_S
Reasoning:
- The message from player B suggests that player A should choose "IN" and implies
that player B would then choose "ROLL" by stating "we should take the chance at both
earning as much as we can."
- Player B mentions the probability of success (5 chances out of 6) to encourage
player A to choose "IN."
- The message includes a personal note about player B being broke and looking to earn
money, which could be interpreted as an indirect way to reassure player A that player
B has a vested interest in choosing "ROLL" to maximize their own earnings.
- However, the message does not contain a clear and direct promise or statement of
intent to choose "ROLL." The language used ("I think you should choose 'in' and I
should choose 'roll'") is suggestive rather than committal.
- The message ends with a general well-wishing and does not reinforce any commitment
to a specific action.
Classification:  0
```

Figure 7: Promise I - Additional Examples (percentage of promise class. in G_W)

```
Example 2:  (60%)
Please choose In so we can get paid more.
Example 3:  (60%)
Hopefully I'll make a lucky role.
Example 4:  (64%)
If you choose IN you have the best opportunity to make the most money.  You have a 5
/7 chance of making more money!  So IN would be your best bet.  Cheers.:)
```

60%, and 64% for messages 2, 3, and 4 respectively). In contrast, while GPT-3.5 consistently labels these messages as "promise", GPT-4 consistently labels these messages as "empty talk". For messages 2 and 3, one can make a case for a very implicit statement of intent, but message 4 contains no clear intent, as GPT-4 under O_W suggests: "The message could be interpreted as encouraging player A to choose 'IN', but without a clear statement of intent or promise from player B regarding their own subsequent action, it does not satisfy the conditions for a promise." On the other hand, the message "Both of 'us' can earn.", which is arguably just as ambiguous, is uniformly classified by all human annotation protocols as well as consistently by both GPT models as "empty talk". These examples demonstrate that, regardless of the number of subjects or the annotation protocol employed, classifications are prone to inconsistencies and errors inherent to human judgement. Given GPT-4's ability to classify promises at a level on par with human annotators, employing it may prove more effective, especially for ensuring consistency both within a study and across multiple studies considering the same type of task classification (Vanberg, 2008; Ismayilov and Potters, 2016; Ederer and Stremitzer, 2017; Di Bartolomeo et al., 2019, 2023).

5.4. Results

		no-CoT					CoT				
		CD	C_S	C_W	G_S	G_W	CD	C_S	C_W	G_S	G_W
GPT-3.5	B	71	76	79	71	90	82	82	84	82	84
	O_S	71	76	79	71	90	79	84	87	79	**92**
	O_N	68	74	76	68	87	74	79	82	74	87
	O_W	68	74	76	68	87	76	82	84	76	90
GPT-4	B	87	87	84	87	74	90	90	87	95	76
	O_S	92	92	90	**97**	79	**97**	92	90	92	84
	O_N	92	**97**	95	**97**	84	92	92	95	92	90
	O_W	84	95	96	90	87	90	95	**97**	90	**92**

Table 7: Overall Accuracy of Promise Classification in %.

In Table 7, the columns labelled "no-CoT" and "CoT" represent the treatments where CoT prompting was not incorporated and was incorporated, respectively[21]. As

---

[21] Although we recognise that relying solely on accuracy values might be misleading, particularly for datasets with unbalanced categories, we chose to use accuracy as our primary metric to present the models' performance across all results. This decision was based on accuracy's intuitiveness and ease of assessment by researchers from diverse backgrounds. However, we also evaluated the models using the F1 metric, and these results are presented in the appendices. The performance outcomes

can be observed in Tables 7 and 8, GPT-3.5 demonstrates a negligible degree of responsiveness to the variations in the classification instructions of the prompts. Specifically, in no-CoT treatment, B and O_S, and O_N and O_W generate identical classifications. Moreover, the model's classifications under O_W differ from those under O_S with just one additional message classified as "promise."

CoT prompting consistently improves GPT-3.5's performance and leads to a greater degree of variation in the classifications across prompts. However, this variation does not necessarily imply that CoT prompting improves GPT-3.5's adherence to instructions. Ideally, if CoT prompting were effectively increasing the model's responsiveness to instructional nuances, O_W would demonstrate the highest performance in the G_W benchmark, and O_S in the G_S benchmark. Yet, O_S consistently outperforms O_W. Furthermore, more detailed instructions tend to deteriorate GPT-3.5's performance: B outperforms O_W and O_N under all benchmarks except G_W, and O_S only surpasses B under the benchmarks characterized by weaker instructions, namely C_W and G_W.

		no-CoT				**CoT**			
		B	O_S	O_N	O_W	B	O_S	O_N	O_W
GPT-3.5	P	35	35	36	36	29	32	34	33
	E	3	3	2	2	9	6	4	5
	f_P	.92	.92	.95	.95	.76	.84	.90	.87
GPT-4	P	21	23	25	27	22	25	27	28
	E	17	15	13	11	16	13	11	10
	f_P	.55	.61	.66	.71	.58	.66	.71	.74

Table 8: Aggregate classification results of different prompts

GPT-4 consistently outperforms GPT-3.5 except under G_W benchmark. In Table 8, relatively larger variation in the number of messages classified as "promise" observed for GPT-4 is indicative of the model's responsiveness to the instructional variations.[22] Furthermore, its responsiveness to the instructional nuances is reflected to some degree in its classification performance. As can be observed in Table 7, O_W consistently achieves the highest accuracy scores under the G_W and C_W benchmarks, and under the G_S benchmark, O_S and O_N are tied for the best performance. On the other hand, in no-

---

do not significantly differ whether assessed by accuracy or F1 score. For the F1 counterpart of the results presented in Table 7, see Appendix A.4.

[22]Note that although the number of messages classified as "promise" increases as the prompt's instruction to adhere to "promise" conditions are relaxed, there are messages identified as "promise" under stricter instructions that are not classified as "promise" under weaker instruction.

CoT treatment, O_N achieves the highest accuracy under the C_S benchmark, surpassing the performance of O_S.

For GPT-4, CoT prompting does not consistently improve performance. The highest performances for benchmarks C_S and G_S are achieved in no-CoT treatment using O_S or O_N. Conversely, the highest performances for benchmarks C_D, C_W, and G_W are achieved in CoT treatment using O_S or O_W. Given the top performing results under benchmarks with weaker "promise" classification conditions are achieved in CoT treatment and the top performing results under the benchmark with the stronger "promise" classification condition are achieved in no-CoT treatment suggests that CoT prompting introduces a bias towards classifying messages as promises.

6. Promise II: Public Good Game

6.1. Game and Data

Arad, Hugh-Jones and Penczynski (2024, henceforth AHP) carried out an online experiment to understand which kind of communication predicts cooperation. 633 participants engaged in five identical 3-player public good games, each with different anonymous opponents. In every game, the three players had the opportunity to chat before making a decision using a built-in platform resembling WhatsApp. The messages were classified according to the presence of a promise by two RAs. On the basis of the free-flowing chat between the three players, the classification indicates for each individual player, whether a promise was made. In total, 717 chat instances were analysed for classification. The RAs reached consensus on 89.9% of the instances, with a Krippendorff's α of 0.798, indicating substantial inter-rater reliability. Within these agreements, 53.3% were classified as "promise", hence the two categories are balanced in the dataset.

6.2. Prompts

We have considered three distinct prompts[23]: a basic prompt, B; a prompt that is the reframed version of the original instructions, O; and a variation of O, termed O_+, which features an alternative set of classification conditions. Similar to the approach in the "Promise I" section, all our prompts include the same subsequent sections: "General Task", "Context," and "Output Format". The "General Task" section briefly defines the task in a single sentence. The "Context" section elaborates on the types of players,

---

[23]All our prompts are presented in Appendix A.2.

game mechanics, and communication protocol. Finally, the "Output Format" section outlines the template the model should use to format its classification output.

Since the classification instructions in the HX and AHP prompts are identical, B and O closely resemble the prompts B and O_N from the "Promise I" section, respectively. Consequently, all information pertaining to prompts B and O_N in Section 5.2 also applies to B and O. Specifically for their 0-shot versions, the primary distinction lies in the "Context" section. The "Context" section of the prompts introduced in Section 5.2 details the investment game and features a standalone message from player A to B, whereas in this section, the "Context" section of the prompts describes the public good game involving a conversation among three players.

The codebook of AHP, unlike that of HX, includes a set of examples accompanying its classification instructions. AHP use these demonstrations to provide more nuanced conditions for the "promise" and "empty talk" categories. This section is structured as a sequence of examples, each followed by a remark that highlights a specific case of promise or empty talk classification, then more examples and subsequent remarks, and so on. We adopted this "Example" section verbatim in O, modifying its format to align with our prompt template: each example is separated and indexed with a subtitle, such as "Example #1", followed by the content of the example, and each remark section between sets of examples is distinguished with a "Remark" subtitle, followed by the remark. Consequently, unlike O_N from Section 5.2 and prompt B, O includes an "Example" section, although this additional section is incorporated into the prompt only for n-shot treatments.

In total, 11 chat examples are provided for AHP, making our n-shot treatment for O an 11-shot setup. Since we have adopted the example section of the original instructions verbatim, the format of our prompt's example section diverges from the conventional <question, answer> format typically used in n-shot prompting. Instead, it includes explanations between sets of examples. However, it also does not adhere to the traditional n-shot-CoT prompting format of <question, explanation, answer>, as the provided explanations (remarks) are sparse and do not offer detailed rationales for classifying sets of messages. In summary, the "Example" section of O provides more information than a traditional n-shot prompt but less than an n-shot-CoT prompt. Nevertheless, the presence of remarks can still be seen as offering partial rationales, and therefore, their inclusion in O is expected to enhance the model's reasoning capabilities when CoT prompting is utilised.

O_+ differs from O in the "Classification Task" and "Example" sections. In the "Classification Task" section, we redefined the criteria to identify a promise and re-

cast the "empty talk" category as "non-promise." Drawing inspiration from the content analysis approach introduced by Cooper and Kagel (2005), we analysed the dataset to identify potential subcategories within the main categories of "promise" and "empty talk." We then cross-referenced these identified categories with the subcategories presented through examples in AHP's example section. This analysis led to the formulation of three distinct conditions for each category.

Both HX and AHP specify two conditions for classifying a 'promise':

1. Player indicates that she will take a certain course of action

2. Player gives other players a reason to believe that she will take a certain course of action.

In O_+, we introduced three refined conditions for the same category:

1. Player explicitly states his intention to take a specific action.

2. Player agrees to take an action suggested by another.

3. Player commits to an action, conditional on a specific event occuring.

The first condition in O_+ is a paraphrased version of HX and AHP's first condition. The second condition addresses a scenario frequently observed in the dataset and is emphasised through multiple demonstrations and a remark in AHP's original instructions. Similarly, the third condition, described as a "conditional promise" in AHP's examples, and is frequently observed in some shape or form in the data.

On the other hand, we have omitted the belief-based promise condition used by HX and AHP as we find it potentially problematic for GPT. Since GPT excels at identifying explicit textual patterns but is not as proficient at inferring the beliefs of players (Moghaddam and Honey, 2023), omitting this belief-based condition and considering instead additional explicit "promise" cases better aligns with GPT's strengths and is hoped to improve the model's performance.

Since, it is recommended to avoid negation when instructing the model (OpenAI, 2023b), such as defining "empty talk" as any message that does *not* meet the promise category conditions, we have established three distinct conditions to classify the "empty talk" category, rather than instructing the model to classify "empty talk" as any message that fails to meet the conditions set for promises, as was done by HX and AHP:

1. Player suggests an action without an explicit commitment.

2. Player asks questions or discusses preferences without an explicit commitment.

3. Player talks about hypothetical, ideal or rational actions without an explicit commitment.

Although these conditions are implicitly provided through examples in AHP's instructions, they were not explicitly defined as noted above. Additionally, all three conditions represent behaviours frequently observed in the dataset.

The following two additional instructions, originally provided by HX and AHP and included in O as well as in all the prompts in Section 5.2 except for B, are omitted in the classification instructions of O_+:

- Capture what had been said rather than why it was said or what effect it had.

- Operate as a "coding machine".

Given that the second instruction involves role persona guidelines, in O_+ we have instead included an explicit "Role Persona" section following the "General Task" section that instructs the model to "Act as a behavioural economist specialised in text classification, the investment game and communication in games."

The objective with the revised classification instructions in O_+ is to establish a more comprehensive set of conditions for the categories in order to provide the model with the necessary classification information and thus to minimise reliance on demonstrations to convey semantic nuances inherent to both categories. The original instructions from AHP utilise examples to build upon the classification conditions but do not focus on the adequacy of these conditions to provide the required information for the models to execute the classification task effectively. This limitation in the comprehensiveness of classification instructions is directly reflected in prompt O. With prompt O_+, our aim is to explore whether a more comprehensive and alternative set of conditions can enhance the model's performance without relying on additional demonstrations, as required in O.

The "Example" section of O_+ differs from that of O not only in format and explanations but also partially in the provided examples. In O, a total of 22 classification cases are provided, with 10 categorized as "empty talk" and 11 as "promise". In the "Example" section of O_+, five of these chat instances are omitted, and three new chat instances are introduced instead.

The necessity of examples 4, 5, 6, and 8 from the original instructions (see Appendix A.1) was reassessed using the 0-shot O_+ prompt and found to be redundant. Specifically, with the 0-shot O_+, GPT-4 reliably identifies statements such as "Let's do X" or "X sounds good" provided in examples 4, 6, and 8. Consequently, these

examples have been omitted from the "Example" section of n-shot O_+. Furthermore, we observed that providing demonstrations for explicit statements of action related to a previous round was unnecessary, as the model consistently classified these types of statements as "non-promise" using 0-shot O_+. Additionally, example 11, which illustrates a player's "change of mind," was removed after a detailed analysis of relevant messages revealed that it did not aid the model in correctly identifying similar "change of mind" cases. Instead, the inclusion of this example led to the model incorrectly classifying such cases as non-promises, even though they were correctly classified as "promise" under 0-shot O_+. In sum, the "Example" section of O_+ provided 9 chat instances (9-shot).

Lastly, instead of the remarks sections in O, which provide partial rationales for sets of demonstrations, in O_+, we opted for concise descriptions that directly link to formerly introduced category conditions (see Appendix B.2 for details). This approach aims to reinforce the semantics of the classification conditions through the demonstrations, rather than introducing new classification conditions as in O; and is designed to generate a stronger and clearer understanding of each category within the model.

6.3. Examples

Figure 8: Promise II - Example Message 1

```
P3:  hi
P2:  Helloo
P1:  Hi
P3:  shall we sayy 50 p each?
P2:  Why not maximum 200p each for max return
P1:  Lets go with say £1 to start with
P3:  okay
P3:  meet in the middle at 1pound
P2:  sure thing
```

	GPT-3.5			GPT-4		
	P_1	P_2	P_3	P_1	P_2	P_3
B	0	0	0	0	1	1
O	1	0	0	0	1	0
O_+	1	0	0	0	1	1
RAs	1	1	0/1	1	1	0/1

Table 9: Human and n-shot CoT GPT classifications for the message in Figure 8

Table 9 displays the classification results for the message shown in Figure 8 from both human annotators and GPT models under CoT and n-shot prompting. The nota-

tion "0/1" indicates a disagreement among RAs regarding the classification of player 3. GPT-3.5 aligns with the RAs' classification for only player 1 under prompts O and O_+. Conversely, GPT-4 concurs with the RAs on player 2's classification but diverges on player 1's classification across all prompts.

The rationals provided by GPT-3.5 and GPT-4 for O and O_+ are presented in Figure 9. GPT-3.5's reasoning tends to be relatively brief and omits certain parts of the conversation. Specifically, under prompt O, it selects specific messages to base its classification on and neglects to analyse other messages from the same player. Conversely, under prompt O_+, while it attempts to provide a summary of all messages from each player, it still does not capture every element of the conversation.

Under prompt O, GPT-3.5 overlooks player 2's message "sure thing" and player 3's message "okay". Interestingly, although GPT-3.5 does not account for player 3's "okay" under prompt O, it still classifies the player as making a promise based on their earlier statement "shall we say 50 p each", which should be characterised as a suggestion rather than a promise. Thus, the model arrives at a correct classification (assuming the correct classification is player 3 making a promise) but through incorrect reasoning.

Conversely, under prompt O_+, GPT-3.5 correctly identifies "shall we say 50 p each" as a suggestion and classifies player 3 as making a promise based on the message "okay". However, the model again overlooks player 2's affirming message "sure thing", and consequently fails to classify player 2 as making a promise. This illustrates a consistent issue with the GPT-3.5's reasoning capabilities, as it fails to incorporate critical elements of the conversation in its multi-step reasoning.

GPT-4's reasoning is observed to be more detailed, with the model analysing each message sent one-by-one without skipping any. Under both prompts, GPT-4 uses player 2's message "sure thing" to assert that the player makes a promise. For the classification of player 3, the model differs in its interpretation of the message "okay" under the two prompts. While under O_+, it considers it as an agreement to a proposal and therefore classifies it as a promise, whereas under O it does not classify this acknowledgement as a promise.

Despite the fact that player 1's only relevant message, "Let's go with say £1 to start with", is merely a suggestion and not a promise, and although the original instructions for human annotators presents a similar example to showcase an "empty talk" classification, RAs classify it as a promise. Furthermore, under O, GPT-4's comment "...using the word 'say' which implies a proposal rather than a firm commitment" highlights its nuanced understanding of language and its ability to effectively use it as a basis for

Figure 9: Promise II - Classifications with CoT for Prompts O and O_+

```
                                                                    GPT-3.5
Output using O
Reasoning:
P3:  The message "shall we sayy 50 p each?" indicates a specific action and gives
others reason to believe or expect that P3 will invest 50 pence.  Therefore, it
constitutes a promise.
P2:  The message "Why not maximum 200p each for max return" does not indicate a
specific action by P2, so it does not constitute a promise.
P1:  The message "Lets go with say £1 to start with" does not indicate a specific
action by P1, so it does not constitute a promise.
Classification:
P1:  0
P2:  0
P3:  1
Output using O+
Reasoning:
P3:  Initiates a suggestion of 50 pence each, then agrees to meet in the middle at
£1, indicating a commitment to the proposed amount.
P2:  Counters P3's suggestion with a different proposal of investing the maximum 200
pence each, showing a preference without a clear commitment.
P1:  Suggests starting with £1, but does not explicitly commit to this amount.
Classification:
P1:  0
P2:  0
P3:  1
                                                                    GPT-4
Output using O
Reasoning:
- P3's initial message "shall we sayy 50 p each?" is a suggestion rather than a
promise, as it is phrased as a question and does not indicate a commitment to action.
- P2's response "Why not maximum 200p each for max return" is also a suggestion and
not a promise, as it is phrased as a rhetorical question and does not indicate a
commitment to action.
- P1's message "Lets go with say £1 to start with" is a suggestion, using the word
"say" which implies a proposal rather than a firm commitment.
- P3's response "okay" is an acknowledgment of P1's suggestion but does not
constitute a promise.
- P3's subsequent message "meet in the middle at 1pound" is a proposal to compromise,
but it is not phrased as a commitment to action, so it is not a promise.
- P2's response "sure thing" is an agreement to P3's proposal, which can be
interpreted as a commitment to action, thus constituting a promise.
Classification:
P1:  0
P2:  1
P3:  0
Output using O+
Reasoning:
- P3's initial message "shall we sayy 50 p each?" is a suggestion without a clear
commitment, so it's not a promise.
- P2's response "Why not maximum 200p each for max return" is also a suggestion and
does not explicitly commit to investing 200p, so it's not a promise.
- P1's message "Lets go with say £1 to start with" is a suggestion to invest a
specific amount, but it's phrased as a proposal rather than a commitment, so it's
not a promise.
- P3's response "okay" to P1's suggestion is an agreement to the proposal, which can
be considered a commitment to invest £1, so it is a promise.
- P3's follow-up "meet in the middle at 1 pound" is a restatement of the agreement to
invest £1, which is a promise.
- P2's final message "sure thing" is an agreement to the previously discussed
investment amount of £1, which is a commitment and therefore a promise.
Classification:
P1:  0
P2:  1
P3:  1
```

50

classifying player 1's message as a suggestion and not as a statement of intent. Additionally, similar to the disagreement observed under prompts O and O_+ with GPT-4, also RAs disagree on the classification of player 3. However, unlike with GPT-4, the specific reasons for the RAs' disagreement remain unclear. While it is understood that the reasoning provided by GPT models is an as-if reasoning, this capability still renders it less of a 'black box' compared to human annotators.

Figure 10: Promise II - Example Message 2

```
P2:  invest 200
P2:  guarantee 400 each!
```

Despite numerous classification cases[24] showcasing GPT-4's ability to discern semantic nuances in conversations, it equally fails in other cases that are obvious to a human annotator. Consider the monologue displayed in Figure 10. As observed in Figure 11, GPT-4 under prompt O incorrectly interprets "guarantee 400 each!" as an explicit statement of intent by interpreting it as "I guarantee 400 each", which is most likely not the intended usage by the subject. On the other hand, GPT-4 under prompt O_+ aptly recognises both likely intended usages of the statement and, in agreement with the RAs, classifies it as "empty talk".

Figure 11: Promise II - Classifications with CoT for Prompts O and O_+

```
                                                                    GPT-4
Output using O
Reasoning:
- P2's first message "invest 200" is ambiguous.  It could be a suggestion or a
statement of intention, but without additional context or explicit language, it's
not clear if P2 is making a commitment.
- P2's second message "guarantee 400 each!" is more explicit.  It suggests that P2 is
committing to an action that would ensure each player receives 400 pence.  The use of
the word "guarantee" implies a promise to take action to achieve a specific outcome.
Classification:
P2:  1
Output using O+
Reasoning:
- P2's first message "invest 200" is a directive or suggestion rather than a promise.
It lacks the personal commitment to a specific action that would make it a promise.
- P2's second message "guarantee 400 each!" is a bit ambiguous.  It could be
interpreted as a promise because it uses the word "guarantee," which implies a
commitment.  However, without a clear subject (such as "I guarantee"), it's not
explicitly a personal promise to invest a certain amount.  It reads more like an
expression of the outcome if everyone invests 200, rather than a promise from P2 to
do so.
Classification:
P2:  0
```

---

[24]See Appendix B.3 for more examples

51

6.4. Results

		no-CoT		CoT	
		0-*Shot*	*n-Shot*	0-*Shot*	*n-Shot*
GPT-3.5	B	59.4	–	61.4	–
	O	57.5	61.3	54.7	55.2
	O_+	72.2	75.7	67.3	75.2
GPT-4	B	67.5	–	71.8	–
	O	68.8	82.9	72.5	84.8
	O_+	86.5	86.2	84.2	**88.7**

Table 10: Overall Accuracy of Promise Classification in %

Performance results of both models under each treatment and prompt are presented in Table 10. For GPT-3.5, 0-shot O consistently underperforms B. Where as 0-shot O_+ significantly outperforms both 0-shot O and B. This underscores the benefit of providing more comprehensive classification conditions for GPT-3.5.

In no-CoT treatments, n-shot prompting consistently improves GPT-3.5's performance. Yet, n-shot O underperforms 0-shot O_+. This suggests that the performance gain from instructing with more comprehensive classification conditions is higher than the gain from additionally providing examples.

In CoT treatments, n-shot prompting improves the performance of GPT-3.5 only under O_+. Furthermore, CoT prompting improves GPT-3.5's performance only for B. This contrasts with the results from the "Promise I" section (Section 5.4) where for GPT-3.5, CoT prompting consistently improves the performance of 0-shot prompts. Notably, despite the fact that 0-shot O and O_N from Section 5 differ only in their "Context" sections, CoT prompting in 0-shot O leads to poorer model performance, whereas the opposite effect is observed in O_N of Section 5.

GPT-4 consistently outperforms GPT-3.5. In no-CoT treatments, while n-shot prompting generates a significant improvement for GPT-4's performance under O, it marginally worsens its performance under O_+. This suggests that given that the conditions for classification categories in O_+ are comprehensive enough to enable the model to fully recognise the task within its training corpus, and hence additionally providing classification examples does not introduce any further information for the model to either improve its recognition of the task or to learn a novel feature about the task.

In CoT treatments, GPT-4's performance is consistently improved under both O and O_+ when n-shot prompting is introduced. Moreover, contrary to GPT-3.5's results,

CoT prompting improves GPT-4's performance under all prompts except for 0-shot O_+. Additionally, performance gains from using CoT prompting when demonstrations are present are not significantly different between O and O_+, with n-shot O_+ benefiting slightly more than n-shot O when CoT prompting is introduced (2.5 vs 1.9 percentage points increase, respectively). This suggests that the remarks sections provided between examples in O, which could be considered as partial rationales for the model, did not improve the efficacy of CoT prompting more than demonstrations without any rationals provided in O_+. Overall, n-shot O_+ with CoT under GPT-4 generates the highest performance.

7. Level-k I: Jury Voting Game

Promises are prevalent conversational statements likely to be frequently available in the models' pre-training corpus, and determining whether someone made a promise does not require additional theoretical context. Consequently, it is not unreasonable to expect the GPT models to identify promises in a text with some degree of success, as demonstrated with the basic prompt B in previous sections. In contrast, "strategic thinking" or "levels of thinking" are concepts defined by the application of level-k theory to specific games or situations and are not commonly known, and therefore less likely to be discussed, outside the behavioral economics literature. Consequently, they are much less likely to be part of the models' pre-training corpus. This distinction provides us with the opportunity to additionally investigate the model's behaviour in a task that potentially consists of more learning-based subtasks compared to the classification task of "promises".

7.1. Intra-team communication

Messages in the level-k I and II datasets are generated by the intra-team communication protocol that was introduced in Burchardi and Penczynski (2014) and classified according to the level-k of strategic reasoning. Teams of two subjects play as one entity and exchange arguments as follows. Both subjects individually make a suggested decision and write up a justifying message. Upon completion, this information is exchanged simultaneously and both subjects can enter individually a final decision. The computer draws randomly one final decision to be the team's action in the game. The protocol has the advantage of recording the arguments of the individual player at the time of the decision making. Furthermore, the subject has incentives to convince his team partner of his reasoning as the partner determines the team action with 50%

chance.

7.2. Game and Data

In Çelebi and Penczynski (2023), we propose a level-k model of strategic thinking in jury voting (JV) games à la Feddersen and Pesendorfer (1998) and Guarnaschelli et al. (2000). In juries of size 3 or 6, jurors receive informative signals (red or blue balls) and then vote to acquit (blue) or convict (red) the defendant, with the jury decision ideally matching the innocence (blue urn) or guilt (red urn) of the defendant. Looking at juries under the unanimity rule for conviction, we show that the jury performance depends on the strategic sophistication of jury members, which in turn depends on the complexity of the task at hand.

Our model assumes non-strategic, random level-0 play, to which a level-1 player best-responds by always voting according to the received informative signal. Given the unanimity rule, the best response to informative voting by level-2 players is to strategically vote always "convict" to make a conviction more likely and rely on other voters to acquit. For level-3 players, the best-response to always convicting is to play informatively like level-1 players do.

The messages are independently classified according to this level-k model by two RAs. The RAs are introduced to the level-k model and received detailed instructions about characteristics of the individual level-k types.

The classification procedure starts with both RAs providing independent sets of classifications. Then, both are anonymously informed about the classifications of the other RA and have the possibility to simultaneously revise their own classification. This revision process is repeated twice. After the process, the two RAs agreed on 93.2% (493) of the classifications.

As can be observed in Figure 11, the distribution of levels is non-degenerate and features a heterogeneity of types, a hump-shape with mode behaviour at level-1, and hardly any level-3 behaviour. This is a standard distribution commonly observed in similar studies and hence represents the type of distribution to be expected when researchers consider to classify such data (Camerer et al., 2004; Costa-Gomes and Crawford, 2006; Burchardi and Penczynski, 2014; Crawford et al., 2013). Their agreed classifications for 493 messages constitute the benchmark for the LLMs in this study.

	L0	L1	L2	L3
f_L	.21	.49	.29	.01

Table 11: Level Distribution

7.3. Prompts

Without the provision of specific characteristics of different levels of thinking, it is unclear how GPT could interpret them or distinguish, say, a level-1 thinker from a level-2 thinker. Given this uncertainty, we do not consider a basic prompt B for this classification task. Instead, we investigate two prompts, O and O_+, which closely follow the original instructions and vary only in how they provide the context for the task.

The original codebook's classification instructions for human annotators begin with a "General Comments" subsection that notes the potential implicit nature of messages and instructs annotators to classify messages at the level they believe is most likely when uncertain. This section is followed by subsections for each level from 0 to 3, each further containing "Characteristics", "Examples" and "Note" subsections. The "Characteristics" subsection outlines the observed traits of that level of thinking within the voting game, while the "Note" subsection provides additional guidance for handling ambiguous cases. The "Examples" subsections vary in number, with five examples for level-0, three for level-1, eight for level-2, and three for level-3. There are no specific comments provided for individual examples.

These classification instructions are incorporated verbatim into both prompts O and O_+, preserving the structure and format of the original level subsections and their respective "Characteristics", "Examples" and "Note" subsections. The only exception is the "Note" section of the level-0 subsection. In the codebook, this "Note" section instructs annotators to leave a message unclassified if it contains ambiguous phrases like "Play red, trust me!" that likely indicate level-1 thinking but lack additional context to confirm it. Given that the messages considered for classification are those where both annotators found enough information to classify, this specific instruction is redundant for the models and therefore omitted in the prompts. Additionally, the instruction in the "General Comments" section to classify messages based on the most likely level when unsure contradicts this instruction to leave messages unclassified when there is not enough context. Hence, to maintain instructional consistency with our prompts, we adhered to the "most likely level" instruction. Lastly, this modification also avoids the risk of the model misinterpreting the option to leave messages unclassified and incor-

rectly generalising it to other unrelated but potentially ambiguous messages and other levels of thinking.

Prior to the classification instruction section, the codebook includes a three-page explanation of the experiment, general level-k theory, and the application of level-k theory to the voting game. We have omitted these sections and instead included the most essential information in the "Context" section of the prompts. This section provides details on general game mechanics and communication protocols, aimed at providing the model with necessary context and textual patterns for its classification task. Compared to prompt O, prompt O_+ offers more detailed information on both game mechanics and communication protocols, and also includes a brief level-k theory section specific to the voting game. Additionally, in prompt O_+, these three sections (game mechanics, communication protocol, and theory) are distinctly separated by subtitles.

All other sections of the prompts are identical. In the "General Task" section, the model is instructed to classify a player's level of strategic thinking in a voting game. This is followed by the "Role Persona" section, where the model is instructed to act as a behavioural economist specialised in level-k modelling, strategic thinking, and text classification. Other auxiliary sections, such as "Classification Process", "Constraint", and "Output Format", follow the format presented in Section 4.1, and are almost identical to prompts from previous sections (see Sections C.1 and C.2 for the original codebook and the prompts).

7.4. Examples

Figure 12: Level-k I - Example Message

```
I believe we have the correct color There is no specific reason behind it If you
think that the danger to decide the whole vote for blue is to high, I will go
with red
```

The classification of the message shown in Figure 12 is detailed in Table 12, which includes the results from RAs and GPT models using n-shot and CoT prompting techniques for prompts O and O_+.

The numbers in Table 12 indicate the level of strategic thinking attributed to the writer of the message. GPT-4's classifications are consistent across both prompts. Except for GPT-3.5 under prompt O_+, all other model classifications differ from those made by the RAs. This highlights a significant divergence in the assessment of strategic thinking between the models and RAs.

56

	GPT-3.5	GPT-4
O	0	2
O_+	1	2
RAs	1	

Table 12: Human and GPT Classifications of message in Figure 12 using n-shot and CoT prompting

The rationales provided by GPT-3.5 and GPT-4 for O and O_+ using n-shot and CoT prompting techniques are displayed in Figure 13. The rationales from GPT-3.5 are significantly shorter. Specifically, under prompt O, GPT-3.5's rationale focuses solely on the phrase "there is no specific reason behind it" and concludes with a level-0 classification. In contrast, under O_+, while GPT-3.5 acknowledges additional statements from the message such as "we have the correct color" and "danger to decide the whole vote for blue", it fails to provide any explicit or detailed interpretation of these statements. Moreover, GPT-3.5 does not articulate in either rationale why the message should be classified in a certain way; such details are only revealed in the classification section of its output.

GPT-4's rationales under both prompts are significantly more detailed and thorough in analysing the message compared to GPT-3.5. As it processes each statement, GPT-4 evaluates every distinct phrase to determine its relevance for or against a specific level of strategic reasoning. For instance, the phrase "no specific reason behind it" initially suggests a possible classification as level-0. However, GPT-4 rejects this initial assessment, recognising that the phrase "the danger to decide the whole vote for blue" indicates the player is considering the impact of her actions on the group's outcome. Consequently, the model deliberates whether the player is operating at a level-1 or level-2 of reasoning and ultimately classifies her as the latter. To ensure a comprehensive analysis, GPT-4 also explores and dismisses the possibility that the player could be a level-3 thinker.

It is important to note that in neither prompt are there specific instructions to check each potential level of reasoning as a candidate for classification and to disregard them if they do not fit its provided descriptions. This approach of systematically evaluating each level is an emergent strategy that GPT-4 adopts when CoT prompting is used.

Human annotators are instructed to classify a player as a level-2 strategic thinker if "(The) player offers an argument acknowledging the potential votes of the other teams and how to vote accordingly" (see Appendix C.1 for the full instructions). These instructions are provided verbatim in both prompts O and O_+. The phrase "the danger

Figure 13: Level-k I - Classifications with CoT and n-shot for Prompts O and O_+ for message in Figure 12

```
                                                                      GPT-3.5
Output using O
Reasoning:
The player seems to be choosing randomly without any specific reasoning.  They
mention that there is no specific reason behind their choice, indicating a lack of
strategic thinking.
Classification:
0
Output using O+
Reasoning:
The player does not provide any specific reasoning for their choice.  They mention
that they believe they have the correct color without any justification.  They also
consider the risk of deciding the whole vote for blue to be too high, indicating a
slight awareness of the potential consequences of their decision.
Classification:
1


                                                                        GPT-4
Output using O
Reasoning:
The player starts with a statement that seems to be based on intuition rather than
strategic reasoning ("I believe we have the correct color There is no specific
reason behind it").  This could suggest a level-0 approach.  However, the player
then acknowledges the potential impact of their vote on the group's decision ("If
you think that the danger to decide the whole vote for blue is to high, I will go
with red").  This shows some awareness of the voting dynamics and the strategic
implications of their choice, which is beyond level-0 reasoning.
The message does not contain a clear probability argument or a direct reference to
the player's own signal, which would be indicative of level-1 reasoning.  Instead,
the player seems to be considering the consequences of their vote in the context of
the group's outcome, which aligns more with level-2 reasoning.  They are aware that
their vote could be pivotal ("the danger to decide the whole vote for blue is too
high") and are willing to adjust their vote accordingly ("I will go with red").  This
suggests they are considering the strategies of other teams, assuming that other
teams might play their signals, and are trying to best respond to that.
The message lacks the complexity of a level-3 player, who would assume that other
players are also thinking at level-2 and would adjust their strategy even further
based on that assumption.
Classification:
2
Output using O+
Reasoning:
The player starts with a statement that seems to be based on intuition or a hunch
("I believe we have the correct color") without providing a strategic justification.
This could hint at a level-0 approach.  However, the player then acknowledges the
potential impact of their vote on the group's decision ("the danger to decide the
whole vote for blue is too high"), which suggests an awareness of the voting dynamics
and the strategic implications of their choice.  This awareness moves the reasoning
beyond level-0.  The player also seems to be considering the risk associated with
voting blue and is willing to adjust their vote to red if their teammate perceives it
as too risky.  This indicates a level of strategic thinking that takes into account
the potential actions of others, aligning more with level-2 reasoning.  The player
is not simply following their own signal (which would be level-1) but is considering
the group dynamics and the possibility that their vote could be pivotal.  They are
not, however, explicitly stating that they believe others will vote contrary to their
signal, which would be indicative of level-3 reasoning.
Classification:
2
```

to decide the whole vote for blue" implicitly indicates an acknowledgement of other teams voting red and the effect of voting blue in such a scenario. While this statement

was not considered explicit enough by RAs to classify the player as a level-2 thinker, GPT-4 does classify it as such. This highlights a difference in the interpretation thresholds between human annotators and GPT.

This case also illustrates the challenges of classification and the difficulties in precisely measuring the modelâs classification performance, due to the inherent ambiguities of natural language. Although there is a mildly convincing argument for the player to be classified as level-2, GPT-4's classifications are considered incorrect. Furthermore, this type of ambiguity is prevalent in numerous other messages. The messages in Figure 14 provide two additional examples where the possibility of level-2 strategic thinking is even more subtle, if present at all. The operative words "harmless" in the first example and "safe" in the second are evaluated by the models as subtle indications that the player is considering other players' behaviour which in turn to some degree affect her action. As a result, for the first message, both GPT-3.5 and GPT-4 under both prompts classified it as level-2, while the RAs classified it as level-1; and for the second message, while GPT-3.5 and RAs classified it as level-0, GPT-4 under both prompts classified it as level-2.

Figure 14: Level-k I - Ambiguous Messages

```
Example 1:
red is more probable and almost harmless for the outcome.

Example 2:
no clue, red is safe.
```

Çelebi and Penczynski (2023) analyse four different treatments within subjects, where each subject plays two rounds. Out of 493 instances, subjects referred to a message from the previous round in nine cases, either explicitly or implicitly. This occurred despite subjects being randomly matched with new teammates in every round and being informed of this arrangement. RAs are instructed to use information from previously classified messages to classify any subsequent message that refers to these earlier messages and to make a note of it if they have done so.

For a more thorough investigation of GPT's performance, GPT is also provided with the previously referred messages for these nine instances to ensure it has full context. This step occurred after the initial classification of all the messages are independently made. As a result, GPT initially provided a classification for each of these messages without the context of the referred messages. In Figure 15, an example of such a message is shown. This message refers to the message displayed in Figure 16.

Both messages are provided by the subject within the same treatment in two consecutive rounds. The referred message in Figure 16 is classified by both models and

Figure 15: Level-k I - Example Message 3

```
Again I think we should go with red.
```

by RAs as level-2. Using this information, the RAs also classified the message in Figure 15 as level 2. Without being provided with the additional context of the previous message, GPT-3.5, under both prompts, classified the message in Figure 15 as level-0. Surprisingly, GPT-4 under prompt O classified this message as level-1, and as level-2 under O_+.

Figure 16: Level-k I - Referred message

```
I think we should stick to read.  The chances that everyone picks the red ball
are really small so we are save with read I hope
```

The rationale for both models' classifications is presented in Figure 69. A detailed analysis of the rationale under GPT-4 shows that the word "again" results in the model to consider whether the player votes strategically or informatively, given her repeated choice of red. Under prompt O, GPT-4 rules out the possibility that the player is a level-0 or level-3 thinker and acknowledges that classifying her as a level-2 thinker is speculative, leading to a level-1 classification. Conversely, under prompt O_+, the model is more inclined to assign a level-2 classification, albeit with an explicitly stated degree of uncertainty.

It is important to note that the original instructions include the directive: "If you are unsure of the level of the message, you should indicate the level you think is more likely." This is also provided verbatim to the GPT models. Consequently, even when uncertain, the model is compelled to make a classification. We have also explored a more relaxed approach for this specific message, where the model is given the option to abstain from classifying in cases of uncertainty. Under the option to abstain, the model consistently chooses not to classify the message, citing a lack of context as the reason.

In Figure 18, the rationale for classifying the message in Figure 15 is shown once the previously referred message from Figure 16 is provided to GPT-4 under prompt O[25]. This demonstrates that when the model is provided with the referred message, it can effectively incorporate it into its classification analysis and accordingly classify the message as level-2.

---

[25]Note that only the referred message, and not its prior classification as level-2, is provided as additional context. Consequently, as can be observed in the rational in Figure 15, the model first classifies this referred message as level-2, then uses this classification to inform its assessment of the message in question, and then, also classifies it as level-2.

Figure 17: Level-k I - Classifications of the message in Figure 15 with CoT and n-shot
for Prompts O and O_+

```
                                                                    GPT-3.5
Output using O
Reasoning:
The player simply states their preference for the red option without providing any
reasoning or justification for their choice.  There is no indication of any strategic
thinking or consideration of other players' decisions.
Classification:
0
Output using O+
Reasoning:
The player simply states their preference for the color red without providing any
reasoning or justification for their choice.  There is no indication of any strategic
thinking or consideration of other teams' potential votes.
Classification:
0


                                                                       GPT-4
Output using O
Reasoning:
The message "Again I think we should go with red" suggests that the player is
following a consistent strategy of voting red.  However, the message lacks any
explicit reasoning or justification for this choice.  It does not mention the color
of the ball they picked, nor does it consider the potential voting behavior of other
teams.  The use of the word "again" implies that this is a repeated strategy, but
without further context or explanation, it is not clear whether this is based on a
strategic consideration of others' likely actions (Level-2) or simply a personal rule
or preference.
Given the lack of strategic reasoning evident in the message, it is not possible to
classify this player as Level-3.  The message also does not contain any probability
argument or acknowledgment of the signal, which would be characteristic of a Level-1
player.  It does not appear to be a random or unrelated choice, which would be
indicative of a Level-0 player.
The message could potentially be interpreted as a Level-2 strategy if we assume that
the player has previously considered the strategies of others and has settled on
always voting red as a best response.  However, without explicit acknowledgment of
others' strategies, this interpretation is speculative.
Classification:
1
Output using O+
Reasoning:
The statement "Again I think we should go with red" suggests that the player is
following a consistent strategy of voting red.  However, the reasoning behind this
choice is not explicitly stated.  The use of the word "again" implies that this
strategy has been used before, possibly indicating a pattern or rule the player is
following.  There is no mention of the color of the ball they drew, nor is there
any consideration of the other teams' potential votes.  This lack of strategic
consideration for others' actions could indicate a level-1 player who consistently
follows their own signal, but the message is too vague to confirm that.  It could
also be a level-2 player who has adopted a strategy of always voting red, assuming
that other players will vote according to their own signal.  Without further
information, it is difficult to determine if this is a level-1 or level-2 strategy.
The classification leans towards level-2 because the player seems to have a
predetermined strategy (voting red) that they are sticking to, which could be a
response to the assumption that other players are voting according to their own
signals (level-1 behavior).  However, due to the lack of explicit reasoning, this
classification is made with some uncertainty.
Classification:
2
```

Although GPT-4's classification capabilities are impressive, there are still instances where it commits obvious blunders. Consider the message in Figure 19, written by a

Figure 18: Level-k I - Classifications of the message in Figure 14 with CoT and n-shot for Prompt O after the refered message is also provided.

```
                                                                    GPT-4
Output using O
Reasoning:
The message "Again I think we should go with red." is brief and lacks explicit
strategic reasoning.  However, the use of the word "again" suggests that this
decision is based on a previously established reasoning pattern.  Given the context
of the previous message, which contained level-2 reasoning, it is likely that the
player is continuing with the same line of thought.  The player is choosing red,
potentially because they believe other players will vote their signal, and red is a
safer strategic choice.  Despite the brevity of the message, the implication is that
the player is still considering the actions of others, which is characteristic of
level-2 reasoning.
Classification:
2
```

subject who is attempting to randomise her response by coordinating with her teammate. The message clearly indicates that the subject is a level-0 thinker. Yet, GPT-3.5 classifies this message as level-2, and GPT-4 classifies it as level-1 under both prompts (with CoT and n-shot).

Both models misinterpret the attempt to coordinate with a teammate as a strategic play, and consequently, they erroneously argue for the player to be classified as either level-1 or level-2 (see Appendix C.3, Figure 73 for the rationales for the message in Figure 19 and other additional examples). Although both models accurately understand what constitutes a teammate, since they perceive this message as reflecting a basic understanding of the game mechanics, rather than a random play or an action unrelated to the task, they eliminate the possibility that the subject is level-0. This suggests that the model still needs this specific type of instance to be provided as an example or described as part of the characteristics of level-0 play in order to classify it correctly[26]. Furthermore, it illustrates that when the model is unable to classify a message under any specific category, it considers other types of behaviours that can be considered strategic to determine the player's level of thinking –even if such behaviours, like attempting to coordinate with a teammate, might not truly reflect sophisticated strategic behaviour.

Figure 19: Level-k I - Example Message 4

```
i think its probably better if we dont pick the same colour, so if i pick blue
you should pick red
```

---

[26]For instance an additional instructions such as "Refrain from considering any argument that involves the behaviour of the player's teammate as strategic"

7.5. Results

		no-CoT		CoT	
		0-Shot	*n-Shot*	*0-Shot*	*n-Shot*
GPT-3.5	O	70.2	64.5	64.1	64.5
	O_+	68.8	75.2	66.1	70
GPT-4	O	79.5	84.4	79.5	**91.3**
	O_+	76.9	79.9	76.3	89.5

Table 13: Overall Accuracy of Level Classification

For GPT-3.5, n-shot O underperforms compared to 0-shot O in no-CoT treatment but slightly outperforms its 0-shot counterpart in CoT treatment. N-shot prompting improves GPT-3.5's performance for both prompts only when CoT prompting is present. However, CoT prompting generally worsens the performance of GPT-3.5, except for n-shot O. For GPT-3.5, the performance under O_+ is better than under O in most cases.

For GPT-4, n-shot prompting consistently improves the model's performance. While in 0-shot treatments, CoT prompting has little to no effect, it significantly boosts the performance of GPT-4 in n-shot treatments. Overall, CoT prompting in conjunction with n-shot prompting achieves the best performance for either prompt. Furthermore, GPT-4's performance under O is consistently better than its performance under O_+ for all treatments.

GPT-4 consistently outperforms GPT-3.5. While GPT-3.5's performance benefits from additional context and level-k theory emphasis, GPT-4's performance declines under the same conditions. This highlights a distinct difference in how the two models process and benefit from additional context. The best performing result is achieved with GPT-4 using CoT and n-shot prompting techniques for prompt O, with an accuracy of 91.3%.

8. Level-k II: Asymmetric-Payoff Coordination Games

8.1. Game and Data

Finally, van Elten and Penczynski (2020) study asymmetric-payoff coordination games (APC) introduced by Crawford et al. (2008, henceforth CGR) and provide textual data

X-Y games (CGR notation)	a	π_1, π_2	Pie games (CGR notation)	a	π_1, π_2
Symmetric Payoffs (**SL**)	X	$5, 5$	Symmetric Payoffs (**S1**)	L (\$)	$5, 5$
	Y	$5, 5$		R (#)	$5, 5$
				B (§)	$5, 5$
Slight Asymmetry (ASL)	X	$5, 5.1$	Symmetric Payoffs (**S2**)	L (\$)	$6, 6$
	Y	$5.1, 5$		R (#)	$6, 6$
				B (§)	$5, 5$
Moderate Asymmetry (AML)	X	$5, 6$	Moderate Asymmetry (AM2)	L (\$)	$5, 6$
	Y	$6, 5$		R (#)	$6, 5$
				B (§)	$6, 5$
Large Asymmetry (ALL)	X	$5, 10$	Moderate Asymmetry (AM4)	L (\$)	$6, 7$
	Y	$10, 5$		R (#)	$7, 6$
				B (§)	$7, 5$

Table 14: Payoff structure of coordination games.

supporting the result that the incidence of level-k reasoning is low in symmetric, pure coordination games and high in asymmetric, "battle of the sexes"-type coordination games. The dataset is of particular interest, because its text analysis involves the classification of non-trivial level-0 beliefs.

Table 14 describes the four X-Y games and four Pie games. In contrast to payoff-symmetric games (in bold), payoff-asymmetric games feature a higher coordination payoff π for one of the two players, depending on the action on which they coordinate. The miscoordination payoff is 0 for both players. The choice is between letters X and Y in the X-Y games and between 3 pie slices (L, R, B) which are identified by (\$, #, §) and of which B is uniquely white.

The dataset consists of 851 messages gathered through intra-team communication as described in Section 7.1. All messages are in German. The benchmark classifications are derived from the agreed assessments of two RAs[27]. The RAs provide a lower bound and an upper bound for the level of reasoning in each message. They also identify whether any label or payoff salience argument is present, and if so, classify the type of salience.

Tables 15a and 15b show the distributions of the lower and upper bound levels where L_n indicates the level-n strategic thinking. Tables 16a and 16b show the distributions of the benchmark classifications for label and payoff salience. In these tables, "$\sim$" indicates indifference to salience or payoff salience, "no" signifies that there is no mention of a payoff or label salience in the text, H and L denote high and low payoff

---

[27]See van Elten and Penczynski (2020) for details

salience respectively, and §, #, $, X and Y represent the label salience for the game choice with the same tag.

	L_0	L_1	L_2	L_3	L_4	L_5		L_0	L_1	L_2	L_3	L_4	L_5
f	.504	.334	.143	.016	.002	–		.341	.37	.234	.042	.006	.007
			(a) Lower Bound							(b) Upper Bound			

Table 15: Level distributions of the benchmark

Like the RAs, we instruct the GPT models to classify lower and upper bounds of the messages' levels of reasoning as well as the label- and payoff-salience of the level-0 belief. The results from the supervised machine learning approach in Penczynski (2019) will provide an additional computerised benchmark.

	§	#	$\sim$	$	X	Y	no		$\sim$	H	L	no
f	.159	.025	.024	.107	.103	.002	.581		.722	.228	.039	.008
			(a) Label salience							(b) Payoff salience		

Table 16: Distributions for label and payoff salience classifications of the benchmark

Each participants goes through all eight games prior to any communication exchange. We dropped those messages in which subjects refer to a reasoning that they laid out in an earlier round, as we did not want the LLM to get into assigning reasoning from earlier messages as the RAs did. In total 104 messages (12.2% of the data) are dropped.

8.2. Prompts

No basic prompt B is considered due to the same reasons outlined in Section 7. Instead, two prompts, O and O_+, which vary in their degree of reflecting the original classification instructions tailored for human annotators, are considered. These two prompts differ in their "Context" and "Classification Task" sections.

The original codebook is the most detailed and lengthy codebook we consider in this study (see Appendix D.1). Given the non-negligible effect of the degree of information in the "Context" section on model performance observed in Section 7, we further investigated this effect with the current game by varying the level of information provided among prompts O and O_+.

"Context" sections of both prompts begin with an identical short subsection on game mechanics, followed by a subsection on the two coordination games played in the

experiments where each game type (Pie and X-Y) is briefly introduced, and payoff tables for the variations of each game type is provided in a similar table format as in Table 14 in Markdown format. The two prompts significantly diverge in terms of the details each provides in the subsequent theory subsection. Prompt O closely follows the original instructions, whereas prompt O_+ provides a summary consisting of single sentence descriptions for each of the fundamental concepts detailed in the original codebook.

The "Context" section in prompt O outlines the theory in a series of subsections. The first subsection, "Salience in Decisions", describes the concept of salience in the games. This is followed by a "Level-k Model" subsection, which details level-0 distribution, level-0 belief, and population belief concepts, and describes the characteristics of each level from 0 to 4. Descriptions of each level's characteristics are unevenly supplemented with example messages: three examples for level-0 thinking, two for level-1, one for level-2, and no examples for higher levels. Some of these examples also entails cases for different types of salience. Nont of the level-0 examples provide label or payoff salience, level-1 examples only include label salience (one for each game type), and the level-2 example feature only a payoff salience example for the X-Y game.

The "Context" section of prompt O is followed by a "Classification Task" section that separates the tasks into two subtasks: classification payoff or label salience and classification of level of strategic thinking. The classification instructions under these subsections are also taken verbatim from the original codebook, and therefore are detailed and lengthy.

Differently from the prompts considered in "Level-k I" Section (Section 7), prompt O provides information related to the classification both in the "Context" section and in the "Classification Task" section. This is a natural outcome of our choice to closely mirror the format and instructions of the original codebook. The original instructions aim to first ensure that the human annotators get a general understanding of the theory pertaining to level-k modelling then proceeds this with the application of this theory for the outlined coordination games. This is provided in the "Context" section of O. After establishing the theory and its application, the codebook presents a detailed classification instructions. This is provided in the "Classification Task" of O[28]. Therefore, prompt O takes a human-centric approach to instruct the model by anthropomorphi-

---

[28]One deviations we made in the "Classification Task" section of O is that we switched the order of the two classification tasks outlined in the original codebook to emphasise the order with which we want the GPT models to classify the message which is to first classify the level-0 belief and decision salience then to classify the level of strategic thinking.

cally assuming that the model needs to "understand" the theory before "understanding" how to classify messages based on this theory.

The original classification instructions, and therefore prompt O, include only five examples. These examples provide very limited coverage given the array of potential examples that can be generated by combining different levels of thinking with payoff or label salience, and with the two types of coordination games (X-Y and Pie Games). Furthermore, three out of the five examples that involve either payoff or label salience lack information indicating the appropriate classification for these categories. Consequently, it is debatable whether prompt O should be classified as a 5-shot prompt, given that the examples do not conform to the typical characteristics of a standard n-shot prompt. Such prompts should include a variety of examples for each possible category and clearly indicate the classification of each example. Instead, it would be more appropriate to consider these examples as supplementary instructional components rather than as explicit demonstrations for the classification task. Hence, given the nature of these examples, we classify prompt O as a 0-shot prompt.

In prompt O_+, we diverge significantly from the detailed theory section provided in prompt O by providing a very brief section on the level-k theory and decision salience. We provide as detailed information on level-0 thinking as we have done in prompt O, followed by a brief level-1 thinking description. We omit explicit description for each of the higher levels of thinking, and instead provide a brief generalised characterisation of a level-k thinker for $k > 1$. Moreover, in its "Classification Task" section, we only provide two lines of instructions that simply instruct the model to classify the payoff or label salience, and lower and upper bound of level of thinking. Additionally, we omit any examples in either of these sections. In other words, in prompt O_+, we do not concern ourselves too much with whether the model fully "understands" the concepts by considering a significantly brief way of presenting the background information and the classification instructions. Hence, prompt O_+ is similar in nature to the basic prompts we have considered in Sections 5 and 6: it provides minimal contextual information and places more weight on the model's existing knowledge on the topic by omitting detailed information on how to perform the task.

O_+ without any demonstrations establishes a 0-shot baseline for the prompt, and enables us to investigate the effect of incorporating demonstrations (n-shot prompting) in a modular fashion. We do so by developing an "Examples" section where we attempt to cover a broad array of possible case with a total of 16 demonstrations, four for each level from 0 to 3. None of the demonstrations are from the dataset, and generated by us. All examples are generated with the aim to explicitly depict their respective level

of thinking. For instance, level-2 messages contain variations of the pattern "They think that I do <X>", and level-3 messages explicitly state variations of "They think that I think that they will do <X>". In level-0 examples, two provides examples of label salience, one provides payoff salience, and one provides neither label nor payoff salience. Level-1 examples consist of three cases of label salience and one case of payoff salience. For levels 2 and 3, two cases for each type of salience are provided. Three of the level-0 examples are for the X-Y game, and one for for the Pie game. Two of the level-1 examples are for the X-Y game, one for the Pie game, and one that can be applicable to either game. For levels 2 and 3, one example is for each game and two examples can be applicable to either game. Each example is followed by information on level-0 belief, label salience, payoff salience, and level classification.

The other sections of both prompts are identical ("General Task", "Role Persona", etc.). Except for the "Input Format" section, the other sections are as defined in Section 4.1 and can be examined in more detail in Appendix D.2. The "Input Format" section is an additional part present only for the current prompts. Human annotators receive contextual information about the game, the team, and the subject's initial decisions, which are essential for understanding the message. The exact game being played, as shown in Figure 14, and the team of the player are necessary for both human annotators and GPT models to effectively grasp the context of the message. Consequently, unlike in our previous classification experiments, GPT is provided not only with the subject's message but also with information about subject's team, the game she is referring to, and her initial decisions. The "Input Format" section provides the template for the input that the model receives. By including this section in the prompt, we aim to help the model better orient itself when provided with the defined input string. This metric is also referred to in the literature as the Intersection over Union (IoU) metric or the Jaccard Index (Müller et al., 2022).

8.3. Examples

Figure 20: Level-k II - Example Message

```
Team:  1
Game:  AML
Decision:  X
Message:  They might think that we think they are selfish.
```

In Figure 20, the input provided for the models is presented. Note that although the message is in English, the actual input provided to the models is in German. The subject, part of team 1, plays the AML variation of the X-Y where she will receive

	GPT-3.5				GPT-4			
	L_B	U_B	S_L	S_P	L_B	U_B	S_L	S_P
O	1	2	*no*	*no*	1	2	*no*	H
O_+	1	2	*no*	*no*	3	3	*no*	H
RAs	3	3	*no*	H	3	3	*no*	H

Table 17: Human and GPT Classifications of message in Figure 20 using CoT prompting for n-shot O_+ and 0-shot O

a payoff of 6 if the teams coordinate on choice "Y" and 5 if the teams coordinate on choice "X".

Table 17 presents the classification results from RAs and GPT models utilising CoT prompting technique for 0-shot O and n-shot O_+ where the columns L_B, U_B, S_L, and S_P represent the lower bound of strategic thinking, upper bound of strategic thinking, label-salience, and payoff salience, respectively. Except for GPT-4 under prompt O_+, all model classifications are identical and differ from those made by the RAs for the lower and upper bounds. Moreover, GPT-3.5 fails to agree with the RAs on the payoff salience classification.

In Figure 21, the rationals for both models under 0-shot O and n-shot O_+ prompts are displayed. Their consecutive classification output is omitted to fit the figure within a single page. The reasoning under both models correlate with the length of the prompts: for relatively longer prompt O, a longer reasoning is generated by the models. The correlation between the length of the prompts and the length of reasoning produced by the models suggests that LLMs might be relying on a straightforward strategy of mirroring the extent of textual input provided in the prompts to generate what appears to be thoughtful reasoning. This pattern indicates that the models may primarily be extending or "echoing" the prompt's text rather than engaging in genuine analytical thinking based on its understanding of the concepts provided in the prompt (Bender et al., 2021).

GPT-3.5 provides significantly shorter rationales compared to GPT-4 under both prompts. Under O, GPT-3.5 briefly addresses both strategic thinking and decision salience, while under O_+, it only offers a rationale for strategic thinking. Although the model acknowledges strategic thinking and correctly notes that the subject considers the other team's perspective, the rationales lack specific justification for classifying the lower bound for her level of thinking as "1" and the upper bound of her level of thinking as "2".

GPT-4 provides a significantly more detailed analysis of the message under both

69

Figure 21: Level-k II - Generated rationals for 0-shot O and n-shot O_+ under CoT for message in Figure 20

Output using O
Reasoning:
The player's message seems to suggest that the other team might think that his team considers them to be selfish. This indicates a level of strategic thinking as the player is considering the other team's potential thoughts and actions. However, the message does not explicitly indicate any salience or strategic reasoning based on the game's framing.
Output using O_+
Reasoning:
The player seems to be considering the other team's perspective and how they might perceive the decision-making process.

Output using O
Reasoning:
The message "sie werden bestimmt denken, wir würden über sie denken, sie wären egoistisch" translates to "they will certainly think that we would think they are selfish." This statement suggests a level of strategic thinking that involves anticipating the other team's thought process about one's own team's perceptions. The player is considering that the other team might expect their own team to assume that they (the other team) will act in their own self-interest, which is a characteristic of higher-level strategic reasoning.
The message does not mention any preference for the label of the decision (X or Y) based on its appearance or position. Therefore, the player does not exhibit label salience.
The message implies that the other team might be expected to act selfishly, which suggests that the player is considering the payoffs. However, the message does not explicitly state a preference for higher or lower payoffs for their own team. It is more focused on the belief about the other team's potential selfishness. Therefore, the player exhibits a level-0 belief about the other team's preference for higher payoff.
- Lower Bound: The player is engaging in strategic thinking beyond level-0, as they are considering the other team's potential beliefs about their own team's perceptions. This indicates at least level-1 reasoning.
- Upper Bound: The player's message suggests they are considering what the other team thinks about their own team's beliefs, which is indicative of level-2 reasoning. However, there is no clear indication that the player is considering higher levels of reasoning, such as level-3 or above, where they would be best responding to the other team's level-2 reasoning.
Output using O_+
Reasoning:
The message "sie werden bestimmt denken, wir würden über sie denken, sie wären egoistisch" translates to "they will certainly think that we would think about them, they are selfish." This message suggests a level of reasoning about what the other team will think about Team 1's beliefs regarding the other team's preferences. The message implies that the other team might believe that Team 1 thinks the other team is selfish, which could mean that they would choose the option that gives them a higher payoff.
In the context of the AML game, where the payoffs are as follows:
- X: 5 Taler for Team 1, 6 Taler for Team 2
- Y: 6 Taler for Team 1, 5 Taler for Team 2
The message does not directly reference a preference for either label (X or Y) or a specific payoff amount. However, it does imply a belief about the other team's potential selfishness, which is related to payoff salience. The message suggests that the other team may choose the option that gives them a higher payoff (Y for Team 2), and Team 1 should consider this in their decision-making.
The message indicates a higher level of reasoning, as it involves thinking about what the other team believes Team 1 thinks about them. This is indicative of at least level-3 reasoning, as it involves multiple layers of belief about the other team's thought process.

70

prompts. It begins its rational by re-iterating the original message and translating it to English. This is a commonly observed behaviour in many other instances, yet there are also instances where the rational does not contain an English translation for the message (See Appendix D.3 for additional examples). Under both prompts, it begins its rational by correctly identifying that the subject is considering the belief of the other team on the subject's belief about the other team. Then without providing a classification for her level of thinking, it proceeds into the analysis of the subject's payoff salience and level-0 belief. Under both prompts, it notes the absence of clear payoff salience but argues that the mention of the other team's "selfishness" indicates subject's belief about the other team's preference for a higher payoff, but only under prompt O, the model explicitly uses the term "level-0 belief".

Under prompt O_+, GPT-4 correctly identifies the pattern "They think that we think that they are selfish" with statement "The message (...) involves thinking about what the other team believes Team 1 thinks about them" and correctly classifies the player's level of thinking. In contrast, under prompt O, although GPT-4 makes the statements "The statement involves (...) anticipating the other team's thought process about one's own team's perceptions" and "The player is considering that the other team might expect their own team to assume that the other team will act in their own self-interest," it falsely classifies the player's lower bound of strategic thinking as "1" and the upper bound as "2".

Despite the analogous arguments concerning levels of thinking between the prompts, the misclassification of the model under prompt O supports the "Stochastic Parrots" view of LLMs as articulated by Bender et al. (2021). According to this view, LLMs primarily mimic language based on patterns and correlations learned during training. If GPT-4 truly "understood" the rationale it provided, it would have also classified the subject's level of thinking as "3" under prompt O as it did under prompt O_+. Alternatively, challenging the 'Stochastic Parrot' interpretation, perhaps GPT-4 under prompt O correctly "understood" its rationale but exhibited either an imprecise grasp of the distinction among higher levels of thinking, or made a computational error in counting the number of iterative best responses detailed in its rationale.

Figure 22: Level-k II - Example Message 2

```
Team:   2
Game:   AM4
Decision:  #
Message:  I would take the second one (7.6) here because it is fairer than (7.5).
```

In Figure 22, the subject is playing AM4 variation of the Pie game and argues for the

	GPT-3.5				GPT-4			
	L_B	U_B	S_L	S_P	L_B	U_B	S_L	S_P
O	1	1	no	H	1	1	no	H
O_+	1	1	no	H	1	1	no	H
RAs	0	0	no	no	0	0	no	no

Table 18: Human and GPT Classifications of message in Figure 22 using CoT prompting for n-shot O_+ and 0-shot O

"fair" choice of alternative # that offers a payoff of 6 to the subject over alternative §, which provides a payoff of 5, while under both alternatives a payoff of 7 is provided for the other team players. The unmentioned alternative $ offers a higher payoff of 7 for the subject's team and a lower payoff of 6 for the other team. The subject is classified as a level-0 thinker by RAs but as a level-1 thinker by both models under both prompts. Furthermore, while the RAs did not identify any label or payoff salience, both models under both prompts classified the message as demonstrating high payoff salience.

Figure 23 provides the rationales for both models under both prompts. All rationales argue that "fairness" should be seen as a type of payoff salience. GPT-4 under prompt O states that the subject's preference for a payoff of 6 over 5 shows a preference for a higher payoff under "the context of fairness". Similarly, GPT-4 under prompt O_+ identifies the message as a "nuanced form of payoff salience". Classifying subject as displaying a higher payoff salience is the initial and crucial point where the RAs and the models differ. Because RAs did not identify any type of salience and therefore any type of level-0 belief, they have consequently classified the player as a level-0 thinker. On the other hand, because the model identifies fairness as a payoff salience and therefore establishes a level-0 belief for the subject, it classifies the subject as a level-1 thinker by interpreting the mention of fairness as an implicit acknowledgement of considering the other team's beliefs.

While attempting to argue that "fairness" implies level-1 thinking, GPT-4 under prompt O_+ makes contradictory statements. Initially, it recognizes the playerâs consideration of payoff fairness and asserts, "A level-0 player would not consider fairness or the other team's payoff". Subsequently, however, it remarks, "The player's message does not explicitly indicate that they are considering what the other team might think or do, which would be characteristic of higher levels of reasoning (level-1 or above)." Yet despite this apparent contradiction, it concludes "Therefore, the message suggests a lower bound of level-1 reasoning, as the player is best responding to a belief about the other teamâs potential preference for fairness". On one hand, these conflicting state-

Figure 23: Level-k II - Generated rationals for 0-shot O and n-shot O_+ under CoT for message in Figure 22

Output using O
Reasoning:
The player suggests choosing the second option because it is fairer than the first option. This reasoning indicates a preference for the option that provides higher payoffs, which suggests a level of strategic thinking. The player is considering the fairness of the payoffs, indicating a level of strategic reasoning.
Output using O_+
Reasoning:
The player's message indicates a preference for the second option due to its fairness compared to the first option. This suggests a form of payoff salience. The reasoning in the message seems to exhibit a level-1 understanding, as the player is considering the fairness of the payoffs and how the other team might perceive the decision.

Output using O
Reasoning:
The player from Team 2 has suggested the decision '#' in the AM4 variation of the Pie game. The AM4 game has an asymmetric payoff structure where the decision '#' yields a payoff of 7 Taler for Team 1 and 6 Taler for Team 2, while the decision '§' yields a payoff of 7 Taler for Team 1 and 5 Taler for Team 2. The player's message, when translated from German, reads "here I would take the second one (7,6) because it is fairer than (7,5)." This message indicates a concern for fairness and a preference for a more balanced distribution of payoffs between the teams.
The player's message does not explicitly mention any label salience, such as a preference for the visual distinction of the '§' slice in the Pie chart. Therefore, we can infer that the player does not exhibit label salience in this decision.
The player's message does indicate payoff salience. The player prefers the decision '#' because it offers a more balanced payoff (7,6) compared to the decision '§' (7,5), which would give a lower payoff to their own team. This suggests that the player is attracted to the action that yields a higher payoff for their own team, but within the context of fairness.
In terms of the level of strategic thinking, the player's message suggests a level-1 reasoning. The player is considering the payoffs and is choosing the decision that seems fairer, which implies a best response to a belief about the other team's action. The player does not consider higher-order beliefs about what the other team might think about their team's decision-making process. Therefore, the lower bound of reasoning is level-1, as the player is best responding to a belief about the other team's likely action based on fairness. The upper bound is also level-1, as there is no indication of higher-order strategic thinking in the message.
Output using O_+
Reasoning:
The player's message suggests choosing the option labeled '#' in the AM4 variation of the Pie Game. The player justifies this choice by stating that the payoff of (7,6) is fairer than (7,5). This indicates that the player is considering the distribution of payoffs between the two teams and prefers the option where the difference in payoffs is smaller, which is a consideration of fairness rather than simply choosing the highest individual payoff.
The player does not mention any label salience, such as preferring the symbol '#' over '$' or '§' for any reason other than the associated payoffs. Therefore, the player does not exhibit label salience.
The player's message does show payoff salience, as they are considering the fairness of the payoffs, which is related to the magnitude of the payoffs for each team. The player prefers the option where their team gets a slightly lower payoff (6) compared to the other team's payoff (7), which is the fairer distribution in their view. This suggests a preference for high payoffs but with a consideration for fairness, which is a nuanced form of payoff salience.
In terms of the level of reasoning, the player is considering the fairness of the payoffs, which suggests a level of strategic thinking beyond level-0. A level-0 player would not consider fairness or the other team's payoff. The player's message does not explicitly indicate that they are considering what the other team might think or do, which would be characteristic of higher levels of reasoning (level-1 or above). Therefore, the message suggests a lower bound of level-1 reasoning, as the player is best responding to a belief about the other team's potential preference for fairness.
The upper bound of reasoning is not clearly indicated in the message. The player does not express any anticipation of the other team's strategic thinking or multiple levels of iteration in their decision-making process. Therefore, the upper bound of reasoning is also level-1, as there is no evidence to suggest that the player is engaging in higher-level strategic thinking.

73

ments in its reasoning might once again suggest that the model does not truly understand what it's stating, aligning with the "statistical parrot" critique. On the other hand, such contradictions could indicate that the model is "thinking out loud", weighing arguments for and against the case of the subject being a level-1 thinker. Therefore, the model's contradictions may not necessarily reflect a lack of understanding but rather an exploration of uncertainty in classifying the subject's level of reasoning.

If fairness is viewed as a preference independent of individuals' beliefs, GPT's classification of this message is incorrect. However several studies argue that fairness is primarily defined by the people's belief about other's expectations (Bicchieri and Chavez, 2010; Castelli et al., 2014), and the weight placed on considering others' beliefs determines a person's degree of fairness concerns in her decision-making process (Takagishi et al., 2010). Thus, GPT's claim that fairness consideration is indicative of a higher level of reasoning has its support in the literature. But more importantly, this highlights the ambiguities inherent in natural language, complicating the exact classification of a player's level of thinking based on their message, and possibly a more appropriate classification for this message would be a lower bound of 0 and an upper bound of 1 for the subject's level of thinking[29].

8.4. Results

Given that the benchmark classifications for levels of thinking are defined as an interval, with a lower and an upper bound across six potential levels in any given instance (message, the model's task is a multi-class, multi-label classification. In this framework, a given instance can be classified as positive under multiple classes. This opens up the possibility for a classification to be partially correct. A measurement that requires an exact match with the actual interval might overlook this partial correctness and is therefore considered a harsh metric (Sorower, 2010). A more nuanced accuracy metric commonly used in multi-label classification is the ratio of correctly predicted labels to the total number of labels –both predicted and actual– averaged across all instances (Godbole and Sarawagi, 2004). This measurement is also referred to in the literature as the Intersection over Union (IoU) metric or the Jaccard Index (Müller et

---

[29]A similar argument applies to the concepts of selfishness and generosity. A subject calling herself selfish may be less likely to be assumed to consider other players' beliefs, whereas a subject calling herself generous may be more likely to be assumed to consider other players' beliefs. Interestingly, while RAs classified the lower bound of the level of thinking as 0 for these two types of a basic messages, they classified the upper bound of the level of thinking of a subject calling herself generous as 1 and the upper bound for a subject calling herself selfish as 0. A similar classification pattern is also observed with GPT-4 under prompt O_+. See Example Messages 4 and 5 in Appendix D.3 for details.

al., 2022). Although IoU can be calculated for every class and averaged (either macro or weighted), this approach has been criticised for not adequately addressing correlations among different classes. Therefore, to assess the classification performance of our models, we have chosen to employ the instance-based accuracy metric as introduced by Godbole and Sarawagi (2004).

		no-CoT		CoT	
		0-*Shot*	*n-Shot*	0-*Shot*	*n-Shot*
GPT-3.5	O	44.2	–	49.4	–
	O_+	52.3	62.6	55.3	65.3
GPT-4	O	64	–	69.2	–
	O_+	65.6	68.5	66.7	**72.8**

Table 19: Level Classification Accuracy (Jaccard)

Table 19 presents the agreement rates of models' classification with the RA classifications for each prompt under each treatment. GPT-3.5 consistently underperforms GPT-4. CoT and n-shot prompting techniques consistently improve the performance of either model. For GPT-3.5, 0-shot O_+ consistently outperforms 0-shot O. This indicates that GPT-3.5 does not benefit from few basic examples and significantly more detailed instructions as provided in prompt O. Similarly, for GPT-4, 0-shot O_+ outperforms 0-shot O in the no-CoT treatment. Yet, the opposite is true in the CoT treatment. Hence, the detailed descriptions and examples present in prompt O enhances CoT prompting's positive effect on GPT-4's performance more than CoT's effect on 0-shot O_+. However, when CoT prompting is used in conjunction with n-shot prompting, O_+ generates the best performing classification results for either model.

Penczynski (2019) classified the same text corpus using a machine learning algorithm (MLA) that employs bag-of-words features. He evaluates the performance of his model by calculating its accuracy via an exact match of the lower bound of the level of thinking between human annotators and MLA classifications. The best-performing MLA model achieves an accuracy rate of 67%. GPT-3.5 achieves a lower bound level of thinking classification accuracy of up to 65.5%. Hence, MLA model's out-of-sample performance outperforms GPT-3.5's best performing prompt. Conversely, GPT-4, except for prompt O under no-CoT treatment (66.4%), outperforms the MLA classification. Furthermore, GPT-4 under prompt n-shot O_+ with CoT achieves the highest accuracy for the lower bound of level of thinking classification with 76% (see Appendix D.4 for the complete set of classifications).

		no-CoT		CoT	
		0-*Shot*	*n-Shot*	0-*Shot*	*n-Shot*
GPT-3.5	O	34.9	–	44.1	–
	O_+	35.9	**72.5**	48.9	66.2
GPT-4	O	61.1	–	69.5	–
	O_+	58.6	71	62.7	69.4

Table 20: Payoff Salience Classification Accuracy

Table 20 presents performance of both models for each prompt for payoff salience classification accuracy. n-shot prompting consistently improves the performance of both models. While CoT prompting consistently improves both model's performance for both prompts in 0-shot treatments, it deteriorates them in n-shot treatments. Consequently, best performing classifications are observed in n-shot no-CoT treatment for both models. In no-CoT treatment, improvement enabled by n-shot prompting is much larger for GPT-3.5. This results in GPT-3.5 to outperform GPT-4 in n-shot no-CoT treatment and to generate the overall best performing classification.

GPT-3.5 consistently performs worse under prompt O which indicates that it fails to benefit from its detailed instructions, whereas GPT-4 performs better under O compared to O_+. These performance differences under prompt O becomes more distinct in the CoT treatment. This indicates that while detailed information worsens GPT-3.5's reasoning, it enhances GPT-4's for the classification of payoff salience. Moreover, for GPT-4, in CoT treatments, O slightly outperforms n-shot O_+. This suggests that GPT-4's rational and its consecutive payoff salience classification benefits equally from detailed instructions without explicit examples as from explicit demonstrations in terms of both the effectiveness of rational and the accuracy of the payoff salience classification.

		no-CoT		CoT	
		0-*Shot*	*n-Shot*	0-*Shot*	*n-Shot*
GPT-3.5	O	40.5	–	58.6	–
	O_+	41.8	47.1	59.2	67.5
GPT-4	O	62.2	–	83.9	–
	O_+	44.4	78.4	78.7	**86.2**

Table 21: Label Salience Classification Accuracy

In Table 21, the performance of both models for each prompt for label salience

classification accuracy is presented. On average, the models are observed to perform better at label salience classification than payoff salience classification. CoT prompting consistently improves both models' performance for both prompts. Similarly, n-shot prompting consistently improves both models' performance for O_+, especially for GPT-4 in no-CoT treatment.

While GPT-3.5 performs slightly better under 0-shot O_+ relative to O, GPT-4, as in payoff salience classification, benefits from the detailed instructions of prompt O and performs better under prompt O compared to under 0-shot O_+. However, contrary to both models' behaviour in payoff salience classification, their performance improves under n-shot O_+ when CoT prompting is introduced. Consequently, the highest classification performance is achieved by GPT-4 under n-shot O_+ with CoT prompting.

9. Costs and benefits

Table 22 presents both monetary and time costs to classify 100 messages in Promise I. Costs are given in terms of USD and time is shown in minutes. The monetary costs for both GPT models are based on the number of input and output tokens, with output tokens being twice as costly as input tokens. This is reflected in the monetary cost difference between the no-CoT and CoT treatments. Furthermore, the token costs for GPT-4 are approximately 20 times higher than those for GPT-3.5. This is reflected in higher costs for each prompt under GPT-4 compared to GPT-3.5.

		no-CoT		CoT	
		Money	*Time*	*Money*	*Time*
GPT-3.5	B	0.04	4.5	0.06	3.4
	O_S	0.06	2.6	0.08	6.4
	O_N	0.06	4	0.08	6.7
	O_W	0.06	2.3	0.08	6.6
GPT-4	B	0.5	1.1	0.9	10.9
	O_S	0.7	1.4	1	11.5
	O_N	0.6	1.5	1	12.2
	O_W	0.6	1.3	1	13.2

Table 22: Promise I: Money and Time costs per 100 messages in USD and minutes.

Given GPT-4's significantly larger parameter count, its time to generate the first token and time per output token are expected to be longer than GPT-3.5's (Talamadupula, 2024). Yet, for no-CoT treatments, GPT-3.5's time costs are two to three times higher.

This may be due to network congestion at the time of inference. Since GPT-3.5 is the underlying model for the free version of ChatGPT, it likely receives significantly more requests, leading to longer queuing times. On the other hand, for CoT treatments, GPT-4's time costs are two to three times higher than GPT-3.5's. As shown in Examples sections of our experiments, GPT-3.5 provides much shorter rationales, this in turn results in the observed time differences between the two models.

HX's classification methods, whether content-based or gamified, incurred costs of approximately $1000 to classify 38 messages within 2 hours. Extrapolating from this, classifying 100 messages would cost around $2500 and take about 5 hours. In contrast, given the high classification accuracy of GPT-4 (ranging from 92% to 97%) and its desired responsiveness to variations in instructions, the use of GPT-4 presents a more appealing option than employing twenty-five human annotators in a lab to classify messages.

Prompts in Promise I are the shortest, whereas those in Level-k II are the longest, leading to higher input token costs for Level-k II. Additionally, the examples from the rational provided for each experiment suggest that in GPT-4, longer prompts tend to correlate with more extensive rationales. Consequently, in CoT treatments, Level-k II incurs higher output token costs and time costs compared to other experiments. Furthermore, in Level-k II experiments, we instruct the models to generate four distinct classification outputs, as opposed to a single output in other experiments, which further increased time and monetary costs.

In Table 23 displays the costs for each model under each prompt type in the Level-k II experiments. For both models, monetary costs can be up to ten times higher than those observed in the Promise I experiments. However, the time costs for GPT-3.5 do not exceed those in Promise I, and for prompt O_+, they are consistently lower, potentially again due to the fluctuating demand for GPT-3.5. On the other hand, time costs for GPT-4 are significantly higher, with the average time of 42.7 minutes to classify 100 messages using prompt O under CoT treatment.

Under either model, due to the significantly larger prompt size of O, it is consistently more costly both in terms of time and money compared to O_+. Yet, together with the results from Table 19, we observe that the most costly prompt is not necessarily the best performing one.

		no-CoT		CoT	
		Money	*Time*	*Money*	*Time*
GPT-3.5	O	0.5	3.3	0.5	6.3
	O_+	0.3	1.5	0.3	2.8
GPT-4	O	5.2	5.2	6.6	42.7
	O_+	2.7	3.8	3.4	25.8

Table 23: Level-k II: Money and Time costs per 100 messages in USD and minutes.

10. Discussion

Tables 24a and 24b present the highest and lowest accuracy achieved for each model in each experiment. P_I and P_{II} refer to sections Promise I and II, and L_I and L_{II} refer to sections Level-k I and II. For P_I, these values are calculated by averaging the best or worst performances over the benchmarks.

	P_I	P_{II}	L_I	L_{II}	P_I	P_{II}	L_I	L_{II}
GPT-3.5	85.6	75.7	75.2	65.3	74	53.2	64.1	44.2
GPT-4	96	88.7	91.3	72.8	83.2	67.5	76.3	64
	(a) Highest				(b) Lowest			

Table 24: Highest and lowest accuracy across Experiments

Across all experiments, GPT-4 consistently outperforms GPT-3.5 in peak performance, and achieves an accuracy above or near 90% in all experiments except L_{II}. By switching from using GPT-3.5 to GPT-4, the best performance improves by 9 to 16 percentage points, and the lowest performance improves by 9 to 20 percentage points. Moreover, except for P_{II}, GPT-4's lowest performance is comparable to, or exceeds, the top performance of GPT-3.5. Therefore, substantial gains in classification performance can be achieved as the models scale.

In P_I and P_{II}, although the classification concept is the same and the classification instructions provided are either identical or similar, there is a noticeable decline in the top performance from P_I to P_{II} . In P_I, by design, subjects are incentivized to provide their intend in a single message. Consecutively, often the messages are explicit and thus relatively straightforward to classify. In contrast, in P_{II} subjects engage in conversation where they try to persuade others to take certain actions without providing their intentions, affirm or oppose previously made statements by others, explicitly state or hint at what they intend to do, and so on. Consequently, chat messages encom-

pass a broader array of linguistic patterns and colloquial, often ambiguous language. Additionally, chat messages are interdependent, and hence, are required to be evaluated within the full context of the chat. This conversational and contextual complexity makes the classification task in P_{II} more challenging, leading to comparatively lower performance by the models.

The classification of strategic thinking levels requires understanding and applying specific game theory concepts, notably level-0 beliefs and iterative best responses, making it more cognitively demanding than classifying promises. Additionally, the coordination game in L_{II} presents a more challenging classification task compared to the voting game in L_I. In the voting game, identifying level-1 and level-2 thinkers is relatively straightforward: subjects voting according to their received signals are likely level-1 thinkers, while those consistently voting red and considering the opposing team's behaviour are likely level-2 thinkers. These explicit cues simplify the classification task for models by reducing the need for a deep understanding of level-0 beliefs and best responses.

In contrast, the coordination games of L_{II} involve more complex reasoning patterns, such as "We think that they think that we think," which can perplex even expert annotators. The cyclical nature of these arguments requires models first to accurately identify in the message the level-0 belief and the label or payoff salience associated with it, then to identify the number of iterative best responses applied to this belief. This multi-step process increases the likelihood of classification errors at steps prior the classification of level of thinking and adds complexity to the task. Furthermore, instead of explicitly stating beliefs about the other team's intentions, such as "the other team wants the higher payoff for themselves," players might use indirect language, describing the other team as "selfish" or "offensive". Similarly, they might imply additional levels of reasoning with statements like "but they may also think this way". The use of such implicit linguistic patterns makes the classification task even more challenging for models. Additionally, unlike in L_I where a single level of thinking is classified, L_{II} requires models to classify both a lower and an upper bound for the subject's level of thinking, adding another layer of complexity. Consequently, it is no surprise that models exhibit lower performance in L_{II}.

Comparing the best and worst performances for each experiment reveals that different prompts and prompting techniques produce a variation of 8 to 20 percentage points for GPT-4 and 11 to 22 percentage points for GPT-3.5. This variation underscores the critical role of prompt engineering and suggests a substantial potential for performance gains with prompt optimisation. Moreover, it underscores the necessity for

social scientists to rigorously explore prompt variations and evaluate existing prompting techniques for their effectiveness in improving the model's performance on each distinct task relevant to social sciences.

In P_I, we relaxed the conditions for classifying promises across a series of prompts. GPT-4's classifications reflected this instructional variation, whereas GPT-3.5 did not respond to it. In P_{II}, we considered an alternative to the original codebook's classification instructions, where "empty talk" was implicitly defined as any message not meeting the criteria for a "promise". Instead, we provided explicit conditions for classifying "empty talk". This alternative prompt led to an average improvement of 15 percentage points in 0-shot treatments and 4.2 percentage points in n-shot treatments for GPT-4, and 13.7 and 17.2 percentage points improvements, respectively, for GPT-3.5. In L_I, an alternative prompt that included more detailed context information resulted in a 3 percentage point decrease in accuracy for GPT-4 on average, but a 4.1 percentage point increase for GPT-3.5. The increase in GPT-3.5 was primarily observed with the n-shot treatment. Lastly, in L_{II}, we experimented with one prompt providing nearly all information from the original codebook, resulting in a lengthy 2200-word prompt, and another with minimal information that was considerably shorter, around 900 words[30]. For GPT-4, the longer, more detailed prompt slightly reduced accuracy by an average of 1.8 percentage points, whereas for GPT-3.5, it resulted in an average decrease of 9.3 percentage points. In summary, we observed that:

1. GPT-4 is responsive to small yet crucial changes in the instructions, while GPT-3.5 is not (P_I).

2. Providing more comprehensive classification instructions improves performance for both models (P_{II}).

3. Additional context information at best slightly worsens GPT-4's performance and produces variable effects on GPT-3.5's performance, ranging from slight improvements to significant reductions as the complexity of the task increases (L_I L_{II}).

In Table 25, the effect of CoT and n-shot prompting on each experiment are presented. A "+" indicates that the treatment increased model performance across all prompts, a "−" indicates that it decreased performance across all prompts, and a "∼" indicates that the treatment had mixed effects across different prompts.

Overall, prompting techniques generally improve model performance. Notably, except for the consistent negative effect of CoT prompting on GPT-3.5 under P_{II}, these

---

[30]The n-shot variation included an additional "Example" section, adding about 500 words.

		P_I	P_{II}	L_I	L_{II}
GPT-3.5	CoT	$+$	$-$	$\sim$	$+$
	n-shot		$+$	$\sim$	$+$
	n-shot & CoT		$\sim$	$\sim$	$+$
GPT-4	CoT	$\sim$	$\sim$	$\sim$	$+$
	n-shot		$\sim$	$+$	$+$
	n-shot & CoT		$+$	$+$	$+$

Table 25: Overall effect of treatments on model performance

techniques invariably improve performance of both models for at least one of the considered prompts. On the other hand, for mixed effect cases ($\sim$), the technique's average effect over the prompts is negative for GPT-3.5 but positive for GPT-4. This suggests that on average, prompting techniques are less effective at improving classification performance for GPT-3.5 compared to GPT-4. Furthermore, n-shot prompting is more often effective at improving performance of each model than CoT prompting, and combining n-shot and CoT prompting consistently leads to improvements only in GPT-4's performance.

Table 26 shows under which treatments best and worst performance for each experiment are observed. "$Best$" and "$Worst$" columns under each experiment indicate the best and worst performance cases, respectively. A "✓" indicates that CoT or n-shot were applied, while a "×" denotes their absence. Specifically, a "×" for n-shot treatment denotes the 0-shot treatment and for CoT treatment, it represents the no-CoT treatment. A "$\sim$", only used under P_I, indicates that the best performance was achieved with the CoT treatment in three out of five benchmarks.

		P_I		P_{II}		L_I		L_{II}	
		$Best$	$Worst$	$Best$	$Worst$	$Best$	$Worst$	$Best$	$Worst$
GPT-3.5	CoT	✓	×	×	✓	×	✓	✓	×
	n-shot			✓	×	✓	×	✓	×
GPT-4	CoT	$\sim$	×	✓	×	✓	✓	✓	×
	n-shot			✓	×	✓	×	✓	×

Table 26: Active treatments for best and worst performance

Except for P_I, best and worst performances are consistently achieved in n-shot and 0-shot treatments, respectively, for either model. For GPT-4, the best performance is

also invariably achieved in CoT treatments; and except for L_I, the worst performance is consistently observed in no-CoT treatments. For GPT-3.5, in P_I and L_{II}, best and worst performances are observed in CoT and no-CoT treatments. Conversely, in P_{II} and L_I, the best performance occurs in CoT treatments and the worst in no-CoT treatments.

In Table 27, we present the average performance gains from switching from GPT-3.5 to GPT-4 (Δ_{GPT}) and from incorporating n-shot prompting (Δ_n) under no-CoT and CoT treatments. Δ_{GPT} shows minimal variation between experiments under no-CoT treatments across experiments. However, when CoT prompting is introduced, its positive effect diminishes for P_I, increases for P_{II} and L_I, and remains the same for L_{II}. Δ_n is similar in P_{II} and L_I, but is approximately three times greater in L_{II} under no-CoT. Furthermore, the incorporation of CoT consistently increases Δ_n, underscoring a positive synergy between the treatments.

		P_I	P_{II}	L_I	L_{II}
no-CoT	Δ_{GPT}	13.3	13.2	10.5	13
	Δ_n		2.5	2.2	6.6
CoT	Δ_{GPT}	9	17.6	18	13
	Δ_n		6.3	7.1	8.1

Table 27: Average performance gains

Pan et al. (2023) suggest that learning tasks benefit from larger LLMs and examples (n-shot prompting), while recognition tasks see minimal gains from them. Isolating the effects of scale and demonstrations and examining Δ_{GPT} and Δ_n under no-CoT treatments, we observe that all experiments benefited from an increase in the scale of the LLM. While all experiments also derived some benefit from demonstrations, L_{II} showed the largest improvement. These findings indicate that all tasks involve elements of learning, and notably, the task of classifying level-k thinking in coordination games (L_{II}) may encompass a relatively larger learning component.

In this study, we adhered closely to the original instructions to minimize the effort required for researchers to utilize LLMs with existing codebooks tailored for human annotators. Given that the original codebooks for P_{II} and L_I included classification examples, n-shot treatment naturally aligns with the existing instructions. Furthermore, 0-shot-CoT does not conflict with our objective of adhering as closely as possible to the original codebooks, as it can be seamlessly integrated with existing instructions. Consequently, we selected n-shot prompting and 0-shot-CoT as our primary treatments.

Computer science literature typically discusses n-shot prompting and n-shot-CoT prompting –which includes both examples and their reasoning– separately, and to our knowledge, there is no documented experimentation on the combined use of n-shot prompting and 0-shot-CoT for classification tasks (Dong et al., 2022; Huang and Chang, 2022; Wei et al., 2022a). Our results suggest that combining n-shot prompting with 0-shot-CoT, without the need to provide rationales for the demonstrations, is a viable and less demanding alternative for GPT-4. Although we do not claim that this combination yields higher performance than n-shot-CoT –as we have not investigated such a comparison– we observe that it can be considered as an alternative for social scientists who want to consider prompts closely based on existing codebooks and prefer not to devise rationales for their examples, a feature that is not typically included in codebooks tailored for human annotators.

If we consider moving beyond the strict adherence to original codebooks, it might be possible to improve model performance by developing instructions specifically tailored for LLMs. One approach could involve segmenting the classification of the level of thinking into three subtasks: identifying level-0 beliefs, classifying salience type, and determining the level of thinking, using the output from each as input for the next, a technique commonly refereed as prompt chaining (Wu et al., 2022; Anthropic, 2024). This method allows each subtask to have its own optimized prompt, focused solely on that specific aspect. Additionally, the classification of level of thinking could be further refined. In an initial step, the model could assess whether a message mentions others' actions or beliefs; messages without such mentions could be classified as level-0. Subsequently, messages that reference others' actions or beliefs could be analyzed in a separate prompt to determine the subject's level of thinking.

The current paper judges the classification performance of GPT relative to the benchmark of human classifications. We acknowledge that human classifications, even the ones agreed between 2 RAs, might deviate from an ideal "true classification". After all, these classifications are specific to the instructions given and might not be perfectly consistent due to learning, fatigue, etc.

High accuracies above 90% – as observed for some LLM classifications here – raise the question whether the human classification might not be the appropriate benchmark, because its own accuracy will likely be below 100% accuracy with the "true classification". For future comparisons, we intend to prepare "high-quality classification benchmarks" that represent the consensus of multiple diverse (expert, layman, LLM, ...) coders after sufficient deliberation.

Given GPT-4's out-performance over GPT-3.5, we are optimistic that LLM's will be

a significant facilitator of future research efforts with text as data. Kaplan et al. (2020) show that as the models' get larger both in their parameter size and in their pre-training corpus, their performance will steadily increase. Hence, we expect the observed performance levels to further improve with larger language models. Furthermore, in the last few months since we conducted our experiments, the computing cost for GPT-4 significantly decreased (approximately 75% cheaper). Lastly, for more involved classification tasks where LLM's performance consistently stays less than desirable, it can at the very least be considered within a human-in-the-loop classification system to assist and improve human annotators' performance.

11. Conclusion

In this study, we investigated the classification of promises and levels of strategic thinking using GPT models. Our results showed that GPT-4 achieves accuracy levels comparable to human annotators and surpass the performance of traditional NLP methods. The comparison between GPT-3.5 and GPT-4 revealed that larger models are more adept at handling relatively more involved classification tasks and benefit more from n-shot and 0-shot-CoT prompting techniques. Furthermore, our investigations of n-shot and 0-shot-CoT prompting on model performance showed that demonstrations generally improved model performance, and particularly for GPT-4, asking the model to provide a step-by-step reasoning before making its classifications often led to improved task performance.

We investigated classification tasks that vary in difficulty both conceptually (promise vs strategic thinking) and in linguistic diversity (standalone message vs conversation), and showed that as task difficulty increases, the model size and prompting techniques became more crucial in achieving performance levels comparable to those of human annotators. Additionally, the consistent improvement in model performance due to increased model size and n-shot prompting suggested that all tasks we considered incorporate a learning component. These effects, and consequently the learning aspect of the classification tasks, are observed to be more pronounced in the classification of strategic thinking compared to promises, especially with the classification of strategic thinking in coordination games, which additionally required an understanding of level-0 beliefs and related payoff or label salience.

We explored adapting existing classification codebooks, tailored for human annotators, to serve as prompts for LLMs. This adaptation involved reframing and reformatting the instructions to better align with the models' input processing. Our findings

indicated that relatively shorter codebooks, which neither contain nor require extensive theoretical background knowledge for the classification task, can be effectively leveraged as prompts with minimal reframing and restructuring. By using prompts that closely adhere to these original codebooks, GPT-4 can achieve satisfactory classification performance, with accuracy levels reaching or exceeding 90%. For longer codebooks, we observed that using the detailed background information verbatim in the prompts often hindered rather than improved the model's performance. Therefore, when reframing extensive codebooks, we recommend against anthropomorphising the model. Instead of providing explanations aimed at expanding its understanding of the topic –as one might do with a human annotator– focus should be placed on presenting textual patterns and linguistic cues that assist the model in a more mechanistic fashion during its classification task.

Lastly, we document that comprehensive classification instructions can be as effective as providing classification examples. This 0-shot method offers a viable alternative to n-shot prompting that relies on expanding a narrower set of classification conditions through examples. Furthermore, it mitigates issues associated with n-shot prompting, such as the model's sensitivity to the selection and order of examples, as well as the frequency of each category within these examples. Therefore, when revising the classification conditions in an existing codebook, researchers should aim to develop criteria with finer specificity with the intent to encompass a broader range of potential instances within each category of the dataset under analysis.

A. Promise I

A.1. Original Instructions

Figure 24: Original Instructions used for Content Analysis

```
Your task:
You will be given a list of messages.  Your task is to evaluate whether each of the messages is:

     • A statement of intent or promise

     • Empty Talk

The messages were written by participants in a previous experiment (Experiment I). To evaluate the messages,
you need to first understand Experiment I. The pages beginning on page 2 describe Experiment I. Please read
those pages carefully.  The message writer is in the role of subject B.

It is important for you to know more about how to code the messages before you read the instructions.  Here
are your specific instructions for how you code the messages:

     1.  (Weak Promise) You should code a message as ``A statement of intent or promise'' if you think at
         least one of the following conditions is probably satisfied.
         (Strong Promise) You should code a message as ``A statement of intent or promise'' if you think at
         least one of the following conditions is certainly satisfied.

            a.  The writer, subject B, indicates in the message he/she would do something favorable to
                subject A or refrain from doing something that hurts subject A.

            b.  The message gives subject A reasons to believe or expect that subject B would do something
                favorable to subject A or refrain from doing something that hurts subject A.

     2.  (Weak Promise) You should code a message as ``Empty Talk'', if the message does not probably satisfy
         any of the above conditions.
         (Strong Promise) You should code a message as ``Empty Talk'', if the message does not certainly
         satisfy any of the above conditions.

     3.  You should independently code all messages.  Do not discuss with anyone else in this room about how
         to code the messages.

     4.  Your job is to capture what had been said rather than why it was said or what effect it had.  Think
         of yourself as a ``coding machine.''

     5.  When you complete the coding, go through the entire list of messages a second time to (1) review all
         your codes and revise them if needed for accuracy; (2) make sure you code every message.
```

Figure 25: Original Instructions used for HX Classification Game

```
Your task:
In this experiment, you will be given a list of messages.  Your task in this experiment is to evaluate whether
each of the messages is:

     • A statement of intent or promise

     • Empty Talk

The messages were written by participants in a previous experiment (Experiment I). To evaluate the messages, you
need to first understand Experiment I. The next few pages describe Experiment I. Please read it carefully.  The
message writer is in the role of subject B.

Subjects were also told:
Weak Promise treatment:  ``You should classify a message as 'Promise or Intent' if, in your opinion, it includes
any statement of intent.''
Strong Promise treatment:  ``You should classify a message as 'Promise or Intent' if, in your opinion, it is
certainly a promise.''
```

A.2. Prompts

Figure 26: Prompt B^*

```
# General Task
- Evaluate a player's message in an economic experiment game to determine if it constitutes a promise.
# Context
- Players:  Two players, A and B.
- Game Mechanics:
  + First, player A chooses between the two decisions 'IN' and 'OUT'.
  + If player A chooses 'OUT', each player receives $5.
  + If player A chooses 'IN', player B chooses between the two decisions 'ROLL' or 'DON'T ROLL' (a die).
  + If player A chooses 'IN' and player B chooses 'DON'T ROLL', then player B receives $14 and A receives $0.
  + If player A chooses 'IN', and player B chooses 'ROLL', player B receives $10 and rolls a six-sided die to
determine player A's payoff.  If the die comes up 1 (1/6 likelihood), player A receives $0; if the die comes up
2-6 (5/6 likelihood), player A receives $12.
  + In addition, each player receives $5 show-up fee for participating in the experiment.
- Communication:  Prior to Player A's decision to 'IN' or 'OUT', player B has the option to send a message to
player A.
- Task:  Classify whether the message sent by player B constitutes a promise or not.
# Classification Task
- Classify player's message as '1' if it's a promise.
- Classify as '0' if it's an empty talk.
# Constraint
- Follow the below output format.
# Output Format
  0/1
```

Figure 27: Prompt O_S^*, O_N^* and O_W^*

```
# General Task
- Evaluate a player's message in an economic experiment game to determine if it constitutes a promise.
# Context
- Players:  Two players, A and B.
- Game Mechanics:
  + First, player A chooses between the two decisions 'IN' and 'OUT'.
  + If player A chooses 'OUT', each player receives $5.
  + If player A chooses 'IN', player B chooses between the two decisions 'ROLL' or 'DON'T ROLL' (a die).
  + If player A chooses 'IN' and player B chooses 'DON'T ROLL', then player B receives $14 and A receives $0.
  + If player A chooses 'IN', and player B chooses 'ROLL', player B receives $10 and rolls a six-sided die to
determine player A's payoff.  If the die comes up 1 (1/6 likelihood), player A receives $0; if the die comes up
2-6 (5/6 likelihood), player A receives $12.
  + In addition, each player receives $5 show-up fee for participating in the experiment.
- Communication:  Prior to Player A's decision to 'IN' or 'OUT', player B has the option to send a message to
player A.
- Task:  Classify whether the message sent by player B constitutes a promise or not.
# Classification Task
(O_S)- Classify a message as a "promise or a statement of intent" if at least one of the following conditions is
**certainly** satisfied:
(O_N)- Classify a message as a "promise or a statement of intent" if at least one of the following conditions is
satisfied:
(O_W)- Classify a message as a "promise or a statement of intent" if at least one of the following conditions is
**probably** satisfied:
  + Player B indicates in the message he would do something favorable to player A or refrain from doing something
that hurts player A
  + The message gives player A reasons to believe or expect that player B would do something favorable to player A
or refrain from doing something that hurts player A.
(O_S)- If the message does not **certainly** satisfy any of the above conditions, classify player B's message as
"Empty Talk".
(O_N)- If the message does not satisfy any of the above conditions, classify player B's message as "Empty Talk".
(O_W)- If the message does not **probably** satisfy any of the above conditions, classify player B's message as
"Empty Talk".
- Capture what had been said rather than why it was said or what effect it had.
- Operate as a "coding machine".
# Classification Coding
- Code player's message as '1' if it's a promise.
- Code as '0' if it's an empty talk.
# Constraint
- Follow the below output format.
# Output Format
0/1
```

A.3. Additional Examples

Figure 28: Promise I - Example Message 2

```
If you choose IN, and I roll, the chances of our getting the most $ are very high.  The likelihood of my
rolling a 1 is small compared to the chances of rolling a 2â6.  So we both get cash.
```

	CD	C_S	C_W	G_S	G_W
$Class.$	0	1	1	1	1
f_1	–	54%	76%	56%	76%

Table 28: Human Classifications for the message in Figure 28

	GPT-3.5		**GPT-4**	
	no-CoT	CoT	no-CoT	CoT
B	1	1	0	0
O_S	1	1	0	0
O_N	1	1	1	1
O_W	1	1	1	1

Table 29: GPT Classifications for the message in Figure 28

Figure 29: Classifications with CoT for Prompts O_W and O_S for Message in Figure 28

```
                                                                                            GPT-4
Output using O_W
Reasoning:
- Player B is explaining the probabilities of the game outcomes if player A chooses 'IN' and player B chooses
'ROLL'.
- Player B is indicating that the likelihood of rolling a 2-6 is high, which would be favorable to player A as it
would result in player A receiving $12.
- The message gives player A a reason to believe or expect that player B is leaning towards choosing 'ROLL', which
is favorable to player A.
Classification: 1

Output using O_S
Reasoning:
- Player B is explaining the probabilities of the outcomes if player A chooses 'IN' and player B chooses 'ROLL'.
- Player B is stating that the likelihood of rolling a 1 is small, which implies that the chance of player A
getting $12 is high.
- However, player B does not explicitly state what they will do; they are only discussing the probabilities and
potential outcomes.
- There is no clear commitment or statement of intent to choose 'ROLL' or 'DON'T ROLL'.
Classification:  0

                                                                                            GPT-3.5
Output using O_W
Reasoning:
The message indicates that if player A chooses 'IN' and player B rolls the die, there is a high likelihood of
both players getting cash.  Player B is providing a reason for player A to believe that choosing 'IN' and player B
rolling the die would result in a favorable outcome for both players.
Classification: 1

Output using O_S
Reasoning:
The message from player B indicates that if player A chooses 'IN' and player B chooses to roll, there is a high
likelihood of both players getting cash.  Player B is providing reasons for player A to believe that choosing 'IN'
and player B choosing to roll would result in favorable outcomes for both players.
Classification: 1
```

Figure 30: Promise I - Example Message 3

```
If you stay in, the chances of the die coming up other than 1 are 5 in 6 â pretty good.  Otherwise, weâd
both be stuck at $5.  (If you opt out)
```

	CD	C_S	C_W	G_S	G_W
$Class.$	0	1	1	0	1
f_1	–	73%	72%	< 50%	84%

Table 30: Human Classifications for the message in Figure 30

	GPT-3.5		**GPT-4**	
	no-CoT	CoT	no-CoT	CoT
B	1	1	0	0
O_S	1	1	0	0
O_N	1	1	0	0
O_W	1	1	1	1

Table 31: GPT Classifications for the message in Figure 30

Figure 31: Classifications with CoT for Prompts O_W and O_S for Message in Figure 30

```
                                                                                        GPT-4
Output using O_W
Reasoning:
- Player B is explaining the probabilities associated with the 'ROLL' decision, indicating that there is a high
chance (5 in 6) that player A will receive a favorable outcome ($12) if player B chooses to 'ROLL'. - Player B
contrasts this with the guaranteed but lower payoff of $5 each if player A chooses 'OUT'. - Although player B is
not explicitly stating what they will do, they are providing information that could influence player A's decision
by highlighting the potential benefits of choosing 'IN' and relying on player B to 'ROLL'. - The message gives
player A a reason to believe that choosing 'IN' could lead to a favorable outcome, which is indirectly suggesting
that player B might be inclined to choose 'ROLL' to allow for that favorable outcome.  Classification:  1

Output using O_S
Reasoning:
- The message from player B is providing information about the probabilities of the outcomes if player A chooses
'IN' and player B subsequently chooses 'ROLL'. - Player B is stating a fact about the likelihood of the die
roll outcome, which is a 5 in 6 chance of player A receiving $12. - However, player B does not explicitly state
what they will do if player A chooses 'IN'. There is no clear indication of player B's intended action, only a
statement of the odds.  - The message does not contain a clear promise or statement of intent to choose 'ROLL' or
'DON'T ROLL'. Classification:  0

                                                                                        GPT-3.5
Output using O_W
Reasoning:
The message provides reasoning to player A about the favorable outcome of choosing 'IN' and the likelihood of
receiving a higher payoff.  It also implies that player B would prefer player A to choose 'IN' for a better
outcome for both players.
Classification: 1

Output using O_S
Reasoning:
The message provides reasoning to player A about the favorable outcome of staying in, indicating that the chances
of the die coming up other than 1 are 5 in 6, which is pretty good.  It also highlights the unfavorable outcome of
opting out, where both players would be stuck at $5.  Therefore, the message gives player A reasons to believe or
expect that player B would do something favorable to player A.
Classification: 1
```

A.4. Additional Analysis

Weighted F1 ($F1_W$) and F1 macro averages for all four prompts across the five human classification baselines for both with (CoT) and without CoT (no-CoT) prompting are displayed in Tables 32 and 33, respectively.

		no-CoT					CoT				
		CD	C_S	C_W	G_S	G_W	CD	C_S	C_W	G_S	G_W
GPT-3.5	B	.65	.71	.74	.65	.88	.80	.83	.84	.80	.85
	O_S	.65	.71	.74	.65	.88	.76	.81	.85	.76	.92
	O_N	.60	.67	.70	.60	.83	.69	.75	.78	.69	.85
	O_W	.60	.67	.70	.60	.83	.73	.79	.82	.73	.89
GPT-4	B	.87	.87	.85	.87	.77	.90	.90	.88	.95	.79
	O_S	.93	.92	.90	**.97**	.81	**.97**	.92	.89	.92	.86
	O_N	.92	**.97**	.95	**.97**	.86	.92	.92	.95	.92	.90
	O_W	.83	.94	**.97**	.89	.87	.89	.94	**.97**	.89	**.93**

Table 32: Weighted F1

		no-CoT					CoT				
		CD	C_S	C_W	G_S	G_W	CD	C_S	C_W	G_S	G_W
GPT-3.5	B	.58	.62	.65	.58	.77	.78	.77	.80	.78	.76
	O_S	.58	.62	.65	.58	.77	.73	.78	.81	.73	.86
	O_N	.52	.56	.58	.52	.68	.64	.68	.70	.64	.74
	O_W	.52	.56	.58	.52	.68	.68	.74	.76	.68	.80
GPT-4	B	.86	.86	.83	.86	.70	.89	.89	.86	.95	.72
	O_S	.92	.92	.88	**.97**	.74	**.97**	.91	.88	.92	.80
	O_N	.92	**.97**	.94	**.97**	.80	.91	.90	.94	.91	.86
	O_W	.82	.94	**.96**	.88	.81	.88	.94	**.96**	.88	**.88**

Table 33: F1 Macro Averages

B. Promise II

B.1. Original Instructions

In Figure 32, the original instructions are presented with certain sections omitted that are not relevant to the prompt. Additionally, to accommodate the instructions on a single page, various line spacings have been reduced. The actual format of the original instructions is much easier to follow (see Arad et al. (2024) for further information).

Figure 32: Original Instructions

```
In the excel file, you will find a list of messages written by participants in an online experiment on
individual and group investments.
The list consists of many conversations between groups of three participants.
Your task is to evaluate, for each conversation, whether each of the participants stated an intent or a
promise to take a particular course of action.
You will classify a participantâs message in a conversation into one of the two categories:
  – A statement of intent or promise (1)
  – Other (0)
To evaluate the messages, you need to first understand the experiment.  The next page describes the
experiment, followed by more detailed instructions for your classification task.
(Explanation of the experiment is skipped)
Classification
Here are your specific instructions for how you code the messages.
1) You should code a participantâs message (including his/her entire text in the conversation) as âA
statement of intent or promiseâ if you think at least one of the following conditions is satisfied.
  a.  The writer indicates in the message he/she would take a certain course of action.
  b.  The message gives the other participants reasons to believe or expect that that the writer of the
message would take a certain course of action.
2) You should code a message as âotherâ, if the message does not satisfy any of the above conditions.
3) You should independently code all messages.  Do not discuss it with anyone else.
4) Your job is to capture what had been said rather than why it was said or what effect it had.  Think of
yourself as a âcoding machine.â
5) When you complete the coding, go through the entire list of messages a second time to review all your
codes and revise them if needed for accuracy.  Make sure you code every message.
Examples
Letâs illustrate how promises may look like:
Participant 1:  "all 200 then?"
Participant 2:  "yes"
The "yes" of Participant 2 is a promise.
Participant 1:  "Are we all just going to with max?â
Participant 2:  "agree"
The "agree" of Participant 2 is a promise.
It would not include statements such as
Participant 1:  "I think it's best if we invest 200" or
Participant 1:  "let's do 200" or
Participant 1:  "I have been doing 200 in the last round" or
Participant 1:  "100 sounds good".
Rather than spelling out numbers such as 200, people might refer to "max" or "all in".
In a particular context, "let's do it" may be a promise.  For example:
Participant 1:  "200?"
Participant 2:  "let's do it"
or similarly:
Participant 1:  "150.  player 2 are you in agreement?"
Participant 2:  "hi sounds good"
----------------------------------------------------------------------
It may include conditional promises (do something if someone else agrees).
For example, in
Participant 1:  "happy with 200 if we all agree"...
Participant 2:  "cool let's do it"
Participant 3:  "Yep."
In this case, all participants 1, 2 and 3 made promises.
But in
Participant 1:  "200 each?"
Participant 2:  "agree"
Participant 3:  "agree"
only participants 2 and 3 make a promise, because "200 each?" does not make it clear that the first
participant *will* do 200 if the others do that as well.
----------------------------------------------------------------------
Initial promises about which people later change their mind do not count:
Participant 1:  "200"
Participant 2:  "agreed"
Participant 3:  "I suggest 100"
Participant 2:  "I'm happy with either"
Participant 2 is not making any promise here.
```

B.2. Prompts

Figure 33: Prompt B

```
# General Task
- Evaluate each player in an investment game to determine whether he makes a promise or not.
# Context
- Players:  Group of three.
- Initial Endowment:  200 pence each.
- Investment:  Maximum 200 pence each.
- Mechanics:  Invested amount is doubled and split equally.
- Communication:  Players can chat before investing.
- Duration:  Multiple rounds.
# Classification Task
- Classify a player as '1' if he made a promise.
- Classify as '0' otherwise.
# Constraints
- Refrain from providing an explanation for your classification.  (only for no-CoT cases)
- Provide a final and single classification for each player.
- Follow the below output format.
# Output Format
P# :  0/1
```

Figure 34: Prompt O

```
# General Task
- Evaluate each player in an investment game to determine whether he makes a promise or not.
# Context
- Players:  Group of three.
- Initial Endowment:  200 pence each.
- Investment:  Maximum 200 pence each.
- Mechanics:  Invested amount is doubled and split equally.
- Communication:  Players can chat before investing.
- Duration:  Multiple rounds.
# Classification Task
- Classify a player's message as "a statement of intent or a promise" if at least one of the following
conditions is satisfied:
  + The message indicates that the player will take a certain course of action.
  + The message gives others reason to believe or expect that the player will take a certain course of action
- If the message does not satisfy either of the above conditions, classify it as an "Empty Talk".
- Capture what had been said rather than why it was said or what effect it had.
- Operate as a "coding machine".
# Classification Coding
- Code player's message as '1' if it's a promise or statement of intent.
- Code as '0' if it's an empty talk.
# Examples (Only in n-shot treatment) (See Figure 35)
# Constraints
- Refrain from providing an explanation for your classification.  (only for no-CoT cases)
- Provide a final and single classification for each player.
- Follow the below output format.
# Output Format
P# :  0/1
```

Figure 35: Example section for Prompt *O*

```
# Examples (Only in n-shot treatment)
## Example 1
### Chat
P1:  all 200 then?
P2:  yes
### Classification
P1:  0
P2:  1
## Example 2
### Chat
P1:  Are we all just going to with max?
P2:  agree
### Classification
P1:  0
P2:  1
## Example 3
### Chat
P1:  I think it's best if we invest 200
### Classification
P1:  0
## Example 4
### Chat
P1:  let's do 200
### Classification
P1:  0
## Remark
- Rather than spelling out numbers such as 200, people might refer to "max" or "all in".
## Example 5
### Chat
P1:  I have been doing 200 in the last round
### Classification
P1:  0
## Example 6
### Chat
P1:  100 sounds good
### Classification
P1:  0
## Remark
- In a particular contexts, "let's do it" or "sounds good" may be a promise (examples 7 and 8).
## Example 7:
### Chat
P1:  200?
P2:  let's do it
### Classification
P1:  0
P2:  1
## Example 8
### Chat
P1:  150.  player 2 are you in agreement?
P2:  hi sounds good
### Classification
P1:  1
P2:  1
## Remark
- A message may include conditional promises (do something if someone else agrees) (example 9).
## Example 9:
### Chat
P1:  happy with 200 if we all agree
P2:  cool let's do it
P3:  Yep.
### Classification
P1:  1 (conditional promise)
P2:  1
P3:  1
## Remark
- In example 10, only players 2 and 3 make a promise, because "200 each?" does not make it clear that the
first participant *will* do 200 if the others do that as well.
## Example 10
### Chat
P1:  200 each?
P2:  agree
P3:  agree
### Classification
P1:  0
P2:  1
P3:  1
## Remark
- Initial promises about which player later change their mind do not count (example 11).
## Example 11
### Chat
P1:  200
P2:  agreed (initial promise)
P3:  I suggest 100
P2:  I'm happy with either (P2 changes her mind)
### Classification
P1:  1
P2:  0 (due to change of mind after an initial promise)
P3:  0
```

```
# General Task
- Evaluate each player in an investment game to determine whether he makes a promise or not.
# Role Persona
- Act as a behavioral economist specialized in text classification, investment game and communication in
games.
# Context
- Players:  Group of three.
- Initial Endowment:  200 pence each.
- Investment:  Maximum 200 pence each.
- Mechanics:  Invested amount is doubled and split equally.
- Communication:  Players can chat before investing.
- Duration:  Multiple rounds.
# Classification Task
- Classify player's message as a promise if the player:
  + Explicitly states their intention to take a specific action.
  + Explicitly agrees to take an action suggested by another.
  + Commits to an action, conditional on a specific event occurring.
- Classify player's message as a non-promise if the player:
  + Suggests actions without commitment.
  + Asks questions or discusses preferences without explicitly committing.
  + Talks about hypothetical, ideal, or rational actions without explicit commitment.
# Classification Coding
- Code player's message as '1' if it's a promise.
- Code as '0' if it's a non-promise.
# Examples (Only in n-shot treatment)
1.  P1:  "I'll invest 150 pence if you do the same." (Code as 1 - Conditional promise)
2.  P2:  "What if we all invest 200 pence?" (Code as 0 - Question, no commitment)
3.  P3:  "Investing 200 pence seems the best strategy." (Code as 0 - Suggestion, no explicit commitment)
4.  P1:  "all 200 then?" P2:  "yes" (Code P2 as 1 - Clear affirmative response to a suggestion is a Promise,
code P1 as 0 - Question, no commitment)
5.  P1:  "Are we all just going with max?â P2:  "agree" (Code P2 as 1 - Clear affirmative response to a
suggestion is a Promise, code P1 as 0 - Question, no commitment)
6.  P1:  "I think it's best if we invest 200" (Code as 0 - Opinion, no commitment)
7.  P1:  "200?" P2:  "let's do it" (Code P2 as 1 - Clear affirmative response to a suggestion is a Promise,
code P1 as 0 - Question, no commitment)
8.  P1:  "happy with 200 if we all agree" P2:  "cool let's do it" P3:  "Yep." (Code P1 as 1 - Conditional
promises, code P2 and P3 as 1 - Clear affirmative response to a conditional promise)
9.  P1:  "200 each?" P2:  "agree" P3:  "agree" (Code P1 as 0 - Question, no commitment, Code P2 and P3 as 1
- Promises by P2 and P3 - clear affirmative response to a question)
# Constraints
- Refrain from providing an explanation for your classification.  (only for no-CoT cases)
- Provide a final and single classification for each player.
- Follow the below output format.
# Output Format
P# :  0/1
```

B.3. Additional Examples

Figure 37: Promise II - Example Message 3

```
P2:  Hello
P3:  Hi there
P3:  Any bots in here?
P2:  Nope
P1:  no
P3:  Nice one
P3:  OK
P3:  I suggest we all invest the maximum to max out our rewards
P2:  we should each invest fully for the best outcome
P3:  I agree, we are all here for the cash :)
P2:  I'm happy to do that
P2:  Excellent
P3:  If we just stuff 200 in we'll all get the most reward
P3:  I'm not here to be dishonest and cheat you all, we know how expensive life is right now
```

	GPT-3.5			GPT-4		
	P_1	P_2	P_3	P_1	P_2	P_3
B	0	1	1	0	1	1
O	0	0	0	0	1	1
O_+	0	1	1	0	1	0
RAs	0	1	1	0	1	1

Table 34: Human and n-shot CoT GPT classifications for the message in Figure 37

Figure 38: Promise II - Classifications with CoT for Prompts O and O_+ for message in Figure 37

```
                                                                              GPT-3.5
Output using O
Reasoning:
P2:  No specific promise made, just general agreement and positive attitude.
P3:  Suggests investing the maximum and agrees to it, emphasizing the desire for the best outcome.  Also mentions
not being dishonest and the expensive nature of life, but does not make a specific promise.
Classification:
P1:  0
P2:  0
P3:  0
Output using O+
Reasoning:
The conversation starts with casual greetings and small talk, with no explicit commitment to any action.  However,
as the conversation progresses, P3 suggests investing the maximum and emphasizes the benefits, while P2 agrees and
expresses happiness with the decision.  P3 also mentions being honest and not cheating, which implies a commitment
to the suggested action.
Classification:
P1:  0
P2:  1
P3:  1
                                                                              GPT-4
Output using O
Reasoning:
- P2 starts with greetings and does not make any promises in the initial exchanges.
- P3 asks a question and then makes a suggestion to invest the maximum, which is a promise to invest the maximum
if others agree.
- P2 then agrees with P3's suggestion and states a willingness to invest fully, which is a promise.
- P3 reiterates the agreement and mentions being there for the cash, which is not a promise but an expression of
intent or motivation.
- P2 expresses happiness to follow through with the full investment, which is a reaffirmation of the promise.
- P3 then explicitly states the intention to invest 200, which is a promise, and follows up with a statement about
honesty and not cheating, which reinforces the commitment to the promise but is not a promise in itself.
Classification:
P1:  0
P2:  1
P3:  1
Output using O+
Reasoning:
- P2 starts with a greeting, which is not a promise (Code as 0).
- P3 also greets and asks a question about bots, which is not a promise (Code as 0).
- P2 and P1 confirm there are no bots, which are statements of fact, not promises (Code as 0).
- P3 makes a suggestion to invest the maximum, which is a suggestion without a commitment (Code as 0).
- P2 echoes the suggestion that each should invest fully for the best outcome, which is still a suggestion without
explicit commitment (Code as 0).
- P3 agrees with the sentiment that they are all there for the cash, which is an expression of agreement with the
rationale but not a commitment to action (Code as 0).
- P2 then expresses happiness to follow the suggestion, which is a commitment to action (Code as 1).
- P2's "Excellent" is an expression of satisfaction, not a promise (Code as 0).
- P3 states what they should do ("If we just stuff 200 in we'll all get the most reward"), which is a continuation
of the suggestion and rationale but not an explicit commitment (Code as 0).
- P3's statement about not being there to be dishonest is an expression of intent but does not constitute a
promise regarding the specific action of investing (Code as 0).
Classification:
P1:  0
P2:  1
P3:  0
```

```
P2:  Hi
P1:  Hi all .  What say you on how much to put in?
P2:  I recommend investing 200 each for maximum returns
P3:  same
P1:  OK:)
P2:  Please stick to it
```

	GPT-3.5			**GPT-4**		
	P_1	P_2	P_3	P_1	P_2	P_3
B	0	1	0	0	0	0
O	0	1	0	0	1	1
O_+	0	1	0	0	0	0
RAs	1	0	0	1	0	1

Table 35: Human and n-shot CoT GPT classifications for the message in Figure 39

Figure 40: Promise II - Classifications with CoT for Prompts O and O_+ for message in Figure 39

```
                                                                                     GPT-3.5
Output using O
Reasoning:
P2's message "I recommend investing 200 each for maximum returns" constitutes a promise as it indicates a specific
action and gives others reason to believe or expect that P2 will invest 200.  P1's message "OK:)" does not
constitute a promise as it does not indicate a specific action.
Classification:
P1:  0
P2:  1
P3:  0
Output using O+
Reasoning:
P2:  The message "I recommend investing 200 each for maximum returns" suggests a specific action and implies a
commitment to that action.
Classification:
P1:  0
P2:  1
P3:  0

                                                                                     GPT-4
Output using O
Reasoning:
P2 starts with a greeting, which does not constitute a promise.  P1's message is a question about the investment
amount, which is also not a promise.  P2 then recommends investing 200 each, which is a suggestion but does not
explicitly state that P2 will do so; however, the context implies that P2 is advocating for this action and is
likely to follow through with it, so it can be considered a promise.  P3's response "same" indicates agreement
with P2's suggestion and implies a commitment to invest 200 as well, which can be classified as a promise.  P1's
"OK:)" is an acknowledgment of the plan but does not explicitly state that P1 will invest 200, so it is not a
promise.  However, P2's follow-up with "Please stick to it" is a request for commitment, reinforcing the earlier
promise to invest 200.
Classification:
P1:  0
P2:  1
P3:  1
Output using O+
Reasoning:
- P2 starts with a greeting, which does not contain any commitment or promise.
- P1 asks for opinions on the investment amount, which is a question without commitment.
- P2 recommends investing 200 each, which is a suggestion but does not explicitly state that they will do so
themselves, hence no commitment.
- P3 agrees with P2's suggestion, which could be interpreted as a commitment, but without an explicit statement of
personal action, it remains a general agreement.
- P1 responds with "OK:)" which indicates agreement but does not explicitly state a commitment to invest a
specific amount.
- P2 then asks to "Please stick to it," which is a request for others to commit but is not a self-commitment.
Classification:
P1:  0
P2:  0
P3:  0
```

97

Figure 41: Promise II - Example Message 5

```
P2:  max?  then doubled.
P1:  200
P3:  Yes, agree.
P2:  *thumbs up*
```

	GPT-3.5			GPT-4		
	P_1	P_2	P_3	P_1	P_2	P_3
B	0	1	0	0	0	0
O	0	1	0	0	0	1
O_+	0	0	1	1	0	1
RAs	0	1	1	0	1	1

Table 36: Human and n-shot CoT GPT classifications for the message in Figure 41

Figure 42: Promise 2 - Classifications with CoT for Prompts O and O_+ for message in Figure 41

```
                                                                           GPT-3.5
Output using O
Reasoning:
P2:  The message "max?  then doubled." indicates a specific action by the player and gives others reason to
believe or expect that the player will invest the maximum amount.  Therefore, it constitutes a promise.
P1:  The message "200" does not constitute a promise as it does not indicate a specific action by the player or
give others reason to believe or expect a certain course of action.
P3:  The message "Yes, agree." does not constitute a promise as it does not indicate a specific action by the
player or give others reason to believe or expect a certain course of action.
P2:  The message "*thumbs up*" does not constitute a promise as it does not indicate a specific action by the
player or give others reason to believe or expect a certain course of action.
Classification:
P1:  0 P2:  1 P3:  0
Output using O+
Reasoning:
P2 suggests "max?  then doubled." without explicitly committing to the action.  P1 responds with "200" which could
be interpreted as a suggestion or a statement of intention, but it's not explicitly a promise.  P3 responds with
"Yes, agree," which indicates a clear affirmative response but is conditional on the previous messages.  P2's
"thumbs up" could be interpreted as agreement, but it's not a clear commitment.
Classification:
P1:  0 P2:  0 P3:  1


                                                                            GPT-4
Output using O
Reasoning:
P2's initial message "max?  then doubled." is a question and a statement of the game's mechanics, not a promise to
invest a specific amount.
P1's message "200" is a clear indication of the amount they are considering to invest.  However, without
additional context or a statement that commits them to this action, it is not a promise.
P3's message "Yes, agree." is a response to the previous messages, indicating agreement with the proposed action.
This can be interpreted as a commitment to invest the amount that is being discussed, which in this context is 200
pence.
P2's message "*thumbs up*" is a non-verbal indication of agreement or approval but does not constitute a verbal
promise to take a specific action.
Classification:
P1:  0 P2:  0 P3:  1
Output using O+
Reasoning:
- P2's initial message "max?  then doubled." is a question about the game mechanics, not a commitment to invest a
specific amount.  Therefore, it's not a promise.
- P1's response "200" is a statement of a specific amount, which in this context seems to be an agreement to
invest 200 pence.  This can be interpreted as a promise to take that action.
- P3's response "Yes, agree." is an explicit agreement to the previously mentioned action (investing 200 pence).
This is a promise to take that action.
- P2's response with a "*thumbs up*" emoji is a non-verbal affirmative response, which in this context can be
interpreted as an agreement to the proposed action.  However, it's less explicit than a verbal commitment, and
without additional context, it's not as strong as a clear statement of intent.  It could be seen as an endorsement
of the plan rather than a personal commitment to action.  Given the ambiguity, I will lean towards not classifying
it as a promise.
Classification:
P1:  1 P2:  0 P3:  1
```

Figure 43: Promise 2 - Example Message 6

```
P3:  Hi everyone
P2:  hi
P1:  I think we should do 100 each
P2:  i was thinking the same
P2:  are we agreed?
P1:  player 3?
P3:  If we all do 200 we get the most back I think?
P2:  we should meet in the middle then and do 150?
P1:  I'm good with that
P3:  Hep
P3:  Yep
P1:  Lets do it
```

	GPT-3.5			**GPT-4**		
	P_1	P_2	P_3	P_1	P_2	P_3
B	1	1	1	0	0	0
O	0	0	0	0	1	0
O_+	1	1	1	1	1	1
RAs	1	1	1	1	1	1

Table 37: Human and n-shot CoT GPT classifications for the message in Figure 43

Figure 44: Promise 2 - Classifications with CoT for Prompts O and O_+ for message in Figure 43

```
                                                                    GPT-3.5
Output using O
Reasoning:
- P1 suggests an amount (100 each) but does not explicitly promise to invest that amount.
- P2 agrees with the suggestion but also does not explicitly promise to invest that amount.
- P3 suggests investing 200 each, but does not explicitly promise to invest that amount.
- P2 suggests meeting in the middle at 150, but does not explicitly promise to invest that amount.
- P1 agrees with the suggestion but does not explicitly promise to invest that amount.
- P3 confirms agreement with "Yep" but does not explicitly promise to invest that amount.
- P1's "Let's do it" does not explicitly promise to invest 150.
- P2's "50 it is then" does not explicitly promise to invest 150.
Classification:
P1:  0 P2:  0 P3:  0
Output using O_+
Reasoning:
P3 initiates the discussion by suggesting that investing 200 each would yield the most return.  P2 then suggests
meeting in the middle at 150, to which P1 agrees.  P3 responds with "Yep" and P2 confirms "50 it is then."
Classification:
P1:  1 P2:  1 P3:  1

                                                                    GPT-4
Output using O
Reasoning:
P3 starts with a greeting, which is not a promise ('0').  P2 also greets, which is not a promise ('0').  P1
suggests doing 100 each, which is a proposal but not a commitment ('0').  P2 agrees with the idea of 100 each and
asks for confirmation ('0'), as this is still part of the negotiation process.  P1 asks for P3's input, which is
not a promise ('0').  P3 suggests that investing 200 each would yield the most return, which is an opinion, not
a promise ('0').  P2 then suggests meeting in the middle with 150, which is a counter-proposal but not a firm
commitment ('0').  P1 agrees with the 150 proposal, which is an acceptance of the suggestion but not a clear
promise ('0').  P3 agrees with "Yep," which is an affirmation but not a clear commitment ('0').  P1 then says
"Let's do it," which is an agreement to proceed but not a specific promise to invest a certain amount ('0').  P2
finally states "150 it is then," which is a confirmation of the agreed amount and can be considered a promise to
invest that amount ('1').
Classification:
P1:  0 P2:  1 P3:  0
Output using O_+
Reasoning:
P1 starts with a suggestion to invest 100 each, which is not a commitment but an opinion (Code as 0).  P2
expresses agreement with the initial suggestion and asks for confirmation (Code as 0 for agreement without
commitment, and 0 for the question).  P1 asks P3 for their opinion, which is a question and not a commitment (Code
as 0).  P3 suggests that investing 200 each would yield the most return, which is an opinion and not a commitment
(Code as 0).  P2 then suggests meeting in the middle with 150 each, which is a suggestion without a commitment
(Code as 0).  P1 agrees with P2's suggestion, which is a commitment to the action (Code as 1).  P3 agrees with the
suggestion by saying "Yep," which is a commitment (Code as 1).  P1 then reaffirms the decision with "Lets do it,"
which is a commitment (Code as 1).  P2 confirms the amount with "150 it is then," which is a commitment (Code as
1).
Classification:
P1:  1 P2:  1 P3:  1
```

99

B.4. Additional Analysis

In Tables 38 and 39, weighted F1 and F1 macro averages are presented. Since the "promise" and "empty talk" categories are balanced with 53.3% of the messages classified as "promise" in the groundtruth, weighted F1 and F1 macro averages are very close to the accuracy results reported in Table 10 in Section 6.4.

		no-CoT		**CoT**	
		$0\text{-}Shot$	$n\text{-}Shot$	$0\text{-}Shot$	$n\text{-}Shot$
GPT-3.5	B	.57	–	.614	–
	O	.574	.574	.547	.525
	O_+	.719	.749	.672	.75
GPT-4	B	.674	–	.707	–
	O	.684	.827	.718	.848
	O_+	.865	.862	.841	**.887**

Table 38: Weighted F1

		no-CoT		**CoT**	
		$0\text{-}Shot$	$n\text{-}Shot$	$0\text{-}Shot$	$n\text{-}Shot$
GPT-3.5	B	.563	–	.614	–
	O	.572	.595	.546	.52
	O_+	.717	.746	.673	.751
GPT-4	B	.675	–	.71	–
	O	.681	.826	.721	.847
	O_+	.865	.861	.841	**.886**

Table 39: F1 Macro Averages

C. Level-k I

C.1. Original Instructions

Figure 45: Original Classification Instructions

General Comments:
Subjects do not necessarily describe every step of their thinking; therefore, it may not always be obvious to decide which level they are. In many comments, any indications of a level of thinking may be partial or implicit, you should then indicate the most likely level of reasoning of the player. If the message indicates to simply refer to a previous message ("same as before/above"), then you can use the previous message's evaluation to determine the level of the current message. If you are unsure of the level of the message, you should indicate the level you think is more likely.

Level-0 Player:
Characteristics: Chooses randomly, without justification or through some justification completely unrelated to the task. Might not have understood the game or shows no interest in the game or in thinking about it.
Examples:
"50 50 chance to get red at least 50 50 could also be 100 percent."
"I like blue, so I chose blue."
"Think it will be red again."
"definitely red this time"
"We have to go for red. No other way than that. I like turtles"
Note: Comments such as "It is obviously blue" or "Play red, trust me!" should not be considered as level-0 thinking as these comments to some extent signal some level of understanding/interest of the task. Such comments are likely to be level-1 comments yet without any additional information, you should leave the specific cell empty.

Level-1 Player:
Characteristics: Always follows his own signal. The subject may argue in favor of playing his own signal through some probability argument
Examples:
"Our signal is blue. Let's play blue."
"The probability that the red ball we observe is out of the red urn is twice the probability that it is out of the blue urn"
"1/3 of all teams is observing wrong color, so we would try to find out whether we have wrong or right ball, keep with red."
Note: The key idea in defining a level-1 player is to identify some thinking process that signals the subject's interest/understanding of the task and the private signal. Furthermore, it is important that the subject does **not** offer any argument acknowledging the potential votes of the other teams and how to vote accordingly (i.e. adjusting the strategy given what others are expected to do).

Level-2 Player:
Characteristics: Assume that all other players almost always follow their signal (i.e. she assumes almost all the other players are level-1 while an epsilon portion of them are level-0). Player does offer an argument acknowledging the potential votes of the other teams and how to vote accordingly (i.e. a best response given others are most likely playing their signal). In other words, if you identify any comment that indicates that the subject assumes (or considers the case) where the other players in her group play their signal, you should consider the possibility that the subject is a level-2 player.
Examples:
"Let's take red because if the urn is red and we got the opposite color and we take blue, the decision will be blue."
"We need to chose Red. If we are the only ones who picked blue, then the urn is red and we guess correct If the urn is blue, then the other guys will pick blue so there will be at least one blue vote and we win as well If the others guys (also blue) think the same way then we lose But this is too many ifs"
"I have a blue ball. If we have the blue urn, someone else also has a blue ball and as a result our group will chose blue regardless of my vote. If we have the red urn, I am the only one with the blue ball and if I vote blue, we will chose the wrong urn. So I should vote for red."
"In case two teams choose red and one chooses blue, blue will be taken. That means that choosing red has a higher chance of being a good decision."
"I guess this is more about luck because there is no way to know it for sure. I would say blue just because of the higher probability. Also I like turtles Also it is likely that one other team will pick blue and then it is that color anyways"
"There is no point for us to take blue I think the chances for us to get the right color are higher if we stick with red" [red ball is observed]
"I suggest red because we donat hurt anyone with this decision If the others go for blue because they have a blue ball, the committeeas decision will be blue regardless of our decision"
"We could be the deciding vote for blue if the other two choose red. Choosing blue isnt as helpful as choosing red, because: only one blue ball can overturn our whole decision but only a unanimous decision for red can help us the same way"
Note: In order to discern the two types, you should look for more than any trivial arguments such as the ones given under level-1. There may be cases where the message starts as a level-1 argument and then as the subjects elaborates on her reasoning, she starts considering the strategy of the other teams and justify her decision accordingly (see the third example above). In such cases, this message should be considered as level-2. The acknowledgment of other teams' voting strategy may not always be obvious or may be worded differently such as "hurting the other's decision" or "not being helpful" (see the last three examples above)

Level-3 Player:
Characteristics:
Assumes that almost all other subjects are level-2 players (partially degenerate beliefs). The reasoning in a level-3 player message will have similarities with a level-1 player message but it will have additional arguments indicating that she assumes others are level-2 players.
Examples:
"In my opinion, if there is another person with blue they may be afraid of voting blue so we should vote blue to make sure."
"Let's now pick the shown colour because the others now will probably enter their opposite colour."
"Risky to vote blue but others may not vote blue even when they draw blue. I say we vote blue."
Note: As stated above, level-3 players are likely to follow their signal like a level-1 player yet they will argue to do so through a much more intricate argument (unlike a level-1 player merely stating probabilities to argue her action). Level-3 players are rare. Higher levels (level-4 etc.) are assumed to not occur; therefore, you should consider only the first 4 levels of thinking.

We have omitted various sections of the original instructions to fit in the instructions into a single page. The original instructions begins with a lengthy section regarding

the general theory of level-k modeling and its specific application to the experiments. These sections aggregate to 3 pages of instructions. Although the context of these sections are relevant in terms how we have very briefly summarized them in the "Context" section of our prompts, we omitted them in Figure 45. Furthermore, some sentences in the instructions that were not related to the classification of instructions, such as how to code certain concepts in the excel sheet, are also omitted. See Çelebi and Penczynski (2023) for the instructions in its entirety.

C.2. Prompts

Figure 46: Prompts O and O_+

```
# General Task
- Classify player's level of strategic thinking in a voting game
# Role Persona
- Act as a behavioral economist specialized in level-k modeling, strategic thinking and text classification.
# Context (see Figures 47 and 48)
# Classification Task
- Classify a player's level of strategic thinking as 0, 1, 2 or 3 based on the message provided.
- Use the below characteristics, examples and notes provided for each level to determine your classification.
## Level-0 Player
### Characteristics
- Chooses randomly, without justification or through some justification completely unrelated to the task.
- Might not have understood the game or shows no interest in the game or in thinking about it.
- Provides a vote without a clear justification to the probability of the game or strategic reasoning.
### Note
- Comments such as 'It is obviously blue' or 'Play red, trust me!' should not be considered as level-0 thinking as
these comments to some extent signal some level of understanding/interest of the task. Such comments are likely
to be level-1 comments.
## Level-1 Player
### Characteristics
- Always follows his own signal.
- The subject may argue in favor of playing his own signal through some probability argument.
### Note
- The key idea in defining a level-1 player is to identify some thinking process that signals the subject's
interest/understanding of the task and the private signal.
- It is important that the subject does not offer any argument acknowledging the potential votes of the other
teams and how to vote accordingly (i.e. adjusting the strategy given what others are expected to do).
## Level-2 Player
### Characteristics
- Assume that all other players almost always follow their signal (i.e. a level-2 player assumes almost all the
other players are level-1 while a small portion of them are level-0).
- Player does offer an argument acknowledging the potential votes of the other teams and how to vote accordingly
(i.e. a best response given others are most likely playing their signal).
- if you identify any comment that indicates that the subject assumes (or considers the case) where the other
players in her group play their signal, you should consider the possibility that the subject is a level-2 player.
### Note
- In order to discern between level-1 and level-2 types, you should look for more than any trivial arguments such
as the ones given under level-1.
- There may be cases where the message starts as a level-1 argument and then as the subjects elaborates on her
reasoning, she starts considering the strategy of the other teams and justify her decision accordingly (see the
fifth example above). In such cases, this message should be considered as level-2.
- The acknowledgment of other teams' voting strategy may not always be obvious or may be worded differently such
as "hurting the other's decision" or "not being helpful" a (see the last three examples above)
## Level-3 Player
### Characteristics
- Assumes that almost all other subjects are level-2 players (partially degenerate beliefs).
- The reasoning in a level-3 player message will have similarities with a level-1 player message but it will have
additional arguments indicating that she assumes others are level-2 players.
### Notes
- level-3 players are likely to follow their signal like a level-1 player yet they will argue to do so through a
much more intricate argument (unlike a level-1 player merely stating probabilities to argue her action).
- Level-3 players are rare.
## General Comments
- Players do not necessarily describe every step of their thinking; therefore, it may not always be obvious to
decide which level they are. In many messages, any indications of a level of thinking may be partial or implicit.
In such cases provide the most likely level of reasoning from the messages.
- If you are unsure of the level of the message, you should indicate the level you think is more likely.
# Constraint
- Only provide a single level classification.
- Follow the below output format.
# Output Format
0/1/2/3
```

"Classification Process" section used for CoT prompting and the variation in the "Output Format" are as presented in Figures 1 and 2 and are omitted. Since the prompts only differ in their "Context" section, we present both prompts all other identical parts in Figure 46. "Examples" section used for n-shot prompting is presented separately in Figure 49.

Figure 47: Prompt O - Context

```
# Context
- Teams of two players, part of larger groups, draw a colored ball from an urn, play a voting game with their
group to guess the color of the urn for multiple periods.
- Teams are randomly paired each period.  Teams consist always of 2 players.
- Teams are grouped into either 3 or 6 teams per group.
- Each period, a group is assigned to an urn with a blue or red color with equal probability.
- An urn only contains blue and red balls.  Blue urn has twice more blue balls than red balls and a red urn has
twice more red balls than blue balls.  The color of the ball has a 2/3 chance of matching the urn's color.
- After the urn with either red or blue color is assigned to a group, each team within a group draws a ball from
the urn to infer the urn's color.  The drawing can be with or without replacement depending on the period.
- Teams do not know the color of the urn.  They do not know the colors of the balls picked by the other teams in
their group.  They only know the color of their own ball.
- The group's objective is to correctly guess the color of the assigned urn.
- Each team in a group provides a single vote.  A team votes for either the color red or blue.  Group's decision
is determined based on the aggregation of its teams' votes.
- Teams communicate internally to decide on a vote for the urn's color.  If all votes are red, the group decision
is red; any blue vote results in a blue group decision.
- Teams do not observe the votes of the other teams in their group.
- Teams weigh their own ball's color and strategize their vote considering the group's outcome.
- Players exhibit levels of strategic reasoning (0 to 3), influencing their decision-making and messaging.
```

Figure 48: Prompt O_+ - Context

```
# Context
## Game Mechanics
- Teams of two players, part of larger groups, draw a colored ball from an urn, play a voting game with their
group to guess the color of the urn for multiple periods.
- Teams are randomly paired each period.  Teams consist always of 2 players.
- Teams are grouped into either 3 or 6 teams per group.
- Each period, a group is assigned to an urn with a blue or red color with equal probability.
- An urn only contains blue and red balls.  Blue urn has twice more blue balls than red balls and a red urn has
twice more red balls than blue balls.  The color of the ball has a 2/3 chance of matching the urn's color.
- After the urn with either red or blue color is assigned to a group, each team within a group draws a ball from
the urn to infer the urn's color.  The drawing can be with or without replacement depending on the period.
- In without replacement case, for group size of 3, if the urn is red (blue), 2 teams will draw red (blue) ball
and 1 team will draw blue (red) ball.  For the group size of 6, if the urn is red (blue), 4 teams will draw red
(blue) ball and 2 teams will draw blue (red) ball.
- In with replacement case, irrespective of the group size, the probability of drawing a red (blue) ball from the
red (blue) urn will be 2/3.
- Teams do not know the color of the urn.  They do not know the colors of the balls picked by the other teams in
their group.  They only know the color of their own ball.
- The group's objective is to correctly guess the color of the assigned urn.
- Each team in a group provides a single vote.  A team votes for either the color red or blue.  Group's decision
is determined based on the aggregation of its teams' votes.
- Teams do not observe the votes of the other teams in their group.
## Communication
- Teams communicate internally to decide on a vote for the urn's color.  If all votes are red, the group decision
is red; any blue vote results in a blue group decision.
- Each player within a team can only send a single message to their teammate.  Both teammates send their messages
before observing their teammate's message.
## Strategic Reasoning
- Teams weigh their own ball's color and strategize their vote considering the group's outcome.
- Players exhibit levels of strategic reasoning (0 to 3), influencing their decision-making and messaging.
- Level-0:  Any action that can be considered as random play, non-strategic or unrelated to the game mechanics
- Level-1:  Almost always votes the color of the ball picked.  Does not assume any behavior by other teams.
- Level-2:  Almost always votes red regardless of the color of the ball picked.  Assumes that everybody else
almost always votes the color of the ball they pick (i.e.  assumes everybody else is almost always level-1).
- Level-3:  Almost always votes the color of the ball picked.  Assumes that everybody else almost always votes red
regardless of the color of the ball they pick (i.e.  assumes everybody else is almost always level-2)
```

```
# Classification Task
## Level-0 Player
### Characteristics ...
### Examples
1.  "50 50 chance to get red at least 50 50 could also be 100 percent."
2.  "I like blue, so I chose blue."
3.  "Think it will be red again."
4.  "definitely red this time"
5.  "We have to go for red.  No other way than that.  I like turtles"
### Notes ...
## Level-1 Player
### Characteristics ...
### Examples
1.  "Our signal is blue.  Let's play blue."
2.  "The probability that the red ball we observe is out of the red urn is twice the probability that it is out of
the blue urn"
3.  "1/3 of all teams is observing wrong color, so we would try to find out whether we have wrong or right ball,
keep with red."
### Notes ...
## Level-2 Player
### Characteristics ...
### Examples
1.  "Let's take red because if the urn is red and we got the opposite color and we take blue, the decision will be
blue."
2.  "We need to chose Red.  If we are the only ones who picked blue, then the urn is red and we guess correct If
the urn is blue, then the other guys will pick blue so there will be at least one blue vote and we win as well If
the others guys (also blue) think the same way then we lose But this is too many ifs"
3.  "I have a blue ball.  If we have the blue urn, someone else also has a blue ball and as a result our group
will chose blue regardless of my vote.  If we have the red urn, I am the only one with the blue ball and if I vote
blue, we will chose the wrong urn.  So I should vote for red."
4.  "In case two teams choose red and one chooses blue, blue will be taken.  That means that choosing red has a
higher chance of being a good decision."
5.  "I guess this is more about luck because there is no way to know it for sure.  I would say blue just because
of the higher probability.  Also I like turtles Also it is likely that one other team will pick blue and then it
is that color anyways"
6.  "There is no point for us to take blue I think the chances for us to get the right color are higher if we
stick with red" [red ball is observed]
7.  "I suggest red because we donat hurt anyone with this decision If the others go for blue because they have a
blue ball, the committeeas decision will be blue regardless of our decision"
8.  "We could be the deciding vote for blue if the other two choose red.  Choosing blue isnt as helpful as
choosing red, because:  only one blue ball can overturn our whole decision but only a unanimous decision for red
can help us the same way"
### Notes ...
## Level-3 Player
### Characteristics ...
### Examples
1.  "If everyone else assumes others play their own signal then they will always play red.  Since I have the blue
ball, it is more likely that we have the blue urn so I will vote blue"
2.  "Let's now pick the shown colour because the others now will probably enter their opposite colour."
3.  "Risky to vote blue but others may not vote blue even when they draw blue.  I say we vote blue."
### Notes ...
```

C.3. Additional Examples

Figure 50: Level-k I - Example Message 4

```
the chance that we recieve the wrong ball is smaller than the chance that we receive the correct ball However I
think every team will think in the same way and there is no chance that the committe gets it right...
```

	GPT-3.5	GPT-4
O	2	1
O_+	2	1
RAs	0	

Table 40: Human and GPT Classifications of message in Figure 50 using n-shot and
CoT prompting

Figure 51: Level-k I - Classifications with CoT and n-shot for Prompts O and O_+ for message in Figure 19 or Figure 50

```
                                                                                    GPT-3.5
Output using O
Reasoning:
The player is considering the possibility of not picking the same color as their partner in order to avoid a tie.
They are strategizing based on the assumption that their partner will follow their signal.
Classification:
2
Output using O+
Reasoning:
The player is considering the possibility of not picking the same color as their teammate to avoid a unanimous
vote, indicating some level of strategic thinking.  They are suggesting a coordinated strategy to ensure their
vote is different from their teammate's.
Classification:
2


                                                                                    GPT-4
Output using O
Reasoning:
The player is suggesting a strategy that involves coordination with their teammate to cover both possible
outcomes.  This indicates that the player is considering the actions of their teammate and how it could affect
the group's decision.  The message implies a basic understanding of the game's mechanics and the importance of
diversifying the votes within the team.  However, the player does not explicitly acknowledge the strategies or
potential votes of other teams in the group, which would be characteristic of a level-2 player.  The message
suggests a simple strategy without deeper strategic reasoning about the other teams' likely actions.
Classification:
1
Output using O+
Reasoning:
The player is suggesting a strategy that involves coordination with their teammate to cover both possible
outcomes.  This message indicates that the player is considering the actions of their teammate and how it could
affect the group's decision.  However, the player does not explicitly consider the strategies of the other teams
in the group, which would be characteristic of a level-2 player.  The message also lacks the assumption that
other players are level-2, which would be necessary to classify the player as level-3.  The player's message
does not fit the characteristics of a level-0 player, as it is not a random choice or unrelated to the task.
Therefore, the message most closely aligns with level-1 reasoning, where the player is interested in the task and
is considering their own signal and that of their teammate but not the strategies of other teams.
Classification:
1
```

Figure 52: Level-k I - Example Message 5

```
I'd say always choose red but maybe we should choose blue since everbody thinks always choose red and it's 2/3
that we win.
```

	GPT-3.5	GPT-4
O	2	3
O_+	2	3
RAs	3	

Table 41: Human and GPT Classifications of message in Figure 52 using n-shot and CoT prompting

Figure 53: Level-k I - Classifications with CoT and n-shot for Prompts O and O_+ for message in Figure 52

```
                                                                              GPT-3.5
Output using O
Reasoning:
The player seems to be considering the strategy of the other players in the group.  They mention that everyone
thinks "always choose red" and then consider choosing blue as a response to this.  This indicates an awareness of
the potential votes of the other teams and how to vote accordingly.
Classification:  2
Output using O+
Reasoning:
The player starts by suggesting "always choose red" but then considers the possibility of choosing blue because
everyone else might be thinking the same way.  The player acknowledges the potential votes of the other teams and
how to vote accordingly, indicating a consideration of the strategy of the other teams.
Classification:  2


                                                                                GPT-4
Output using O
Reasoning:
The player starts by considering the strategy of "always choose red," which is indicative of level-2 thinking,
as it shows an awareness of a common strategy that assumes other players will vote their signal (level-1
behavior).  However, the player then suggests choosing blue, acknowledging that other players might also be
thinking strategically at level-2 and therefore might "always choose red." This suggests that the player is
considering the possibility that others are at level-2 and is adjusting their strategy accordingly, which is
characteristic of level-3 thinking.
Classification:  3
Output using O+
Reasoning:
The player starts with a consideration that might suggest a level-2 strategy ("always choose red") but then
moves on to consider what other players might be thinking.  The player acknowledges that other players might be
following a level-2 strategy ("everybody thinks 'always choose red'") and suggests adapting their own strategy
based on this assumption.  This indicates that the player is considering the strategic choices of others and is
trying to best respond to what they perceive as the common strategy.  This is characteristic of level-3 reasoning,
where the player assumes that others are level-2 players and adjusts their strategy accordingly.
Classification:  3
```

Figure 54: Level-k I - Example Message 6

```
the chance that we recieve the wrong ball is smaller than the chance that we receive the correct ball
However I think every team will think in the same way and there is no chance that the committe gets it
right...
```

	GPT-3.5	GPT-4
O	2	2
O_+	2	2
RAs	1	

Table 42: Human and GPT Classifications of message in Figure 54 using n-shot and CoT prompting

Figure 55: Level-k I - Classifications with CoT and n-shot for Prompts O and O_+ for message in Figure 54

```
                                                                                    GPT-3.5
Output using O
Reasoning:
The player acknowledges the probability of receiving the wrong ball being smaller than receiving the correct ball.
However, they also express a belief that every team will think in the same way, leading to the conclusion that
there is no chance that the committee gets it right.  This indicates a consideration of the potential votes of the
other teams and how to vote accordingly, suggesting a level-2 strategic thinking.
Classification:  2
Output using O+
Reasoning:
The player acknowledges the probability of receiving the wrong ball and the correct ball.  They also express a
belief that every team will think in the same way, leading to the conclusion that there is no chance that the
committee gets it right.  This indicates an awareness of the potential votes of the other teams and how to vote
accordingly, suggesting a level-2 strategic thinking.
Classification:  2


                                                                                      GPT-4
Output using O
Reasoning:
The player acknowledges the probability of receiving the correct ball versus the wrong ball, which shows an
understanding of the game's mechanics.  However, the player also considers the thought process of other teams,
assuming that they will think similarly and that this collective reasoning will lead to an incorrect group
decision.  This indicates that the player is considering the strategies of others and how those strategies will
affect the group outcome, which aligns with level-2 strategic thinking.
Classification:  2
Output using O+
Reasoning:
The player starts with a probability argument, which is a characteristic of level-1 thinking, by acknowledging
that the chance of receiving the correct ball is higher than receiving the wrong one.  However, the player then
moves on to consider what other teams might think and do, which is a characteristic of level-2 thinking.  The
player assumes that every team will think in the same way, which suggests they are considering the strategies of
other teams and how it affects the group outcome.  This is a key feature of level-2 strategic reasoning, where
the player assumes others are likely to be level-1 and play their signal, and thus adjusts their own strategy
accordingly.
Classification:  2
```

Figure 56: Level-k I - Example Message 7

```
I suggest we vote red.  There is a higher chance that the urn is red.  if even one person chooses blue - the
committee's decision is blue, so our decision won't make a difference.
```

	GPT-3.5	GPT-4
O	2	2
O_+	2	2
RAs	1	

Table 43: Human and GPT Classifications of message in Figure 56 using n-shot and CoT prompting

Figure 57: Level-k I - Classifications with CoT and n-shot for Prompts O and O_+ for message in Figure 56

```
                                                                                   GPT-3.5
Output using O
Reasoning:
The player suggests voting red because they believe there is a higher chance that the urn is red.  They also
consider the scenario where even one person chooses blue, in which case the committee's decision will be blue,
making their decision not impactful.
Classification: 2
Output using O+
Reasoning:
The player suggests voting red based on the higher chance of the urn being red.  They also consider the potential
committee's decision, stating that if even one person chooses blue, the committee's decision will be blue, so
their decision won't make a difference.
Classification: 2


                                                                                   GPT-4
Output using O
Reasoning:
The player acknowledges the rules of the game, specifically the impact of a single blue vote on the committee's
decision.  They are considering the strategic implications of their vote in the context of the group's outcome.
This indicates that they are thinking about the potential votes of the other teams and how to vote accordingly.
The player is not just following their own signal but is also considering the likelihood of others voting blue and
how it affects the final decision.  This suggests that the player is at least at level-2, as they are considering
the best response given that others are most likely playing their signal.
Classification: 2
Output using O+
Reasoning:
The player is considering the rules of the game, specifically the voting mechanism where a single blue vote can
determine the group's decision.  They are also acknowledging the likelihood of the urn being red and the strategic
implications of their vote in relation to the votes of others.  This indicates that the player is thinking about
the potential votes of other teams and how to vote accordingly, which aligns with the characteristics of a level-2
player.
Classification: 2
```

C.4. Additional Analysis

Weighted F1 scores are presented in Table 44. The level-k distribution shown in Table 11 is a standard distribution commonly observed in similar studies and represents the expected distribution when researchers classify such data (Camerer et al., 2004; Costa-Gomes and Crawford, 2006; Burchardi and Penczynski, 2014; Crawford et al., 2013). Given this, weighted F1 is the most relevant metric for assessing the models' performance, and F1 macro averages are omitted.

		no-CoT		CoT	
		0-$Shot$	n-$Shot$	0-$Shot$	n-$Shot$
GPT-3.5	O	.694	.639	.651	.641
	O_+	.653	.757	.667	.693
GPT-4	O	.797	.843	.792	**.913**
	O_+	.768	.789	.744	.895

Table 44: Weighted F1 of Level Classification

D. Level-k II

D.1. Original Instructions

Figure 58: Original Instructions - Part 1

```
Classification Instructions
Thank you for participating in this experiment.  In this section you find instructions as to how this experiment
works.  To take part in the experiment, we assume that you are familiar with the level-k model as it has been
introduced by Nagel (1995) and also with the concept of team reasoning as it has been introduced by Schelling
(1960).  In the experiment, subjects play pure coordination games with symmetric and asymmetric payoffs.  We
assume that you are familiar with the concept of coordination games as they have been carried out by Crawford,
Gneezy and Rottenstreich (2008).
However, in order to clarify potential questions of terminology, we reproduce the main features of the level-k
model and the concept of team reasoning.  In addition we provide detailed experimental instructions, which explain
the game and also give you a short introduction to coordination games.  Please read all information carefully in
order to know how the original experiment proceeded.
Experimental Setting
Introduction
This section describes the main features of the experiment.  Subjects are randomly assigned into teams of two
players.  For a given strategic situation, each player makes suggestions for the team action at two points in
time.  First, the so-called "suggested decision" and a justifying written message are exchanged between the
team partners simultaneously.  After this, the "final decision" is taken individually by each team player.  The
computer chooses randomly one of the two final decisions to obtain the "team's action."
All teams play a series of eight coordination games.  Coordination games are characterised by situations in which
all parties can realize mutual gains, but only by making mutually consistent decisions.  Each team is randomly
matched with another team.  If a matched pair of teams both decide on identical team actions, they coordinate
their behavior successfully and are rewarded with a payoff.
However, if both teams choose different team actions, they fail to coordinate their behavior and do not receive
any payoff.  Thus both teams are motivated solely to coordinate their strategies in order to obtain an outcome
that is best for them.  The following example illustrates a random coordination game in which each team decides
on one strategy X, Y or Z simultaneously.  Only if both teams make mutually consistent decisions they receive a
payoff of 2 units each.


                                      Team 2
                                X        Y        Z
                       X      (2,2)    (0,0)    (0,0)
             Team 1    Y      (0,0)    (2,2)    (0,0)
                       Z      (0,0)    (0,0)    (2,2)


The payoff is represented through an experimental currency unit ("Taler").  One Taler is worth 0,40 Euro.  In a
symmetric coordination game each team is rewarded the same payoff if they coordinate their behavior successfully.
In asymmetric coordination games players usually disagree on which action they prefer to coordinate.  There may be
one outcome where one team disproportionately benefits in comparison to the other team.
X-Y Coordination Games
All subjects face a series of eight coordination games composed of four "X-Y Games" and four "Pie Games".  We
reproduce the main features and attributes of those games in the following.  "X-Y Games" are characterised by a
binary choice option "X" or "Y".  The assignment of payoffs for successful coordination is indicated in brackets.
Example:
X [6 Taler for Team 1 and 5 Taler for Team 2]
Y [5 Taler for Team 1 and 6 Taler for Team 2]
If a matched pair of teams both decide on the identical team action "X", team one receives 6 Taler and team two
receives 5 Taler.  If both teams chose "Y", the assignment of payoffs would be reversed.  If both teams chose
decisions with different labels "X" and "Y", neither team receives any payoff.  The payoff differences vary within
the four "X-Y treatments".
Pie Coordination Games
"Pie Games" are characterised by a visual representation of different choice options as indicated in the following
figure.  Each team simultaneously selects one of the three "pie slices".  Each slice is labeled with an abstract
decision label §, $ or #.  The assignment of payoffs for successful coordination is indicated in brackets within
the three slices.  The first number represents the quantity of Taler for team one, the second number the quantity
of Taler for team two.
--Image of a Pie with payoffs-- (omitted see van Elten and Penczynski (2020) page 47)
If a matched pair of teams both decide on the identical team action "#", team one receives 7 Taler and team
two receives 6 Taler.  If both teams chose decisions with different labels §, $ or #, neither team receives any
payoff.  The payoff differences alternate within the four "Pie treatments".
Note that the "X-Y Game" and the "Pie Game" might both contain one alternative that is visually distinctive from
another alternative.  For instance, the unshaded bottom slice is visually distinctive from the two upper slices ($
and #) that are shaded in a light grey color.  We refer to a visually distinctive alternative as label-salient.
Moreover an alternative might be payoff-salient in a way that it is distinctive with respect to its payoff
structure.  The concept of label and payoff salience is important for the classification process.
Treatment Overview
We conducted six sessions in Mannheim and three sessions Heidelberg.  All sessions consist of the same eight
treatments (four "X-Y games" and four "Pie games"), however the sequence of treatments in Mannheim is different
from the sequence of treatments in Heidelberg.  The following two tables provide a brief overview over the
sessions conducted in Mannheim (session 1-3, session 7-9 [rounds 7 and 8 moved to the beginning]) and the sessions
conducted in Heidelberg (session 4-6).  The payoff for successful coordination is indicated in brackets.  The
first number represents the quantity of Taler for team one, the second number represents the quantity of Taler for
team two, if both teams coordinate their behavior.
--Table presenting games played in each round in each experiment-- (omitted see van Elten and Penczynski (2020)
page 49)
```

Classification Process
Remember: Each player makes suggestions for the team action at two points in time. First, the so-called "suggested decision" and a justifying written message are exchanged between the team partners simultaneously. After this, the "final decision" is taken individually by each team player. The computer chooses randomly one of the two final decisions to obtain the "team's action." Your task is to classify the written messages into different categories. In the following we will describe the classification process for the analysis of the experiment.
Level k Model
Notation of the level k model
It is assumed that you are familiar with the level-k model as it has been introduced by Nagel (1995) or represented by Camerer (2004). The model here is extended to incorporate salience in the level-0 belief according to Bacharach and Stahl (2000). In order to clarify potential questions of terminology and introduce the main features of the model we quickly reproduce the main features of the model in the terminology used in this document. The level-k model of bounded rationality assumes that players only think through a certain number (k) of best responses. The model has four main ingredients:
Population distribution: This distribution reflects the proportion of types with a certain level $k \in N_0 = \{0, 1, 2, 3, 4, 5, ...\}$.
Level-0 distribution: By definition, a level-0 player does not best respond. Hence, his actions are random to the game and distributed randomly over the action space. In our case, the action space is $A = \{\{X\}, \{Y\}\}$ or $A = \{\{§\}, \{\$\}, \{\#\}\}$. The model incorporates salience by assuming higher probabilities in the level-0 distribution for actions that are visually distinctive (salient). An action might be salient in terms of payoffs and in terms of labels. In the "X-Y" treatments, the level-0 distribution would not assign a uniform probability of 0.5 to each possible action, but $p > 0.5$ to the salient one and $q_i < p$ for the remaining actions. In the "Pie" treatments, the level-0 distribution would not assign a uniform probability of 1/3 to each possible action, but $p > 1/3$ to the salient one and $q_i < p$ for the remaining actions.
Level-0 belief: In the model, the best responses of players with $k > 0$ are anchored in what they believe the level-0 players play. Their level-0 belief might not be consistent with the level-0 distribution. For best responding, all that matters is the expected payoff from choosing an action from the action space $A = \{\{X\}, \{Y\}\}$ or $A = \{\{§\}, \{\$\}, \{\#\}\}$. A subject would therefore decide on a particular action, when the probability is highest, that the other team chooses the same action.
Population belief: Players do not expect other players to be of the same or a higher level of reasoning. For a level-k player, the population belief is therefore defined on the set of levels strictly below k. It follows that level-0 players have no defined belief, level-1 players have a trivial belief with full probability mass on $\{0\}$, level-2 players have a well defined belief on $\{\{0\}, \{1\}\}$. From level 3 higher order beliefs are relevant as level-3 players have to form a belief about level-2's beliefs.
Characterisation of the different levels
Level 0 The player does not exhibit any strategic reasoning whatsoever. Different versions of this might be randomly chosen or purely guessed actions, misunderstanding of the game structure or other non-strategic 'reasons' for picking a location, e.g. by taste or salience. It is important that no best-responding to the other's play occurs. There could be considerations of what others might play, but without best responding to it. Examples: "Well, it's a pure guess", "There are no arguments. Simply choose any."
Level 1 This player best responds to some belief about the other teams' action. However, he does not realise that others will be strategic as well. Example: "They are probably picking X, so we do as well", "The other team would naturally go for the visual distinctive bottom slice, no?"
Level 2 This player not only shows the basic strategic consideration of playing best response (matching/mismatching), but also realises that other players best respond as well according to the belief they entertain. A level-2 player clearly contemplates how the other player might best respond to his frame. The player plays a best response to this hypothesised consideration. Example: "The other team may think we are most attracted to the alternative # with the highest payoff. In order to coordinate our behavior we should also choose the # slice."
Level 3 This player realises that others could be level-2 and reacts by best responding to the associated expected play. Put differently, he realises that others realise that others best respond to their initial belief. Therefore, a level-3 player clearly states that his opponent expects that he (the level-3 player at question) best-responds to a certain belief.
Level 4, 5, ... The process goes on in a similar fashion. A level k player realises that other subjects could be level k-1 and reacts by best responding to the associated expected play.
Category 1: Lower and upper bound on the level of reasoning
Your aim
is to classify the written messages into the underlying level $k \in N_0 = \{0, 1, 2, 3, 4, 5, ...\}$ of reasoning. For a given statement it might not be possible to exactly determine the underlying level of reasoning. To extract as much information as possible, we ask you to indicate a lower and an upper bound on the level of reasoning.
For the lower bound on the level of reasoning, you should ask yourself: "What is the minimum level of reasoning that this statement clearly exhibits?" Once noted, you should be able to say to yourself: "It seems impossible that the players' level of reasoning is below this number!" Here we ask you to be very cautious with the classification, not giving away high levels easily.
The upper bounds should give the maximum level of reasoning that could be interpreted into the statement. Therefore, you should ask yourself: "What is the highest level of reasoning that can be underlying this statement?" Once noted, you should be able to say: "Although maybe not clearly communicated, this statement could be an expression of this level. If the player reasoned higher than this number, this was not expressed in the statement!" For both lower and upper bound, please refer to the characterisation of the different levels.
There are two necessary conditions for a player to exhibit a level greater than 0. First, the player has to be responsive to the salience of the games' framing. Secondly, the player has to be strategic in best-responding to his level-0 belief, which is shaped by label or payoff salience. If he did not react to salience, he would have no reason to chose one over the other object, resulting in random level 0 play.
For this category, the excel-sheet for the classification will feature a drop-down menu where you can choose upper and lower bounds between 0 and 5. If no inference can be made since nothing or nothing to the point is written, you can choose not applicable (n/a).
Category 2: Level-0 belief
Your aim
is to indicate the underlying level-0 belief that is connected with the lowest possible level of reasoning. If level reasoning is observed in the statement, there has to be a starting point in the argument which states an attraction or aversion to one alternative. This is then not derived by strategic reasons, but is an intuitive reaction to the framing of the coordination game.
Otherwise, level reasoning would not occur. Please indicate the underlying level-0 belief that is connected with the lowest possible level of reasoning. Note that the level-0 belief of a person reasoning on an odd level, i.e. level 1, 3, 5, etc. is always with respect to how a player of the opposite side intuitively reacts to the framing. The belief of a person reasoning on an even level, i.e. level 2, 4 etc. is always with respect to what the opposite type believes about the own typeâs intuitive reaction.

Figure 60: Original Instructions - Part 3

There are two kinds of framing in these games. On the one hand, subjects might react to the framing of the
coordination game (label salience). Imagine a subject that you classify to be level-1. It might communicate
that the other team is most attracted to the visual distinctive white bottom slice § and therefore proposes §̂ as
team decision. A subject that you classify to be level-2 might indicate that the other team believes that one's
own team is more likely to choose "X", because this alternative is mentioned first on the screen. To reflect
a level-0 belief of an attraction to X or Y , or to ♯, §, or $, the excel-sheet features a drop-down menu that
allows to indicate such a preference or an indifference. If such a preference or indifference over labels is not
indicated, or if the subjectsâ level of attractiveness cannot be distinguished or is not expressed clearly within
the message, please indicate that the level-0 belief from the message does not exhibit any label salience.
On the other hand, subjects might respond to the payoffs (payoff salience). For example, consider a subject that
you classify to be level-1. It might communicate that the other team is most likely to choose alternative X as
it offers the highest payoff to this very team. Or, a subject that you classify to be level-2 might indicate
that the other team remains of the conviction that oneâs own team is not attracted to the action that gives oneâs
own team high payoffs. To reflect the exhibited level-0 beliefs you can indicate in the excel-sheet whether the
team that the level-0 belief is formed about is believed to be attracted to a) the action that yields --under
coordination-- a higher payoff for this team, to b) the action that yields --under coordination-- a higher payoff
to the other team or c) is indifferent. If no such preference or indifference over salient payoff actions is
indicated, please indicate that the level-0 belief from the message does not exhibit any payoff structure.
Please note that payoff and label salience are not mutually exclusive, please indicate both if both is expressed
in the message. Finally, for players whose lower bound is 0, the level-0 belief classification can be used to
indicate whether a level-0 player states for his action a preference with respect to label or payoff salience.

Classification Summary
In coordination games both teams are motivated solely to coordinate their strategies in order to obtain an outcome
that is best for them. For a given strategic situation, each player proposes a suggested decision and writes
a justifying written message to the team partner. Your task is to classify the written messages into different
categories that are summarized in the following:
Category 1 Please classify the written messages into the underlying level $k \in N_0 = \{0, 1, 2, 3, 4, 5, ...\}$ of reasoning.
Provide the lower and an upper bounds on the level of reasoning as described.
Category 2 Please indicate the underlying level-0 belief that is connected with the lowest conceivable level of
reasoning. Information about the underlying level-0 belief that one might obtain out of the communication is
how subjects respond to payoffs (payoff salience) and how subjects react to the framing (label salience) of the
coordination games.

D.2. Prompts

Figure 61: Prompts O - Part 1

```
# General Task
- Evaluate player's message from a specific coordination game to identify their decision-process and to classify
their level of strategic thinking.
# Role Persona
- Act as a behavioral economist specialized in coordination games, decision salience and text analysis.
# Context
## Game Mechanics
- Subjects (players) participate in an experiment where they play a coordination game.
- Players are assigned into teams of two.
- Each team is matched with another team to play a series of coordination games where the teams try to coordinate
on a specific alternative.
- If both teams pick the same alternative, each team is rewarded with a payoff.  Otherwise, neither team receives
any payoff.
- Each player sends a suggested decision and a justifying message to their teammate.
- Coordination game is not played within the team members but between the two teams.
- There is no communication between the teams.
- Payoffs are represented in currency called Taler.
- 1 Taler = 40 cents (ct).
- Each player plays a series of 8 (rounds of) coordination games, split between *X-Y* games and *Pie* games.
## Coordination Games
- Payoff tables for each variation of each game is represented below.
- In each payoff table, 'd' represents decision taken, 'π1' represents payoff of team 1, and 'π2' represents
payoff of team 2.
- Payoff tables only represent the cases where both teams match in the given decision 'd', if teams' decisions do
not match, each team receives 0 Taler.
### Pie Game
- Payoffs are displayed on a pie chart that is divided into three equally sized segments.
- Top left segment is labeled as '$'.
- Top right segment is labeled as '#'.
- Top segments are shaded in gray.
- Bottom segment is labeled as '§' and is highlighted in white.
- There are 4 payoff variations labeled as S1, S2, AM2 and AM4.
#### S1
| d      |π1, π2|
|-------|----- |
| L ($) | 5, 5 |
| R (#) | 5, 5 |
| B (§) | 5, 5 |
#### S2
| d      |π1, π2|
|-------|----- |
| L ($) | 6, 6 |
| R (#) | 6, 6 |
| B (§) | 5, 5 |
#### AM2
| d      |π1, π2|
|-------|----- |
| L ($) | 5, 6 |
| R (#) | 6, 5 |
| B (§) | 6, 5 |
#### AM4
| d      |π1, π2|
|-------|----- |
| L ($) | 6, 7 |
| R (#) | 7, 6 |
| B (§) | 7, 5 |
### X-Y Game
- Alternatives are displayed in two consecutive lines.
- Alternative 'X' is displayed on the first line.
- Alternative 'Y' is displayed on the second line.
- There are 4 payoff variations labeled as SL, ASL, AML and ALL.
#### SL
| d |π1, π2 |
|---|------- |
| X | 5, 5  |
| Y | 5, 5  |
#### ASL
| d |π1, π2 |
|---|------- |
| X | 5, 5.1 |
| Y | 5.1, 5 |
#### AML
| d |π1, π2 |
|---|------- |
| X | 5, 6  |
| Y | 6, 5  |
#### ALL
| d |π1, π2 |
|---|------- |
| X | 5, 10 |
| Y | 10, 5 |
```

```
## Salience in Decisions
### Salient Label
- Both *Pie* and *X-Y* games may contain a decision that is visually distinctive from other available decision(s).
- In the Pie game, the bottom segment (slice) that is highlighted in white in the Pie game is visually distinctive
from the upper slices ($ and #) that are shaded in gray color.
- In the X-Y game, the decision X may be perceived as more salient because it is the top or the first decision
presented.
- We refer to a visually distinctive decision as **label-salient**.
### Salient Payoff
- A decision might be also **payoff-salient** in a way that is distinctive with respect to its payoff structure.
- A decision alternative that provides the highest or lowest payoff for one of the teams can be considered as
payoff salient.
- In ALL variation of the X-Y game, option X may be considered to have high payoff salience for team 1, while
option Y may be considered to have high payoff salience for team 2.
## Level-k Model
### Level-0 Distribution
- Level-0 player does not best respond but instead play according to some probability distribution over the action
space (level-0 distribution).
    + For the X-Y game the action space is X,Y
    + For the Pie game the action space is $,#,§
- Without any salient actions, a level-0 player's actions distributed randomly and evenly over the action space.
- The model incorporate salience by assuming higher probabilities for either payoff salience or label salience
actions in the level-0 player's action space distribution.
   + In the X-Y game, level-0 distribution would not assign equal probability of 0.5 to each action but instead
assign p > 0.5 to the salient action and and q < 0.5 for the non-salient action.
    + In the Pie game, level-0 distribution would not assign equal probability of 1/3 to each possible action, but
p > 1/3 to the salient action and q < 1/3 for the non-salient actions.
### Level-0 Belief
- The best response of a level-k (k > 0) player is anchored in what he believes a level-0 player plays. This is
called the level-0 belief of the level-k player.
- A level-k player performs k many iterative best responses and always starts its iterative reasoning from his
level-0 belief.
### Population Belief
- Players do not expect other players to be of the same or higher level of reasoning. For a level-k player, the
population belief is therefore defined on the set of levels strictly below k.
- A level-0 player has no defined population belief.
- A level-1 player has a trivial belief with full probability on all other players being level-0.
- A level-2 player has a well defined belief distribution on all other players being level-1 and level-0.
- A level-3 player has a well defined belief distribution on all other players being level-2, level-1 and level-0
(and so on for higher order of levels of thinker).
### Characterization of different levels
#### Level-0
- A level-0 player does not exhibit any strategic reasoning.
- A level-0 player may be randomly choosing or purely guessing an action.
- A level-0 player may misunderstand the game structure.
- A level-0 player may pick an action for non-strategic reasons such as taste or salience.
- A level-0 player does not best respond to other players potential actions.
##### Examples
- "Well, it's a pure guess"
- "There are no arguments. Simply choose any."
#### Level-1
- A level-1 player best responds to some belief about the other teams' action, but he does not consider that the
other teams may be strategic as well.
##### Examples
- "They are probably picking X, so we do as well"
- "The other team would naturally go for the visually distinctive bottom slice"
#### Level-2
- A level-2 player does not only show the basic strategic consideration of playing best response to a his level-0
belief, but also recognizes that other players may best respond as well according to their level-0 belief.
- A level-2 player clearly contemplates how the other player might best respond to his frame. The player plays a
best response to this hypothesized consideration.
##### Example
- "The other team may think we are most attracted to the alternative with the highest payoff. In order to
coordinate our behavior, we should also choose the  slice."
#### Level-3
- A level-3 player realizes that the other players (team) could be level-2 and best-responds accordingly
- A level-3 player realizes that others realize that others best-respond to their level-0 belief.
- A level-3 player states that his opponent expects that he (the level-3 player at question) best-responds to a
certain belief.
#### Level-4,5,...
- The process goes on in a similar fashion
- A level-k player realizes that other subjects could be level-(k-1) and reacts by best responding to the
associated expected play.
```

Figure 63: Prompts O - Part 3 - Classification Task

Classification Task
Task 1: Level-0 Belief
- Your task is to indicate the underlying level-0 belief that is connected with the lowest possible level of reasoning.
- If the level of reasoning is observed in the message, there has to be a starting point in the argument which states an attraction or aversion to one alternative. This is then not derived by strategic reasons, but is an intuitive reaction to the framing of the coordination game. Otherwise, level reasoning would not occur.
- Information about the underlying level-0 belief that one might obtain out of the communication is how players respond to payoffs (payoff salience) and how players react to framing (label salience) of the coordination games.
- Level-0 belief of a person reasoning on an odd level (level 1,3 or 5) is always with respect to how a player of the opposite side intuitively reacts to the framing (salience).
- Level-0 belief of a person reasoning on an even level (level 2 or 4) is always with respect to what the opposite type believes about the own type's intuitive reaction.
Salience
- There are two kinds of salience in these games: label and payoff
- Payoff and label salience are not mutually exclusive. A player may display both payoff and label salience within the same reasoning.
- For players whose lower bound is 0, the level-0 belief classification can be used to indicate whether a level-0 player states for his action a preference with respect to label or payoff salience.
Label
- Players may react to the framing of the coordination game (label salience)
- A player that you classify to be level-1 may communicate that the other team is most attracted to the visual distinctive white bottom slice §and therefor proposes §as team decision.
- A player that you classify to be level-2 may indicate that the other team believes that one's own team is more likely to be choose 'X' because this alternative is mentioned first on the screen.
- For X-Y game, classify the label salience of the message as one of the following:
  + prefers 'X' over 'Y' (a level-0 belief of an attraction to 'X')
  + prefers 'Y' over 'X' (a level-0 belief of an attraction to 'Y')
  + is indifferent between labels 'X' and 'Y'
  + does not exhibit label salience
- For Pie game, classify the label salience of the message as one of the following:
  + prefers '$' (a level-0 belief of an attraction to '$')
  + prefers '#' (a level-0 belief of an attraction to '#')
  + prefers '§' (a level-0 belief of an attraction to '§')
  + is indifferent across labels
  + does not exhibit label salience
Payoff
- Players respond to the payoffs' salience.
- A player you classify to be level-1 may communicate that the other team is most likely to choose alternative 'X' as it offers the highest payoff to this very team.
- A player you classify to be level-2 may indicate that the other team remains of the conviction that one's own team is not attracted to the action that gives one's own team high payoffs.
- For either game, classify the payoff salience of the message as one of the following:
  + prefers high payoffs (a level-0 belief of an attraction to high payoff)
  + prefers low payoffs (a level-0 belief of an attraction to low payoff)
  + is indifferent across payoffs
  + does not exhibit payoff salience
Task 2: Level of Strategic Thinking
- Your task is to classify the written messages into the underlying level-k of reasoning.
- For a given statement it might not be possible to exactly determine the underlying level of reasoning.
- To extract as much information as possible, you are asked to indicate a lower and an upper bound on the level of reasoning
- Use the information provided under the subsection "Characterization of different levels" in the "Context" section above to classify lower and upper bounds on the level of reasoning.
Lower bound
- Ask yourself: "What is the minimum level of reasoning that this statement clearly exhibits?". Once noted, you should be able to state to yourself: "It seems impossible that the players' level of reasoning is below this number!"
- Be very cautious with the classification of the lower bound.
- Do not give away high levels easily.
Upper bound
- The upper bounds should give the maximum level of reasoning that could be interpreted into the statement.
- You should ask yourself: "What is the highest level of reasoning that can be underlying this statement?". Once noted, you should be able to say: "Although maybe not clearly communicated, this statement could be an expression of this level. If the player reasoned higher than this number, this was not expressed in the statement!"
Necessary conditions for a player to exhibit a level of thinking greater than 0:
1. The player has to be responsive to the salience of the games' framing (either payoff or label salience)
2. The player has to be strategic in best-responding to his level-0 belief, which is shaped by label or payoff salience. If he did not react to salience, he would have no reason to choose one over the other action, resulting in random level 0 play.

Figure 64: Prompts O - Part 4

```
# Classification Coding
## Label Salience
### X-Y Game
- Code as `X` if player's label salience is "prefers X over Y"
- Code as `Y` if player's label salience is "prefers Y over X"
- Code as `~` if player's label salience is "indifferent across payoffs"
- Code as `no` if the player does not exhibit payoff salience.
### Pie Game
- Code as `$` if player's label salience is "prefers $"
- Code as `#` if player's label salience is "prefers #"
- Code as `§` if player's label salience is "prefers §"
- Code as `~` if player's label salience is "indifferent across labels"
- Code as `no` if the player does not exhibit label salience.
## Payoff Salience
- Code as `H` if player's payoff salience is "prefers high payoffs"
- Code as `L` if player's payoff salience is "prefers low payoffs"
- Code as `~` if player's payoff salience is "indifferent across payoffs"
- Code as `no` if the player does not exhibit payoff salience.
## Upper and Lower Bounds
- 0,1,2,3,4 or 5.
# Input Format
    Team:
    Game:
    Decision:
    Message:
# Constraint
- Follow the below output format
# Output Format
    Label Salience:
    Payoff Salience:
    Lower Bound:
    Upper Bound:
```

Figure 65: Prompts O_+ - Examples

```
# Examples
- "X is first, let's pick X" (Level-0 belief: prefers X. Label salience: X. Payoff salience: no. Level: 0.)
- "$is highlighted in white, hence §" (Level-0 belief: prefers §. Label salience: §. Payoff salience: no.
Level: 0.)
- "Y provides a higher payoff, let's go X" (Level-0 belief: prefers higher payoff. Label salience: no. Payoff
salience: higher payoff. Level: 0.)
- "It is random. You pick X and I pick Y." (Level-0 belief: random play. Label salience: no. Payoff salience:
no. Level: 0.)
- "Other team may pick X as it is on top, let's pick X" (Level-0 belief: prefers X. Label salience: X. Payoff
salience: no. Level: 1.)
- "X is first, let's pick X. And other team may think the same way." (Level-0 belief: prefers X. Label salience:
X. Payoff salience: no. Level: 1.)
- "People will go for the highlighted segment. Hence §" (Level-0 belief: prefers §. Label salience: §. Payoff
salience: no. Level: 1.)
- "Other team may want to the high payoff for themselves, let's coordinate with them and pick the higher payoff
for them (which is the lower payoff for us)." (Level-0 belief: prefers higher payoff. Label salience: no.
Payoff salience: higher payoff. Level: 1.)
- "Others will think we go for top. So let's go for top." (Level-0 belief: prefers X. Label salience: X. Payoff
salience: no. Level: 2.)
- "The other team will think we pick the highlighted segment. So we should coordinate and pick §" (Level-0
belief: prefers §. Label salience: §. Payoff salience: no. Level: 2.)
- "Other team may think that we want the high payoff for us, let's coordinate with them and pick the higher payoff
for us." (Level-0 belief: prefers higher payoff. Label salience: no. Payoff salience: higher payoff. Level:
2.)
- "Other team may want to the high payoff for themselves. But they may assume the same thing about us and pick
the alternative that gives us the higher payoff. So let's coordinate with them and pick the higher payoff for
us (which is the lower payoff for them)." (Level-0 belief: prefers higher payoff. Label salience: no. Payoff
salience: higher payoff. Level: 2.)
- "Others will think that we think that they will go for top. So let's go for top." (Level-0 belief: prefers X.
Label salience: X. Payoff salience: no. Level: 3.)
- "The other team will think that we think that they pick the highlighted segment. So we should coordinate and
pick §" (Level-0 belief: prefers §. Label salience: §. Payoff salience: no. Level: 3.)
- "Other team may think that we want the high payoff for us, let's coordinate with them
and pick the higher payoff for them." (Level-0 belief: prefers higher payoff. Label salience: no. Payoff
salience: higher payoff. Level: 3.)
- "Other team may think that we want the high payoff for us. But they may assume the same thing about us and pick
the alternative that gives us the higher payoff for themselves. So let's coordinate with them and pick the higher
payoff for them." (Level-0 belief: prefers higher payoff. Label salience: no. Payoff salience: higher payoff.
Level: 3.)
```

```
# General Task
- Evaluate player's message from a specific coordination game to identify their decision-process and to classify
their level of strategic thinking.
# Role Persona
- Act as a behavioral economist specialized in coordination games, decision salience and text analysis.
# Context
## Game Mechanics
- Subjects (players) participate in an experiment where they play a coordination game.
- Players are assigned into teams of two.
- Each team is matched with another team to play a series of coordination games where the teams try to coordinate
on a specific alternative.
- If both teams pick the same alternative, each team is rewarded with a payoff.  Otherwise, neither team receives
any payoff.
- Each player sends a suggested decision and a justifying message to their teammate.
- Coordination game is not played within the team members but between the two teams.
- There is no communication between the teams.
- Payoffs are represented in currency called Taler.
- 1 Taler = 40 cents (ct).
- Each player plays a series of 8 (rounds of) coordination games, split between *X-Y* games and *Pie* games.
## Coordination Games
- Payoff tables for each variation of each game is represented below.
- In each payoff table, 'd' represents decision taken, 'π1' represents payoff of team 1, and 'π2' represents
payoff of team 2.
- Payoff tables only represent the cases where both teams match in the given decision 'd', if teams' decisions do
not match, each team receives 0 Taler.
### Pie Game
- Payoffs are displayed on a pie chart that is divided into three equally sized segments.
- Top left segment is labeled as '$'.
- Top right segment is labeled as '#'.
- Top segments are shaded in gray.
- Bottom segment is labeled as '§' and is highlighted in white.
- There are 4 payoff variations labeled as S1, S2, AM2 and AM4.
#### S1
| d      |π1, π2|
|-------|------ |
| L ($) | 5, 5  |
| R (#) | 5, 5  |
| B (§) | 5, 5  |
#### S2
| d      |π1, π2|
|-------|------ |
| L ($) | 6, 6  |
| R (#) | 6, 6  |
| B (§) | 5, 5  |
#### AM2
| d      |π1, π2|
|-------|------ |
| L ($) | 5, 6  |
| R (#) | 6, 5  |
| B (§) | 6, 5  |
#### AM4
| d      |π1, π2|
|-------|------ |
| L ($) | 6, 7  |
| R (#) | 7, 6  |
| B (§) | 7, 5  |
### X-Y Game
- Alternatives are displayed in two consecutive lines.
- Alternative 'X' is displayed on the first line.
- Alternative 'Y' is displayed on the second line.
- There are 4 payoff variations labeled as SL, ASL, AML and ALL.
#### SL
| d |π1, π2 |
|---|------- |
| X | 5, 5   |
| Y | 5, 5   |
#### ASL
| d |π1, π2 |
|---|------- |
| X | 5, 5.1 |
| Y | 5.1, 5 |
#### AML
| d |π1, π2 |
|---|------- |
| X | 5, 6   |
| Y | 6, 5   |
#### ALL
| d |π1, π2 |
|---|------- |
| X | 5, 10  |
| Y | 10, 5  |
```

```
## Salience
- Label salience:  players may react to the framing of alternatives.
- Payoff salience:  Players may react to the payoff differences of alternatives .
## Level-k Model
- A level-k player performs k many iterative best responses and always starts its iterative reasoning from his
level-0 belief.  This starting point is called the level-0 belief of the level-k player.
- Level-0 belief is the belief of the level-k player on how the level-0 player will potentially play the game.
- A level-0 player picks an alternative for non-strategic, instinctive reasons such as payoff or label salience.
- A level-0 player does not best respond to other players potential actions.
- A level-0 player may be randomly choosing or purely guessing an action.
- A level-0 player may misunderstand the game structure.
- A level-1 player assumes that the other team consists of level-0 players and best responds based on his level-0
belief to these level-0 players.
- A level-k (k>1) player recognizes the possibility that the other team may consist of level-(k-1) players.
- A level-k player assumes that the level-(k-1) players assume he (the level-k player) is a level-(k-2) player.
## Level-0 Belief
- Level-0 belief of a player reasoning on an odd level (level 1,3 or 5) is always with respect to how a player of
the opposite side intuitively reacts to the framing (label or payoff salience).
- Level-0 belief of a player reasoning on an even level (level 2 or 4) is always with respect to what the opposite
type believes about the own type's intuitive reaction (label or payoff salience).
## Lower bound
- The minimum level of reasoning that the message clearly exhibits.
## Upper bound
- The maximum level of reasoning that can be inferred from the message.
# Classification Tasks
- Classify player's label or payoff salience (if any).
- Classify lower and upper bounds for the player's level of reasoning.
# Classification Coding
## Label Salience
### X-Y Game
- Code as 'X' if player's label salience is "prefers X over Y"
- Code as 'Y' if player's label salience is "prefers Y over X"
- Code as '~' if player's label salience is "indifferent across payoffs"
- Code as 'no' if the player does not exhibit payoff salience.
### Pie Game
- Code as '$' if player's label salience is "prefers $"
- Code as '#' if player's label salience is "prefers #"
- Code as '§' if player's label salience is "prefers §"
- Code as '~' if player's label salience is "indifferent across labels"
- Code as 'no' if the player does not exhibit label salience.
## Payoff Salience
- Code as 'H' if player's payoff salience is "prefers high payoffs"
- Code as 'L' if player's payoff salience is "prefers low payoffs"
- Code as '~' if player's payoff salience is "indifferent across payoffs"
- Code as 'no' if the player does not exhibit payoff salience.
## Upper and Lower Bounds
- 0,1,2,3,4 or 5.
# Examples (see Figure 65) (Only used in n-shot treatments)
# Input Format
    Team:
    Game:
    Decision:
    Message:
# Constraint
- Follow the below output format
# Output Format
    Label Salience:
    Payoff Salience:
    Lower Bound:
    Upper Bound:
```

D.3. Additional Examples

Figure 68: Level-k II - Example Message 3

```
Team:   1
Game:   ASL
Decision:   Y
Message:   I think they will propose a larger payout to us because they think we will keep it for ourselves
to ensure coordination...  however, I can imagine that they have the exact same mindset as I do and will do
the opposite...  I'll call it Y
```

	GPT-3.5				GPT-4			
	L_B	U_B	S_L	S_P	L_B	U_B	S_L	S_P
O	1	2	$\sim$	H	1	2	no	H
O_+	1	2	no	H	3	3	no	H
RAs	3	3	no	H	3	3	no	H

Table 45: Human and GPT Classifications of message in Figure 68 using CoT prompting for n-shot O_+ and 0-shot O

Figure 69: Level-k II - Generated rationals for 0-shot O and n-shot O_+ under CoT for message in Figure 68

```
                                                                                          GPT-3.5
Output using O
Reasoning:
The player seems to be considering the other team's potential decision and how it relates to their own team's
payoff.  The player is suggesting that the other team might propose the action that yields a higher payoff for
them, assuming that their team will keep the higher payoff for themselves.  However, the player also acknowledges
that the other team might have the same thought process and propose the opposite action.  This indicates a level
of strategic thinking beyond pure guessing.
Output using O+
Reasoning:
The player seems to be considering the other team's potential strategy and trying to anticipate their decision
based on the belief that the other team may think they want the higher payoff for themselves.  This indicates a
level of strategic thinking beyond just instinctive or random play.


                                                                                            GPT-4
Output using O
Reasoning:
The player's message indicates a complex thought process involving multiple levels of strategic reasoning.  The
player starts by suggesting that the other team will propose the action that gives team 1 the larger payoff, under
the assumption that team 1 will want to keep the larger payoff for themselves.  This is a strategic consideration
and indicates that the player is not operating at level-0, which would involve no strategic reasoning or a random
choice.
The player then goes on to consider that the other team might have the same line of thinking and might, therefore,
choose the opposite action to ensure coordination.  This indicates a higher level of reasoning, as the player is
considering the other team's potential strategy and how they might anticipate team 1's actions.
The player's final decision to suggest 'Y' seems to be a best response to the anticipated strategy of the other
team, which suggests a level-2 reasoning.  The player is considering that the other team is also reasoning
strategically (level-1) and is best responding to that.
The message does not clearly indicate a level-3 reasoning, as the player does not explicitly state that they
believe the other team is considering that team 1 might be level-2 thinkers.  Therefore, the upper bound of the
reasoning level would be level-2.
Regarding the level-0 belief, the player's message indicates a consideration of the payoffs, as they mention
the larger payoff and the desire to keep it.  This suggests a payoff salience in their reasoning.  There is no
explicit mention of label salience in the message.
Output using O+
Reasoning:
The player's message suggests a complex line of reasoning.  The player starts by considering that the other team
will propose the alternative that gives Team 1 the larger payoff, under the assumption that Team 1 would want to
keep the larger payoff for themselves.  This is a strategic consideration that goes beyond a simple preference for
a label or payoff, indicating a higher level of reasoning.
The player then goes on to consider that the other team might have the same line of reasoning and therefore might
do the opposite.  This is an additional layer of strategic thinking, suggesting that the player is trying to
anticipate the other team's anticipation of their own team's actions.
The player concludes by suggesting "Y" as the decision.  Since the game is ASL, where the payoffs are 5 for X and
5.1 for Y for Team 1, and 5.1 for X and 5 for Y for Team 2, the player is suggesting to pick the alternative that
gives the other team the higher payoff, which aligns with the initial part of the reasoning.
The message indicates a level-0 belief that the other team prefers higher payoffs (since they are expected to
propose the alternative that gives Team 1 the larger payoff).  However, the player's own decision does not
directly reflect a preference for higher payoffs for themselves, but rather a strategic move to coordinate with
the other team's expected strategy.
The player's reasoning involves anticipating the other team's anticipation of their own team's actions, which is
characteristic of level-3 reasoning.  The player assumes that the other team is level-2, who in turn assumes that
Team 1 is level-1.
```

Figure 70: Level-k II - Example Message 4

```
Team:  1
Game:  ALL
Decision:  X
Message:  We also need to be selfish sometimes.
```

	GPT-3.5				**GPT-4**			
	L_B	U_B	S_L	S_P	L_B	U_B	S_L	S_P
O	1	1	*no*	*H*	0	0	*no*	*H*
O_+	0	1	*no*	*H*	0	0	*no*	*H*
RAs	0	0	*no*	*no*	0	0	*no*	*no*

Table 46: Human and GPT Classifications of message in Figure 72 using CoT prompting for n-shot O_+ and 0-shot O

Figure 71: Level-k II - Generated rationals for 0-shot O and n-shot O_+ under CoT for message in Figure 72

```
                                                                          GPT-3.5
Output using O
Reasoning:
The player's message suggests that they believe their team should be selfish.  This indicates a preference for
the action X, which yields a higher payoff for their team.  The player is exhibiting payoff salience by being
attracted to the action that provides a higher payoff for their team.
Output using O+
Reasoning:
The player's message suggests a focus on being selfish, which indicates a potential preference for the higher
payoff.

                                                                          GPT-4
Output using O
Reasoning:
The message "Wir müssen auch mal egoistisch sein," which translates to "We also have to be selfish sometimes,"
suggests that the player is considering the payoffs for their own team.  In the context of the ALL game, the
decision 'X' yields a higher payoff for Team 1 (5 Taler for Team 1 and 10 Taler for Team 2).  By choosing 'X', the
player is indicating a preference for the action that benefits their own team more, which is consistent with the
message's emphasis on being selfish.
Given the message, the player is not making any explicit reference to the strategic considerations of what the
other team might do or how the other team might perceive their team's actions.  The message does not indicate any
iterative reasoning about the other team's potential level of thinking.  Therefore, the player's reasoning seems
to be at a basic strategic level, where they are considering their own payoffs without explicitly considering the
other team's response.
The message does not mention anything about the label salience, such as the order of the options or any visual
distinction between them.  Therefore, we cannot infer any label salience from this message.
Output using O+
Reasoning:
The message "Wir müssen auch mal egoistisch sein" translates to "We also have to be selfish sometimes." This
message suggests that the player is considering their own interests, which in the context of the ALL game, where
the payoffs are asymmetric (5 Taler for one team and 10 Taler for the other), implies a focus on the higher payoff
for themselves.  The message does not provide any indication of label salience, as it does not reference the
labels X or Y. The level of reasoning is not explicitly stated, but the use of "egoistisch" (selfish) suggests a
level-0 belief that prefers higher payoffs.  Since the message does not indicate any anticipation of the other
team's reasoning or actions, it is best classified as level-0 reasoning.
```

```
Team:  1
Game:  ASL
Decision:  X
Message Hmm, tricky because hardly any difference - I'll be generous - alternative X
```

	GPT-3.5				GPT-4			
	L_B	U_B	S_L	S_P	L_B	U_B	S_L	S_P
O	0	0	no	no	0	1	no	$\sim$
O_+	0	1	no	$\sim$	1	1	no	L
RAs	0	1	no	H	0	1	no	H

Table 47: Human and GPT Classifications of message in Figure 72 using CoT prompting for n-shot O_+ and 0-shot O

Figure 73: Level-k II - Generated rationals for 0-shot O and n-shot O_+ under CoT for message in Figure 72

GPT-3.5

Output using O
Reasoning:
The player's message indicates that there is hardly any difference between the options and he is being generous by choosing alternative X. The message does not explicitly exhibit any strategic reasoning or salience. The player seems to be randomly choosing an option without any clear indication of salience or strategic thinking.
Output using O_+
Reasoning:
The player seems to be considering the small difference in payoffs and is leaning towards being generous by choosing alternative X.

GPT-4

Output using O
Reasoning:
The message provided by the player from Team 1 in the ASL game, where they have chosen decision X, is brief and suggests a level of indifference or difficulty in distinguishing between the options due to the small difference in payoffs ("kaum Unterschied"). The use of the word "großzügig" (generous) implies that the player is aware of the slight asymmetry in the payoffs but decides to not prioritize this difference. The player's choice of "Alt. X" (alternative X) does not provide a clear strategic rationale for the decision.
The player does not mention any aspect of the label salience in their message. There is no indication that the player prefers X over Y due to its position or any other visual feature. Therefore, the player does not exhibit label salience.
The player acknowledges the small difference in payoffs but does not express a clear preference for the higher payoff. The use of "großzügig" could suggest a willingness to not focus on the slight advantage in payoffs for their team, which could be interpreted as a preference for low payoffs. However, since the player ultimately chooses X, which for the ASL game would provide a slightly higher payoff for their own team (5 Taler for Team 1 vs. 5.1 Taler for Team 2), it is not clear if the player is indeed being generous to the other team or simply indifferent to the payoff difference. Therefore, the player's message could be interpreted as showing indifference across payoffs.
The player's message does not clearly exhibit strategic reasoning. The player does not articulate a belief about what the other team might choose or how their own choice might influence the other team's decision. The message seems to reflect a random or non-strategic choice, which aligns with a level-0 player. However, the acknowledgment of the small difference in payoffs and the use of "großzügig" could suggest a minimal level of strategic consideration, potentially indicating a level-1 reasoning where the player is aware of the payoffs but does not articulate a strategic response to them.
Given the lack of clear strategic reasoning and the absence of a stated belief about the other team's likely action, the minimum level of reasoning that this statement clearly exhibits is level-0.
The highest level of reasoning that could be underlying this statement, given the player's acknowledgment of the payoff difference, could be level-1. The player does not express any higher-order strategic considerations, such as anticipating the other team's response to their own decision or considering multiple levels of reasoning.
Output using O_+
Reasoning:
The message "Hmm, fies weil kaum Unterschied – ich bin mal großzügig – Alt. X" translates to "Hmm, tricky because there's hardly any difference – I'll be generous – Alt. X". This message indicates that the player is aware of the small difference in payoffs between the alternatives X and Y in the ASL game, where the payoffs are 5 Taler for the player's team and 5.1 Taler for the other team for alternative X, and vice versa for alternative Y. The use of the word "generous" suggests that the player is willing to choose the option that gives the other team a slightly higher payoff, which is alternative X for the ASL game.
The player does not express a preference for a label (X or Y) but rather comments on the small difference in payoffs, indicating a recognition of payoff salience. However, the player does not explicitly state a preference for higher or lower payoffs, just that they are being "generous" by choosing the alternative that gives the other team a slightly higher payoff. This could be interpreted as a preference for lower payoffs for themselves (since they are giving the other team the higher payoff), but it is not a strong indication of such a preference.
The player's level of reasoning seems to be at least level-1 because they are considering the other team's potential payoff and are making a decision based on that consideration. However, there is no clear indication that the player is thinking beyond this, such as considering what the other team might think they will do. Therefore, the upper bound of reasoning is also level-1.

Figure 74: Level-k II - Example Message 6

```
Team:     2
Game:     S2
Decision: §
Message:  Even if § is the lower payout, it is the only one where both teams get a clear solution, so they
          don't have to hesitate!
```

	GPT-3.5				**GPT-4**			
	L_B	U_B	S_L	S_P	L_B	U_B	S_L	S_P
O	1	2	§	$\sim$	0	1	§	L
O_+	1	2	§	no	1	1	no	L
RAs	0	1	§	no	0	1	§	no

Table 48: Human and GPT Classifications of message in Figure 74 using CoT prompting for n-shot O_+ and 0-shot O

Figure 75: Level-k II - Generated rationals for 0-shot O and n-shot O_+ under CoT for message in Figure 74

```
                                                                                       GPT-3.5
Output using O
Reasoning:
The player seems to be considering the fact that § is the only option where both teams can get a clear solution
without any ambiguity.  This indicates that the player is considering the clarity of the decision and the ease of
coordination rather than focusing solely on the payoff.
Output using O+
Reasoning:
The message indicates that the player is considering the unique solution where both teams can make a clear
decision without hesitation.  This suggests a level of strategic thinking beyond just focusing on the payoff
amounts.

                                                                                       GPT-4
Output using O
Reasoning:
The player from Team 2 has suggested the decision '§' in the S2 game of the Pie game.  The message in German
translates to "Even though § is the lower payout, it is the only one where both teams get a clear solution, so
they don't have to waver!" This message indicates a few key points:
1.  The player acknowledges the payoff structure, recognizing that '§' is not the highest paying option.
2.  The player emphasizes the clarity and definiteness of the '§' option, suggesting a preference for a
straightforward and unambiguous decision.
3.  The player does not mention any strategic anticipation of the other team's decision-making process but focuses
on the inherent property of the '§' option as being a clear solution for both teams.
Given these points, the player seems to be responding to the label salience of the game, as '§' is the visually
distinctive option in the Pie game.  The player also shows an attraction to the action that provides a clear
solution, which can be interpreted as a preference for simplicity or fairness over maximizing payoffs.  This
suggests that the player is not exhibiting strategic reasoning based on the expected choices of the other team but
is instead focused on an intrinsic property of the '§' option.
The lower bound is set at 0 because the player's message does not clearly exhibit strategic reasoning; they are
focused on the clarity of the decision rather than the strategic implications of the payoffs.  The upper bound is
set at 1 because the player's message could be interpreted as a basic strategic consideration, where they suggest
'§' as a clear solution that both teams might gravitate towards without considering that the other team might also
be strategic.  However, the message does not explicitly state that the other team will think the same way, which
would be necessary for a higher level of reasoning.
Output using O+
Reasoning:
The message suggests that the player is focusing on the fact that the § option is the only one that provides
a clear solution for both teams, despite it being the lower payoff in the S2 variation of the Pie Game.  The
player is emphasizing coordination on a unique solution rather than the amount of the payoff.  This indicates
that the player is not primarily driven by the desire for a higher payoff but rather by the desire for a clear and
unambiguous coordination point.
The message does not explicitly mention a preference for the label of §, but the emphasis on the unique solution
that §provides could be interpreted as a form of label salience, as the player is pointing out a distinctive
feature of the §option.  However, since the message is more about the strategic aspect of coordination rather than
an inherent preference for the label itself, it is more appropriate to classify the label salience as "no."
The player does not exhibit a preference for higher payoffs, as they are advocating for the option with a lower
payoff.  Instead, they are focused on the strategic aspect of ensuring coordination.  Therefore, the payoff
salience is "L" for preferring low payoffs.
In terms of the level of reasoning, the player is considering the strategic aspect of the game, which involves
thinking about the other team's potential decision.  This suggests a level of reasoning beyond level-0, as they
are not choosing based on instinctive reactions or random choice.  The player is reasoning at least at level-1,
as they are best responding to their belief about the other team's decision.  However, since the message does not
indicate a belief about higher-order reasoning (i.e., what the other team thinks about their team's thinking), the
upper bound of reasoning is also level-1.
```

D.4. Additional Analysis

		no-CoT		CoT	
		0-*Shot*	*n-Shot*	0-*Shot*	*n-Shot*
GPT-3.5	O	.499	–	.522	–
	O_+	.625	.622	.652	.655
GPT-4	O	.664	–	.712	–
	O_+	.68	.70	.681	**.76**

Table 49: Lower Bound Accuracy

		no-CoT		CoT	
		0-*Shot*	*n-Shot*	0-*Shot*	*n-Shot*
GPT-3.5	O	.309	–	.402	–
	O_+	.378	.577	.412	.609
GPT-4	O	.597	–	.651	–
	O_+	.616	.66	.641	**.681**

Table 50: Upper Bound Accuracy

		no-CoT		CoT	
		0-*Shot*	*n-Shot*	0-*Shot*	*n-Shot*
GPT-3.5	O	.541	–	.577	–
	O_+	.643	.698	.651	.722
GPT-4	O	.704	–	.75	–
	O_+	.719	.721	.719	**.772**

Table 51: F1

F1 measure presented in Table 51 is with respect to the classification of the level of thinking intervals, which are defined by the classified upper and lower bounds of the level of thinking. It is a multi-label classification, calculated using the equations presented in Godbole and Sarawagi (2004, page 6). It is neither a micro nor a macro averaged measure but calculated as the average over instances (messages).

References

Aiyappa, Rachith, Jisun An, Haewoon Kwak, and Yong-Yeol Ahn, "Can we trust the evaluation on ChatGPT?," *arXiv preprint arXiv:2303.12767*, 2023.

Alonso-Robisco, Andres and José Manuel Carbó, "Analysis of CBDC narrative by central banks using large language models," *Finance Research Letters*, 2023, *58*, 104643.

Amin, Mostafa M, Erik Cambria, and Björn W Schuller, "Will affective computing emerge from foundation models and general artificial intelligence? a first evaluation of chatgpt," *IEEE Intelligent Systems*, 2023, *38* (2), 15–23.

Anthropic, "Mitigating Jailbreaks and Prompt Injections," 2023. Accessed: 2023-04-25.

Anthropic, "Prompt Chaining," `https://docs.anthropic.com/en/docs/chain-prompts` 2024. Accessed: 2024-06-03.

Arad, Ayala, David Hugh-Jones, and Stefan P. Penczynski, "Communication is informative about cooperation," Technical Report, School of Economics, University of East Anglia, Norwich, UK. 2024.

Bach, Stephen H, Victor Sanh, Zheng-Xin Yong, Albert Webson, Colin Raffel, Nihal V Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Fevry et al., "Promptsource: An integrated development environment and repository for natural language prompts," *arXiv preprint arXiv:2202.01279*, 2022.

Baktash, Jawid Ahmad and Mursal Dawodi, "Gpt-4: A review on advancements and opportunities in natural language processing," *arXiv preprint arXiv:2305.03195*, 2023.

Bartolomeo, Giovanni Di, Martin Dufwenberg, and Stefano Papa, "Promises and partner-switch," *Journal of the Economic Science Association*, 2023, *9* (1), 77–89.

__ , __ , __ , **and Francesco Passarelli**, "Promises, expectations & causation," *Games and Economic Behavior*, 2019, *113*, 137–146.

Beltagy, Iz, Matthew E Peters, and Arman Cohan, "Longformer: The long-document transformer," *arXiv preprint arXiv:2004.05150*, 2020.

Bender, Emily M, "On achieving and evaluating language-independence in NLP," *Linguistic Issues in Language Technology*, 2011, *6*.

__ , **Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell**, "On the dangers of stochastic parrots: Can language models be too big?ï¿½ï¿½," in

"Proceedings of the 2021 ACM conference on fairness, accountability, and transparency" 2021, pp. 610–623.

Bhat, Savita and Vasudeva Varma, "Large Language Models As Annotators: A Preliminary Evaluation For Annotating Low-Resource Language Content," in "Proceedings of the 4th Workshop on Evaluation and Comparison of NLP Systems" 2023, pp. 100–107.

Bicchieri, Cristina and Alex Chavez, "Behaving as expected: Public information and fairness norms," *Journal of Behavioral Decision Making*, 2010, *23* (2), 161–178.

Bjerva, Johannes, Robert Östling, Maria Han Veiga, Jörg Tiedemann, and Isabelle Augenstein, "What do language representations really represent?," *Computational Linguistics*, 2019, *45* (2), 381–389.

Black, Michael J., "I asked #Galactica about some things I know about and I'm troubled. In all cases, it was wrong or biased but sounded right and authoritative.," Twitter Nov 2022. Tweet.

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

Burchardi, Konrad B. and Stefan P. Penczynski, "Out of your mind: Eliciting individual reasoning in one shot games," *Games and Economic Behavior*, 2014, *84* (1), 39 – 57.

Camerer, Colin F., Teck-Hua Ho, and Juin-Kuan Chong, "A Cognitive Hierarchy Model of Games," *The Quarterly Journal of Economics*, August 2004, *119* (3), 861–898.

Capra, C Mónica, "Understanding decision processes in guessing games: a protocol analysis approach," *Journal of the Economic Science Association*, 2019, *5* (1), 123–135.

Castelli, Ilaria, Davide Massaro, Cristina Bicchieri, Alex Chavez, and Antonella Marchetti, "Fairness norms and theory of mind in an ultimatum game: judgments, offers, and decisions in school-aged children," *PloS one*, 2014, *9* (8), e105024.

Çelebi, Can and Stefan P. Penczynski, "Strategic Thinking in Jury Decisions: An Experimental Study," Technical Report, School of Economics, University of East Anglia, Norwich, UK. 2023.

Chae, Youngjin and Thomas Davidson, "Large language models for text

classification: From zero-shot learning to fine-tuning," *Open Science Foundation*, 2023.

Chang, Ting-Yun and Robin Jia, "Data curation alone can stabilize in-context learning," *arXiv preprint arXiv:2212.10378*, 2022.

Charness, Gary and Martin Dufwenberg, "Promises and partnership," *Econometrica*, 2006, *74* (6), 1579–1601.

Chowdhery, Aakanksha, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann et al., "Palm: Scaling language modeling with pathways," *Journal of Machine Learning Research*, 2023, *24* (240), 1–113.

Chung, Hyung Won, 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," *arXiv preprint arXiv:2210.11416*, 2022.

Clavié, Benjamin, Alexandru Ciceu, Frederick Naylor, Guillaume Soulié, and Thomas Brightwell, "Large language models in the workplace: A case study on prompt engineering for job type classification," in "International Conference on Applications of Natural Language to Information Systems" Springer 2023, pp. 3–17.

Conneau, Alexis and Guillaume Lample, "Cross-lingual language model pretraining," *Advances in neural information processing systems*, 2019, *32*.

Cooper, David J and John H Kagel, "Are two heads better than one? Team versus individual play in signaling games," *American Economic Review*, 2005, *95* (3), 477–509.

__ , Ian Krajbich, and Charles N Noussair, "Choice-process data in experimental economics," 2019.

Costa-Gomes, Miguel A. and Vincent P. Crawford, "Cognition and Behavior in Two-Person Guessing Games: An Experimental Study," *American Economic Review*, December 2006, *96* (5), 1737–1768.

Crawford, Vincent P., Miguel A. Costa-Gomes, and Nagore Iriberri, "Structural Models of Nonequilibrium Strategic Thinking: Theory, Evidence, and Applications," *Journal of Economic Literature*, September 2013, *51* (1), 5–62.

__ , Uri Gneezy, and Yuval Rottenstreich, "The Power of Focal Points Is Limited: Even Minute Payoff Asymmetry May Yields Large Coordination Failures," *American Economic Review*, July 2008, *98* (4), 1443–1458.

den Assem, Martijn J Van, Dennie Van Dolder, and Richard H Thaler, "Split

or steal? Cooperative behavior when the stakes are large," *Management Science*, 2012, *58* (1), 2–20.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

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

Ederer, Florian and Alexander Stremitzer, "Promises and expectations," *Games and Economic Behavior*, 2017, *106*, 161–178.

Efrat, Avia and Omer Levy, "The turking test: Can language models understand instructions?," *arXiv preprint arXiv:2010.11982*, 2020.

Fanta, Nicolas and Roman Horvath, "Artificial intelligence and central bank communication: the case of the ECB," *Applied Economics Letters*, 2024, pp. 1–8.

Feddersen, Timothy and Wolfgang Pesendorfer, "Convicting the innocent: The inferiority of unanimous jury verdicts under strategic voting," *American Political Science Review*, 1998, pp. 23–35.

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

Gekhman, Zorik, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig, "Does Fine-Tuning LLMs on New Knowledge Encourage Hallucinations?," *arXiv preprint arXiv:2405.05904*, 2024.

Gentzkow, Matthew, Bryan Kelly, and Matt Taddy, "Text as data," *Journal of Economic Literature*, 2019, *57* (3), 535–574.

Gilardi, Fabrizio, Meysam Alizadeh, and Maël Kubli, "ChatGPT outperforms crowd workers for text-annotation tasks," *Proceedings of the National Academy of Sciences*, 2023, *120* (30), e2305016120.

Glasserman, Paul and Caden Lin, "Assessing Look-Ahead Bias in Stock Return Predictions Generated By GPT Sentiment Analysis," *arXiv preprint arXiv:2309.17322*, 2023.

Godbole, Shantanu and Sunita Sarawagi, "Discriminative methods for multi-labeled classification," in "Pacific-Asia conference on knowledge discovery and data mining" Springer 2004, pp. 22–30.

Gu, Jiasheng, Hongyu Zhao, Hanzi Xu, Liangyu Nie, Hongyuan Mei, and Wenpeng Yin, "Robustness of learning from task instructions," *arXiv preprint*

arXiv:2212.03813, 2022.

Guarnaschelli, Serena, Richard D McKelvey, and Thomas R Palfrey, "An experimental study of jury decision rules," *American Political Science Review*, 2000, pp. 407–423.

Hansen, Anne Lundgaard and Sophia Kazinnik, "Can chatgpt decipher fedspeak," *Available at SSRN*, 2023.

Haviv, Adi, Jonathan Berant, and Amir Globerson, "BERTese: Learning to speak to BERT," *arXiv preprint arXiv:2103.05327*, 2021.

He, Zeyu, Chieh-Yang Huang, Chien-Kuang Cornelia Ding, Shaurya Rohatgi, and Ting-Hao'Kenneth' Huang, "If in a Crowdsourced Data Annotation Pipeline, a GPT-4," *arXiv preprint arXiv:2402.16795*, 2024.

Heseltine, Michael and Bernhard Clemm von Hohenberg, "Large language models as a substitute for human experts in annotating political text," *Research & Politics*, 2024, *11* (1), 20531680241236239.

Houser, Daniel and Erte Xiao, "Classification of natural language messages using a coordination game," *Experimental Economics*, 2011, *14*, 1–14.

Huang, Allen H, Hui Wang, and Yi Yang, "FinBERT: A large language model for extracting information from financial text," *Contemporary Accounting Research*, 2023, *40* (2), 806–841.

Huang, Jie and Kevin Chen-Chuan Chang, "Towards reasoning in large language models: A survey," *arXiv preprint arXiv:2212.10403*, 2022.

Hüning, Hendrik, Lydia Mechtenberg, and Stephanie Wang, "Detecting arguments and their positions in experimental communication data," *Available at SSRN 4052402*, 2022.

__ , __ , **and** __ , "Using Arguments to Persuade: Experimental Evidence," *Available at SSRN 4244989*, 2022.

Ismayilov, Huseyn and Jan Potters, "Why do promises affect trustworthiness, or do they?," *Experimental Economics*, 2016, *19*, 382–393.

IV, Robert L Logan, Ivana Balažević, Eric Wallace, Fabio Petroni, Sameer Singh, and Sebastian Riedel, "Cutting down on prompts and parameters: Simple few-shot learning with language models," *arXiv preprint arXiv:2106.13353*, 2021.

Jha, Manish, Jialin Qian, Michael Weber, and Baozhong Yang, "ChatGPT and corporate policies," Technical Report, National Bureau of Economic Research 2024.

Jiang, Zhengbao, Frank F Xu, Jun Araki, and Graham Neubig, "How can we know what language models know?," *Transactions of the Association for*

Computational Linguistics, 2020, *8*, 423–438.

Kaddour, Jean, Joshua Harris, Maximilian Mozes, Herbie Bradley, Roberta Raileanu, and Robert McHardy, "Challenges and applications of large language models," *arXiv preprint arXiv:2307.10169*, 2023.

Kaplan, Jared, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei, "Scaling laws for neural language models," *arXiv preprint arXiv:2001.08361*, 2020.

Kim, Alex, Maximilian Muhn, and Valeri Nikolaev, "From Transcripts to Insights: Uncovering Corporate Risks Using Generative AI," *arXiv preprint arXiv:2310.17721*, 2023.

Kojima, Takeshi, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa, "Large language models are zero-shot reasoners," *Advances in neural information processing systems*, 2022, *35*, 22199–22213.

Kong, Aobo, Shiwan Zhao, Hao Chen, Qicheng Li, Yong Qin, Ruiqi Sun, and Xin Zhou, "Better zero-shot reasoning with role-play prompting," *arXiv preprint arXiv:2308.07702*, 2023.

Kumar, Sawan and Partha Talukdar, "Reordering examples helps during priming-based few-shot learning," *arXiv preprint arXiv:2106.01751*, 2021.

__ **and** __ , "Reordering examples helps during priming-based few-shot learning," *arXiv preprint arXiv:2106.01751*, 2021.

Kuzman, Taja, Igor Mozetic, and Nikola Ljubešic, "ChatGPT: beginning of an end of manual linguistic data annotation," *Use case of automatic genre identification*, 2023.

Kuznia, Kirby, Swaroop Mishra, Mihir Parmar, and Chitta Baral, "Less is more: Summary of long instructions is better for program synthesis," *arXiv preprint arXiv:2203.08597*, 2022.

Lai, Viet Dac, Nghia Trung Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Huu Nguyen, "Chatgpt beyond english: Towards a comprehensive evaluation of large language models in multilingual learning," *arXiv preprint arXiv:2304.05613*, 2023.

Lester, Brian, Rami Al-Rfou, and Noah Constant, "The power of scale for parameter-efficient prompt tuning," *arXiv preprint arXiv:2104.08691*, 2021.

Li, Lingyao, Lizhou Fan, Shubham Atreja, and Libby Hemphill, "âHOTâ ChatGPT: The promise of ChatGPT in detecting and discriminating hateful, offensive, and toxic comments on social media," *ACM Transactions on the Web*,

2024, *18* (2), 1–36.

Li, Xiang Lisa and Percy Liang, "Prefix-tuning: Optimizing continuous prompts for generation," *arXiv preprint arXiv:2101.00190*, 2021.

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

__ **and** __ , "Finding support examples for in-context learning," *arXiv preprint arXiv:2302.13539*, 2023.

Liu, Jiachang, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen, "What Makes Good In-Context Examples for GPT-3?," *arXiv preprint arXiv:2101.06804*, 2021.

Liu, Peng and Zhizhong Li, "Task complexity: A review and conceptualization framework," *International Journal of Industrial Ergonomics*, 2012, *42* (6), 553–568.

Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

Lopez-Lira, Alejandro and Yuehua Tang, "Can chatgpt forecast stock price movements? return predictability and large language models," *arXiv preprint arXiv:2304.07619*, 2023.

Lou, Renze and Wenpeng Yin, "Toward zero-shot instruction following," in "Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop" 2024, pp. 50–60.

__ , **Kai Zhang, and Wenpeng Yin**, "Is prompt all you need? no. a comprehensive and broader view of instruction learning," *arXiv preprint arXiv:2303.10475*, 2023.

Lu, Yao, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp, "Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity," *arXiv preprint arXiv:2104.08786*, 2021.

Luo, Man, Xin Xu, Zhuyun Dai, Panupong Pasupat, Mehran Kazemi, Chitta Baral, Vaiva Imbrasaite, and Vincent Y Zhao, "Dr. icl: Demonstration-retrieved in-context learning," *arXiv preprint arXiv:2305.14128*, 2023.

Madaan, Aman and Amir Yazdanbakhsh, "Text and patterns: For effective chain of thought, it takes two to tango," *arXiv preprint arXiv:2209.07686*, 2022.

Matter, Daniel, Miriam Schirmer, Nir Grinberg, and Jürgen Pfeffer, "Close to Human-Level Agreement: Tracing Journeys of Violent Speech in Incel Posts with GPT-4-Enhanced Annotations," *arXiv preprint arXiv:2401.02001*, 2024.

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

Mishra, Swaroop, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi, "Cross-task generalization via natural language crowdsourcing instructions," *arXiv preprint arXiv:2104.08773*, 2021.

__ , __ , __ , **Yejin Choi, and Hannaneh Hajishirzi**, "Reframing Instructional Prompts to GPTk's Language," *arXiv preprint arXiv:2109.07830*, 2021.

Moghaddam, Shima Rahimi and Christopher J Honey, "Boosting theory-of-mind performance in large language models via prompting," *arXiv preprint arXiv:2304.11490*, 2023.

Müller, Dominik, Iñaki Soto-Rey, and Frank Kramer, "Towards a guideline for evaluation metrics in medical image segmentation," *BMC Research Notes*, 2022, *15* (1), 210.

Nivre, Joakim, Marie-Catherine De Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira et al., "Universal dependencies v1: A multilingual treebank collection," in "Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)" 2016, pp. 1659–1666.

Obaid, Khaled and Kuntara Pukthuanthong, "Distortions in Financial Narratives: A ChatGPT Approach," 2024.

OpenAI, "Fine-tuned Classification," `https://github.com/openai/openai-cookbook/blob/main/examples/Fine-tuned_classification.ipynb` 2023. Accessed: 2023-04-20.

__ , "Prompt design," `https://platform.openai.com/docs/guides/completion/prompt-design` 2023.

__ , "Using Logprobs," `https://cookbook.openai.com/examples/using_logprobs` 2023. Accessed: 2023-04-21.

Ouyang, Long, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray et al., "Training language models to follow instructions with human feedback," *Advances in neural information processing systems*, 2022, *35*, 27730–27744.

Pan, Jane, Tianyu Gao, Howard Chen, and Danqi Chen, "What In-Context Learning" Learns" In-Context: Disentangling Task Recognition and Task

Learning," *arXiv preprint arXiv:2305.09731*, 2023.

Pangakis, Nicholas, Samuel Wolken, and Neil Fasching, "Automated annotation with generative ai requires validation," *arXiv preprint arXiv:2306.00176*, 2023.

Penczynski, Stefan P, "Using machine learning for communication classification," *Experimental Economics*, 2019, *22* (4), 1002–1029.

Peskine, Youri, Damir Korenčić, Ivan Grubisic, Paolo Papotti, Raphael Troncy, and Paolo Rosso, "Definitions Matter: Guiding GPT for Multi-label Classification," in "Findings of the Association for Computational Linguistics: EMNLP 2023" 2023, pp. 4054–4063.

Peskoff, Denis, Adam Visokay, Sander Schulhoff, Benjamin Wachspress, Alan Blinder, and Brandon M Stewart, "GPT Deciphering Fedspeak: Quantifying Dissent Among Hawks and Doves," in "Findings of the Association for Computational Linguistics: EMNLP 2023" 2023, pp. 6529–6539.

Puri, Ravsehaj Singh, Swaroop Mishra, Mihir Parmar, and Chitta Baral, "How many data samples is an additional instruction worth?," *arXiv preprint arXiv:2203.09161*, 2022.

Radford, Alec, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever, "Language Models are Unsupervised Multitask Learners," *OpenAI Blog*, 2019.

Rathje, Steve, Dan-Mircea Mirea, Ilia Sucholutsky, Raja Marjieh, Claire Robertson, and Jay J Van Bavel, "GPT is an effective tool for multilingual psychological text analysis," 2023.

Razeghi, Yasaman, Robert L Logan IV, Matt Gardner, and Sameer Singh, "Impact of pretraining term frequencies on few-shot reasoning," *arXiv preprint arXiv:2202.07206*, 2022.

Reiss, Michael V, "Testing the reliability of chatgpt for text annotation and classification: A cautionary remark," *arXiv preprint arXiv:2304.11085*, 2023.

Reynolds, Laria and Kyle McDonell, "Prompt programming for large language models: Beyond the few-shot paradigm," in "Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems" 2021, pp. 1–7.

Rytting, Christopher Michael, Taylor Sorensen, Lisa Argyle, Ethan Busby, Nancy Fulda, Joshua Gubler, and David Wingate, "Towards coding social science datasets with language models," *arXiv preprint arXiv:2306.02177*, 2023.

Salewski, Leonard, Stephan Alaniz, Isabel Rio-Torto, Eric Schulz, and Zeynep Akata, "In-Context Impersonation Reveals Large Language Models' Strengths and Biases," *Advances in Neural Information Processing Systems*, 2024, *36.*

Savelka, Jaromir, Kevin D Ashley, Morgan A Gray, Hannes Westermann, and Huihui Xu, "Can gpt-4 support analysis of textual data in tasks requiring highly specialized domain expertise?," *arXiv preprint arXiv:2306.13906*, 2023.

Scao, Teven Le and Alexander M Rush, "How many data points is a prompt worth?," *arXiv preprint arXiv:2103.08493*, 2021.

Schick, Timo and Hinrich Schütze, "It's not just size that matters: Small language models are also few-shot learners," *arXiv preprint arXiv:2009.07118*, 2020.

Schotter, Andrew, "Decision making with naive advice," *American Economic Review*, 2003, *93* (2), 196–201.

Sclar, Melanie, Yejin Choi, Yulia Tsvetkov, and Alane Suhr, "Quantifying Language Models' Sensitivity to Spurious Features in Prompt Design or: How I learned to start worrying about prompt formatting," *arXiv preprint arXiv:2310.11324*, 2023.

Shen, Xinyue, Zeyuan Chen, Michael Backes, Yun Shen, and Yang Zhang, ""do anything now": Characterizing and evaluating in-the-wild jailbreak prompts on large language models," *arXiv preprint arXiv:2308.03825*, 2023.

Smales, Lee A, "Classification of RBA monetary policy announcements using ChatGPT," *Finance Research Letters*, 2023, *58*, 104514.

Sorower, Mohammad S, "A literature survey on algorithms for multi-label learning," *Oregon State University, Corvallis*, 2010, *18* (1), 25.

Su, Hongjin, Jungo Kasai, Chen Henry Wu, Weijia Shi, Tianlu Wang, Jiayi Xin, Rui Zhang, Mari Ostendorf, Luke Zettlemoyer, Noah A Smith et al., "Selective annotation makes language models better few-shot learners," *arXiv preprint arXiv:2209.01975*, 2022.

Takagishi, Haruto, Shinya Kameshima, Joanna Schug, Michiko Koizumi, and Toshio Yamagishi, "Theory of mind enhances preference for fairness," *Journal of experimental child psychology*, 2010, *105* (1-2), 130–137.

Talamadupula, Kartik, "A Guide to LLM Inference Performance Monitoring," `https://symbl.ai/developers/blog/a-guide-to-llm-inference-performance-monitoring/` 2024. Accessed: 2024-05-29.

Taylor, Ross, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic, "Galactica: A large language model for science," *arXiv preprint arXiv:2211.09085*, 2022.

Törnberg, Petter, "Chatgpt-4 outperforms experts and crowd workers in

annotating political twitter messages with zero-shot learning," *arXiv preprint arXiv:2304.06588*, 2023.

van Elten, Jonas and Stefan P Penczynski, "Coordination games with asymmetric payoffs: An experimental study with intra-group communication," *Journal of Economic Behavior & Organization*, 2020, *169*, 158–188.

Vanberg, Christoph, "Why do people keep their promises? An experimental test of two explanations 1," *Econometrica*, 2008, *76* (6), 1467–1480.

Vossen, Rainer and Gerrit J Dimmendaal, *The Oxford handbook of African languages*, Oxford University Press, USA, 2020.

Wang, Lei, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim, "Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models," *arXiv preprint arXiv:2305.04091*, 2023.

Wang, Xuezhi, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou, "Self-consistency improves chain of thought reasoning in language models," *arXiv preprint arXiv:2203.11171*, 2022.

Webson, Albert and Ellie Pavlick, "Do prompt-based models really understand the meaning of their prompts?," *arXiv preprint arXiv:2109.01247*, 2021.

Wei, Jason, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le, "Finetuned language models are zero-shot learners," *arXiv preprint arXiv:2109.01652*, 2021.

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

__ , **Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler et al.**, "Emergent abilities of large language models," *arXiv preprint arXiv:2206.07682*, 2022.

White, Jules, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C Schmidt, "A prompt pattern catalog to enhance prompt engineering with chatgpt," *arXiv preprint arXiv:2302.11382*, 2023.

Wu, Tongshuang, Michael Terry, and Carrie Jun Cai, "Ai chains: Transparent and controllable human-ai interaction by chaining large language model prompts,"

in "Proceedings of the 2022 CHI conference on human factors in computing systems" 2022, pp. 1–22.

Wu, Zhaofeng, Linlu Qiu, Alexis Ross, Ekin Akyürek, Boyuan Chen, Bailin Wang, Najoung Kim, Jacob Andreas, and Yoon Kim, "Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks," *arXiv preprint arXiv:2307.02477*, 2023.

Yang, Jie, Judith Redi, Gianluca Demartini, and Alessandro Bozzon, "Modeling task complexity in crowdsourcing," in "Proceedings of the AAAI Conference on Human Computation and Crowdsourcing," Vol. 4 2016, pp. 249–258.

Ye, J, X Chen, N Xu, C Zu, Z Shao, S Liu, Y Cui, Z Zhou, C Gong, Y Shen et al., "A comprehensive capability analysis of gpt-3 and gpt-3.5 series models. arXiv 2023," *arXiv preprint arXiv:2303.10420.*

Yoo, Kang Min, Junyeob Kim, Hyuhng Joon Kim, Hyunsoo Cho, Hwiyeol Jo, Sang-Woo Lee, Sang goo Lee, and Taeuk Kim, "Ground-truth labels matter: A deeper look into input-label demonstrations," *arXiv preprint arXiv:2205.12685*, 2022.

Yuan, J, P Bao, Z Chen, M Yuan, J Zhao, J Pan, Y Xie, Y Cao, Y Wang, Z Wang et al., "Advanced prompting as a catalyst: Empowering large language models in the management of gastrointestinal cancers," *The Innovation*, 2023.

Yuan, Weizhe, Graham Neubig, and Pengfei Liu, "Bartscore: Evaluating generated text as text generation," *Advances in Neural Information Processing Systems*, 2021, *34*, 27263–27277.

Zhang, Bowen, Daijun Ding, and Liwen Jing, "How would stance detection techniques evolve after the launch of chatgpt?," *arXiv preprint arXiv:2212.14548*, 2022.

Zhang, Susan, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin et al., "Opt: Open pre-trained transformer language models," *arXiv preprint arXiv:2205.01068*, 2022.

Zhao, Wayne Xin, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong et al., "A survey of large language models," *arXiv preprint arXiv:2303.18223*, 2023.

Zheng, Huaixiu Steven, Swaroop Mishra, Xinyun Chen, Heng-Tze Cheng, Ed H Chi, Quoc V Le, and Denny Zhou, "Take a step back: Evoking reasoning via abstraction in large language models," *arXiv preprint arXiv:2310.06117*, 2023.

Zhong, Qihuang, Liang Ding, Juhua Liu, Bo Du, and Dacheng Tao, "Can chatgpt understand too? a comparative study on chatgpt and fine-tuned bert," *arXiv preprint arXiv:2302.10198*, 2023.

Zhu, Yiming, Peixian Zhang, Ehsan-Ul Haq, Pan Hui, and Gareth Tyson, "Can chatgpt reproduce human-generated labels? a study of social computing tasks," *arXiv preprint arXiv:2304.10145*, 2023.

Ziems, Caleb, William Held, Omar Shaikh, Jiaao Chen, Zhehao Zhang, and Diyi Yang, "Can large language models transform computational social science?," *Computational Linguistics*, 2024, pp. 1–55.