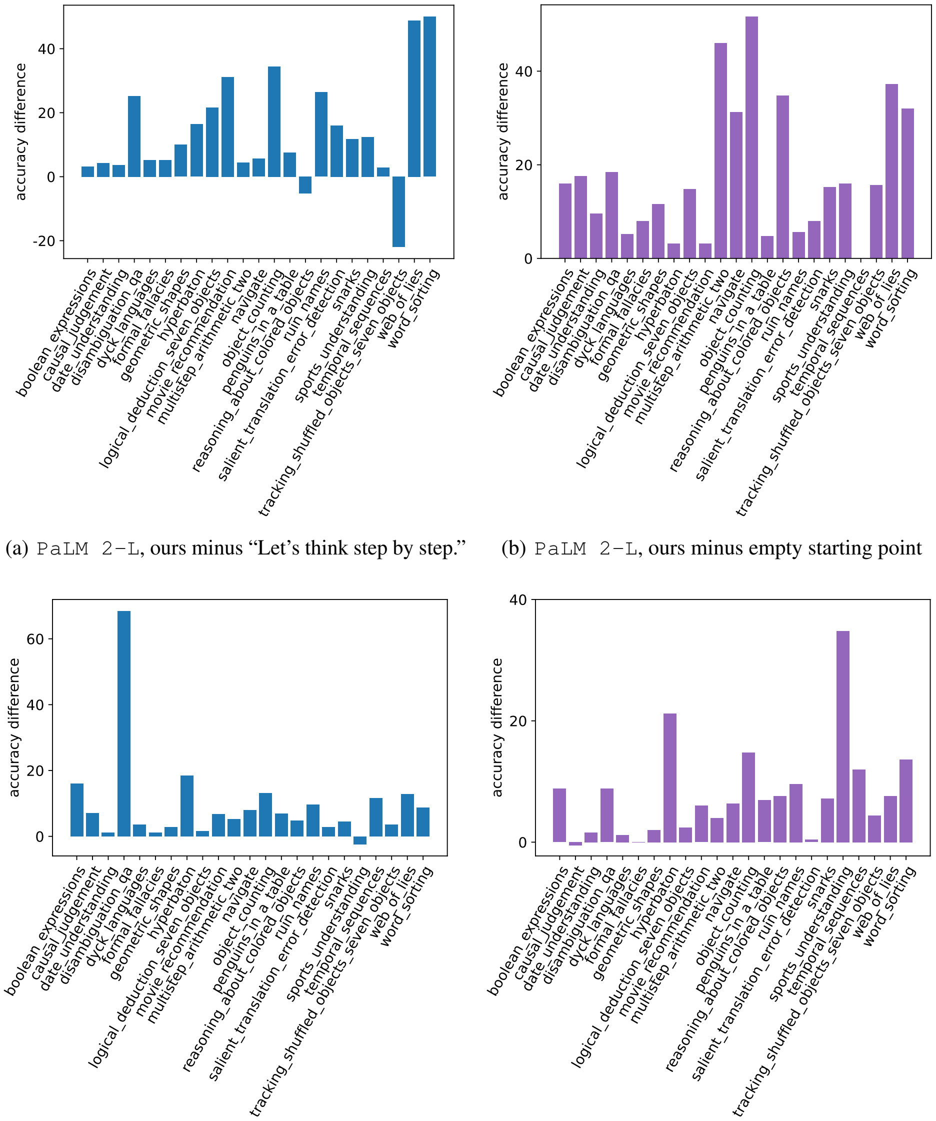
Table 9: BBH task-wise instructions found by prompt optimization with the `text-bison` scorer and the `PaLM 2-L-IT` optimizer. The optimization starts from the empty string.

Task	Our Instruction
boolean_expressions	Not (not False) and not not False is False
causal_judgement	A typical person would likely answer the questions about causation as follows:
date_understanding	Today is February 28, 2023. It is a Tuesday. Yesterday was Monday, February 27, 2023. Tomorrow will be Wednesday, March 1, 2023. A week ago, it was February 21, 2023, and a month ago, it was January 28, 2023. A year from now, it will be February 28, 2024. The day of the week is important to note because it will help us to correctly answer the questions below. Not all years are leap years that contain February 29.
disambiguation_qa	A pronoun is a word that stands in for a noun. The noun that a pronoun refers to is called its antecedent. To identify the antecedent of a pronoun, look for the noun that the pronoun could be referring to. If there is only one possible noun, then that is the antecedent. If there are two or more possible nouns, then the antecedent is ambiguous. Use the context of the sentence to help you determine the correct antecedent.
dyck_languages	{ }
formal_fallacies	How to Evaluate Deductive Validity of an Argument
geometric_shapes	What shape is this SVG code drawing, and how many sides does it have?
hyperbaton	In English, adjectives are typically placed before nouns in a specific order. The order is: opinion, size, shape, age, color, origin, material, purpose, noun. For example, the sentence "the big, old, red barn" would be considered grammatically correct, while the sentence "the old, big, red barn" would not. Adjectives that come before nouns are called attributive adjectives, while adjectives that come after nouns are called predicative adjectives.
logical_deduction _seven_objects	In this logical reasoning task, you will be given a series of paragraphs, each of which describes a set of objects arranged in a fixed order. The statements in each paragraph are logically consistent. You must read each paragraph carefully and use the information given to determine the logical relationships between the objects. You will then be asked a question about the order of the objects. Read each question carefully and choose the option that answers the question correctly.
movie_recommendation	What is the highest-rated movie similar to the given movies, with a similar IMDb rating and released in the same year?
multistep_arithmetic_two	Let's solve these equations using PEMDAS order of operations. Remember that PEMDAS stands for parentheses, exponents, multiplication and division, and addition and subtraction.
navigate	Starting at the origin, facing north, follow the instructions. If your displacement from the origin is zero and your direction is unchanged, then your answer is Yes. Otherwise, your answer is No.
object_counting	Let me help you count the items you have. Just list them one by one, separated by commas. I will then count each item and tell you how many items there are in total.
penguins_in_a_table	This table shows information about penguins. The columns show the penguin's name, age, height (in cm), and weight (in kg). The penguins are listed in order of their age, from youngest to oldest.
reasoning_about _colored_objects	First, read the input carefully. Then, identify all the objects mentioned, their colors, and their positions. Next, visualize the objects and their positions in your mind. Finally, answer the questions accurately based on the information given. Make sure to pay attention to the order of the objects.
ruin_names	A humorous edit of an artist or movie name can be created by replacing one or more letters to form a new word or phrase that sounds similar but has a different meaning. The new word or phrase should be relevant to the original word, but it should also be a surprise, which makes the edit funny. For example, the artist or movie name "Rocky" can be changed to "Ricky," and "Schindler's List" can be changed to "Schindler's Lift." Be creative and have fun!
salient_translation _error_detection	The following translations from German to English contain a particular error. The error may be one of the following types: Named Entities, Numerical Values, Modifiers or Adjectives, Negation or Antonyms, Facts, or Dropped Content. Please identify the error.
snarks	The statement
sports_understanding	To determine the plausibility of a sports sentence, I will first identify the sport, athletes, teams, and events mentioned in the sentence. Then, I will use my knowledge of the rules of the sport, the context of the sentence, common sense, and my knowledge of the world to determine whether the sentence is plausible. I will also consider the time period and location, as well as any other relevant information. Finally, I will return a score of 1 for plausible sentences and 0 for implausible ones.
temporal_sequences	To determine the time period when a person went to a place, first identify all the time periods when the person's whereabouts are unknown. Then, rule out any time periods during which the person was seen doing something else or the place was closed. The remaining time periods are the possible times when the person could have gone to the place.
tracking_shuffled_objects _seven_objects	At the start of the game, Claire has a blue ball. Throughout the game, pairs of people swap balls. Claire ends up with the yellow ball.
web_of_lies	People in a group either tell the truth or lie. The truthfulness of a person's statement is determined by the statement of the previous person. If the previous person told the truth, then the current person who says the opposite is lying. If the previous person lied, then the current person who says the opposite is telling the truth. This rule applies to all subsequent statements.
word_sorting	Sort the following words alphabetically, ignoring case and punctuation. Print the sorted list.

E.2 GPT-3.5-TURBO AS OPTIMIZER, OPTIMIZATION STARTING FROM THE EMPTY STRING

Table 11, 12 and 13 show the instructions found by prompt optimization. Their accuracies are listed in Table 10. Figure 25 visualizes the difference between their accuracies and those of the baselines "Let's think step by step." and the empty string. The optimizations find instructions better than the empty starting point, and most of the found instructions are better than "Let's think step by step".

One caveat in the A_begin instructions (Table 11) is that a lot of the found instructions are imperative or interrogative sentences that are more suitable to be put into "Q:" rather than "A:", like "Solve the sequence by properly closing the parentheses." for dyck_languages and "Which movie option from the given choices ...?" for movie_recommendation. Such styles appear more often here than the PaLM 2-L-IT optimizer results (Table 8), showing PaLM 2-L-IT understands the needed style better. In Section E.3, we show the A_begin optimization results with the non-empty starting point "Let's solve the problem.". Most results there are declarative sentences – more suitable for A_begin.



(a) PaLM 2-L, ours minus "Let's think step by step."

(b) PaLM 2-L, ours minus empty starting point

(c) text-bison, ours minus "Let's think step by step."

(d) text-bison, ours minus empty starting point

Figure 25: On 23 BBH tasks, the accuracy differences among instructions found by prompt optimization (with the gpt-3.5-turbo optimizer), "Let's think step by step.", and the empty string (optimization starting point).

Table 10: Accuracies on BBH tasks with the `gpt-3.5-turbo` optimizer that starts from the empty string. The `PaLM 2-L` scores are from A_begin (left) instructions; the `text-bison` scores include Q_begin (left) and Q_end (right) instructions.

Task	Scorer	Our Acc (`begin`)	Our Acc (`end`)
		training / test / overall	training / test / overall
boolean_expressions	`PaLM 2-L`	92.0 / 86.5 / 87.6	N/A
causal_judgement	`PaLM 2-L`	81.1 / 58.7 / 63.1	N/A
date_understanding	`PaLM 2-L`	86.0 / 82.0 / 82.8	N/A
disambiguation_qa	`PaLM 2-L`	80.0 / 74.0 / 75.2	N/A
dyck_languages	`PaLM 2-L`	100.0 / 100.0 / 100.0	N/A
formal_fallacies	`PaLM 2-L`	88.0 / 63.5 / 68.4	N/A
geometric_shapes	`PaLM 2-L`	60.0 / 41.0 / 44.8	N/A
hyperbaton	`PaLM 2-L`	88.0 / 93.0 / 92.0	N/A
logical_deduction_seven_objects	`PaLM 2-L`	76.0 / 56.5 / 60.4	N/A
movie_recommendation	`PaLM 2-L`	84.0 / 86.0 / 85.6	N/A
multistep_arithmetic_two	`PaLM 2-L`	52.0 / 49.0 / 49.6	N/A
navigate	`PaLM 2-L`	76.0 / 67.0 / 68.8	N/A
object_counting	`PaLM 2-L`	78.0 / 79.0 / 78.8	N/A
penguins_in_a_table	`PaLM 2-L`	82.8 / 72.6 / 74.7	N/A
reasoning_about _colored_objects	`PaLM 2-L`	86.0 / 67.5 / 71.2	N/A
ruin_names	`PaLM 2-L`	90.0 / 83.0 / 84.4	N/A
salient_translation_error_detection	`PaLM 2-L`	62.0 / 65.0 / 64.4	N/A
snarks	`PaLM 2-L`	85.7 / 70.6 / 73.6	N/A
sports_understanding	`PaLM 2-L`	68.0 / 57.5 / 59.6	N/A
temporal_sequences	`PaLM 2-L`	100.0 / 99.5 / 99.6	N/A
tracking_shuffled_objects_seven_objects	`PaLM 2-L`	44.0 / 34.5 / 36.4	N/A
web_of_lies	`PaLM 2-L`	92.0 / 91.0 / 91.2	N/A
word_sorting	`PaLM 2-L`	62.0 / 52.0 / 54.0	N/A
boolean_expressions	`text-bison`	84.0 / 78.5 / 79.6	80.0 / 78.0 / 78.4
causal_judgement	`text-bison`	78.4 / 57.3 / 61.5	83.8 / 53.3 / 59.4
date_understanding	`text-bison`	52.0 / 45.0 / 46.4	64.0 / 52.4 / 54.8
disambiguation_qa	`text-bison`	68.0 / 75.5 / 74.0	64.0 / 71.5 / 70.0
dyck_languages	`text-bison`	100.0 / 99.5 / 99.6	100.0 / 100.0 / 100.0
formal_fallacies	`text-bison`	70.0 / 54.5 / 57.6	74.0 / 53.5 / 57.6
geometric_shapes	`text-bison`	28.0 / 15.0 / 17.6	48.0 / 28.0 / 32.0
hyperbaton	`text-bison`	86.0 / 85.0 / 85.2	80.0 / 76.5 / 77.2
logical_deduction_seven_objects	`text-bison`	66.0 / 57.5 / 59.2	62.0 / 55.0 / 56.4
movie_recommendation	`text-bison`	76.0 / 69.5 / 70.8	82.0 / 70.5 / 72.8
multistep_arithmetic_two	`text-bison`	28.0 / 20.5 / 22.0	28.0 / 22.5 / 23.6
navigate	`text-bison`	72.0 / 61.0 / 63.2	68.0 / 59.5 / 61.2
object_counting	`text-bison`	68.0 / 71.0 / 70.4	72.0 / 69.0 / 69.6
penguins_in_a_table	`text-bison`	65.5 / 59.8 / 61.0	79.3 / 53.0 / 58.2
reasoning_about_colored_objects	`text-bison`	84.0 / 76.5 / 78.0	86.0 / 74.0 / 76.4
ruin_names	`text-bison`	80.0 / 74.0 / 75.2	74.0 / 75.0 / 74.8
salient_translation_error_detection	`text-bison`	44.0 / 50.5 / 49.2	48.0 / 51.0 / 50.4
snarks	`text-bison`	82.9 / 79.7 / 80.3	88.6 / 84.6 / 85.4
sports_understanding	`text-bison`	84.0 / 76.5 / 78.0	90.0 / 80.0 / 82.0
temporal_sequences	`text-bison`	50.0 / 54.5 / 53.6	64.0 / 61.5 / 62.0
tracking_shuffled_objects_seven_objects	`text-bison`	22.0 / 18.5 / 19.2	30.0 / 21.5 / 23.2
web_of_lies	`text-bison`	64.0 / 57.5 / 58.8	68.0 / 55.0 / 57.6
word_sorting	`text-bison`	26.0 / 19.0 / 20.4	32.0 / 25.5 / 26.8