Final Report
Project Module: Evaluating Chat-Optimized Language Models
Uday Bhaskar Kale (828247)

Multilingual Assessment of LLMs in Dialogue based game environment.

LLMs are evolving and have started to solve more complex problems. To evaluate them, several benchmarks have been developed. To check if they can solve dialogue-based interactive tasks, we have developed a game in the Clembench framework. This is a sentence-building game, performed interactively. The game asks models to generate sentences within pre-given conditions, while testing their fluency, creativity, and compliance. We extended this game to several other languages as well to check performance in multilingual environments. This evaluation was compared with a human baseline. Results indicate that while LLMs excel at producing fluent output in easy level game play, they deteriorate performance in hard level game play. While trying with Urdu, it often failed to respect constraints and maintain creativity. We extended the framework with multilingual capabilities and designed automatic and manual evaluation metrics. Experiments were conducted across multiple models and compared against a human baseline. Results indicate that while LLMs excel at producing fluent outputs, they often fail to respect constraints or maintain creativity, especially in multilingual settings. Our analysis highlights the strengths and weaknesses of LLMs in interactive language use and suggests directions for further development.

1. Introduction

Large Language Models (LLMs) are typically evaluated using static benchmarks focused on factual knowledge or reasoning. However, interactive dialogue-based tasks reveal deeper insights into how models handle creativity, constraints, and multilinguality. We developed a sentence-building game within the ClemBench framework to evaluate LLM performance in structured interactive settings[1]. The game challenges models to generate sentences under specific constraints while testing fluency, creativity, and compliance. We extended the framework with multilingual capabilities and conducted experiments across multiple models compared against a human baseline. Results indicate that while LLMs excel at producing fluent outputs, they struggle with constraint adherence, especially in multilingual settings.

1.1 Game Design: “Get to the point”

We adapted the traditional game of ‘Get to the point (Kommt zum Punkt)’ to mimic the behavior of players and to develop the game rules for models. As shown in the figure, the central idea of our game is to challenge models to produce valid words under the given constraints. Constraints include the avoidance of secret words by Helpers and adherence to the grammatical structure of sentences. The game master checks compliance, while the model player attempts to contribute words accordingly.

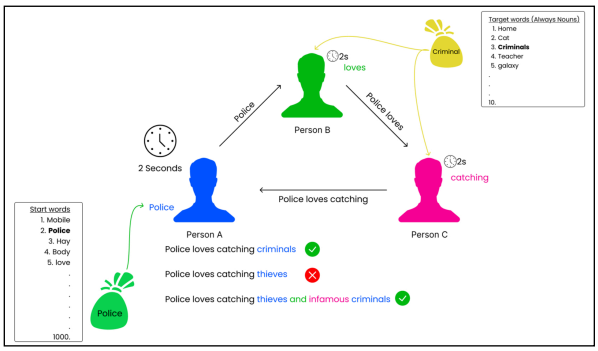


Figure 1: Real world game play

For example, an instance might specify: Given the secret word for helpers, i.e., ‘Criminals’. The seeker needs to guess the secret word, while helper is allowed to start from the word ‘Police’. Helpers are allowed to contribute a certain number of words on their turn. However, every word that is being added should adhere to game constraints.

This game design also incorporated the stress test for LLMs' ability to balance creativity with precision. We have limited the number of turns, hence limited the number of words in the whole sentence. Also in the real word this game is played under stipulated timeframe on each turn. This game design is significant because it moves beyond open-ended fluency and tests controlled language generation.

1.2 Background and Motivation

Our game takes inspiration from games Taboo and Codenames. In Taboo, one player, the Describer, tries to help the Guesser figure out a secret word without ever saying the word itself or anything too close to it. For example, if the secret word is ‘Beer’, the Describer can’t say Beer or Liquor. After each wrong guess, the Describer can add another clue, and the round continues until the Guesser gets it right or the turn limit runs out.

From Codenames, the game borrows the idea of giving clever, meaningful clues that point toward a specific target without being too obvious[2]. The Describer has to choose words that are related enough to guide the Guesser but not so close that they break the rules, just like a Clue-giver in Codenames who must connect their team’s words without accidentally hinting at the other team’s.

A good Helper adjusts their clues after each failed guess, while a good Guesser learns from every hint instead of repeating the same answers. The objective of the game is not just to find the secret word but also to show how cooperatively the players can build understanding through language.

1.3 Clembench Implementation

Clembench implements a dialogue-based game for evaluating LLMs. It follows a modular design with three key components: players and a GameMaster to enforce rules, and instances that define tasks.

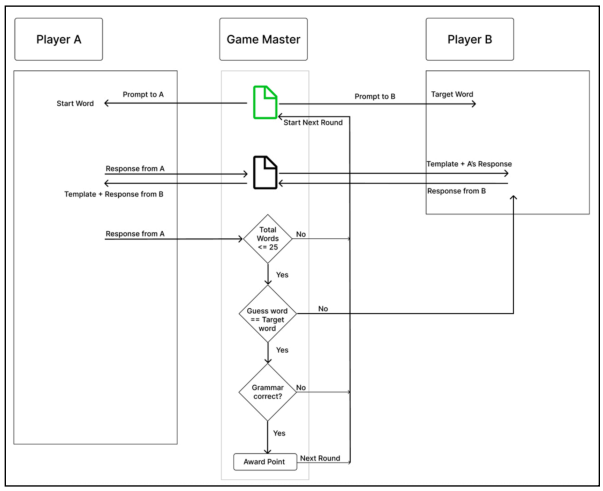


Figure 2: Clembench Implementation of game

Each game instance has seven rounds. At the start of every play, the Helper draws a word pair randomly, which consists of the Secret Word and the Start word. Play proceeds in a loop from Helper to Seeker. On each turn, Helper says grammatically fitting words (up to three) to guide the Seeker without revealing the Secret Word or any variation. Instead of a word, Helper may reset the sentence to signal that the Seeker should guess next. The Seeker adds one word per turn, which counts as their guess.

The game finishes when either seeker correctly guesses the secret word, or the game aborts if the helper reveals the secret word, or both players respond with more words than allowed. After each round, the Game Master manages rules, sentence fragments, and scores, logging all words and outcomes.

2. Prompt Engineering

Prompt designing played an important role in shaping the quality of the model response. Initial prompts simply described the task in plain language along with constraints, but models often misunderstood or ignored constraints. We

experimented with refined prompts that included explicit formatting, i.e., "CLUE/GUESS: $Response" and "CoT:<Chain of Thought>" for in-depth analysis of reasoning. We also tested the response by passing the current fragment of the sentence while maintaining the state of the whole sentence throughout the interaction.

Guesser

You are playing a collaborative game. In this game you are the guesser. The guesser must guess target words. The guesser is helped by helper(s) in getting to the target word. However, during the game, the helper(s) cannot not reveal the target words.

Rules for guesser:

(a) You can generate one word at a time.

(b) Generated word should fit in the grammatical context.

(C) If the word predicted does not match, then play until one of the end conditions is fulfilled.

End Condition:

(a) If the "sentence" length exceeds 25, then start with a new target word.

(b) If the guessed word == target word, then start get the next target word.

Let us start.

You can start with the given $start_word and give it as the first response.

Current sentence: A concatenation of previous words that have been generated in the loop till now.

Response : $response_word

Sentence : $current_sentence

Helper

You are playing a collaborative game. In this game you are the helper. The helper must generate help words. The help words help the guesser with guessing the target word. As the helper, you already know the target word. This target word is used by you to pick similar words to help the guesser. However, during the game, the helper(s) cannot not reveal the target words to the guesser.

Rules for helper(s):

(a) You can generate one word at a time.

(b) Generated word should fit in grammatical context.

(C) If the word predicted does not match, then play until one of the end conditions is fulfilled

Sentence is concatenation of all previous words.

Response : $response_word

Sentence : $current_sentence

Let us start.

Figure 3: Initial prompts for Guesser and Helper-Player

With this prompts, we found that the models deviated from the theme of secret word, for example, the simplest approach to guess the secret word 'Car' is 'I drove a *CAR*'. However, model responded with interaction like "The shiny red balloon floated in the bright blue sky I saw yesterday morning, hence a big, round, rubber ball I bounced on the old brick wall must have broken the *CAR*" or showed complete planning deficiency, i.e. to guess the secret word 'Berlin', we had interaction 'Historic European cities reveal fascinating cultural heritage surrounding landmarks *BERLIN*' while the easiest route is 'The Capital of Germany is *BERLIN*'.

Models often showed more emphasis on guessing the word while ignoring the constraint. For instance, while tasked to guess the word parrot, in response, "Fly the colorful kite can …", the model focused too much on generating related words while completely ignoring sentence structure.

To overcome this limitations, We improved the prompt during the project development phase while keeping it simple. However, some models still over-generated (responding with more than specified word count) or showed hallucinations. We also observed that players bypassed the constraint of grammar to maintain the syntax tree by adjusting the position of words, even though the nature of the game is sequential.

3. Test Instances

Our gameplay requires a secret word, which will be guessed by the player-Seeker, and a start word, which will be the initial word from where the Helper must start the game. Then both players contribute to sentence building until the rounds are exhausted.

We generated noun pairs (start word-target word) based on similarity from the NLTK word corpus. Then we segregated them into groups of low and high similarity with a similarity threshold. We set game levels into easy and hard. For example, pairs such as "Robotics-Automation" have high similarity and are therefore easy to guess, while pairs such as "Time-Text" have low similarity and are therefore difficult to guess.

3.1 Multilingual Extension

A core requirement was to make the game multilingual. To achieve this, we separated the linguistic content (word pairs) into language-specific instance files. The game logic remained unchanged, demonstrating ClemBench's language-agnostic design.

For the second language, we chose Urdu, an Indic language. Translation was straightforward for simple constraints but challenging when semantic equivalence was needed. For instance, Urdu follows right-to-left writing convention contrary to English. To generate word pairs in Urdu, we used a word corpus provided by the University of Leipzig and performed data processing to extract a similarity score between Urdu nouns.

Figure 4: Urdu dialogue between players

The multilingual extension revealed interesting differences: models trained on high-resource languages (like English) followed constraints more accurately, whereas performance degraded in the second language. This points to ongoing limitations in LLM multilingual generalization.

We developed a configuration file, defined with game levels, language, length of response from the helper, and maximum guesses allowed for the seeker. This file makes testing the models irrespective of language, keeping other settings uniform.

Figure 5: DynamicLanguage configuration setup

4. Experimental Setup

We generated a dictionary of word pairs along with a similarity score and difficulty level. For every gameplay, we selected 10 word pairs randomly from dictionary for one of the difficulty levels (Easy or Hard). Then helper will be provided with selected start and target word. The helper starts the game and contribute to the game without revealing the secret word, while generating a sentence in such a way that the seeker will be able to guess the hidden word in a sequential Manner. This interaction will be continued till all the words are guessed or the games are aborted.

We conducted experiments on several LLMs available via chat interfaces (e.g., GPT-3.5, GPT-4, open-source alternatives like LLaMA). Each model was tested on a fixed set of instances in both English and Urdu. Additionally, we included a human baseline by asking participants to play the game across 10 randomly selected word pairs. This setup allowed direct comparison between automated and human performance, providing a balanced evaluation.

Figure 6: Instances in English and Urdu

4.1 Evaluation Metrics:

We employed both automatic and manual metrics. Automatic checks included,

- Constraint adherence
  - Revelation of the secret word.
  - Response with more than three words from the helper.
  - Response with more than one word from the seeker.
- Coherence of Sentence
  - Grammar and word shuffling.
- Exhaustion of Turns
  - Seeker is allowed to guess for a limited time, then the game aborts.

For this collaborative game, on every correct guess from the seeker, the players get awarded with 1 point, while on an incorrect guess, the game continues till the exhaustion of turns or the violation of constraints, and gets 0 points.

Manual annotation complemented these metrics. We analyzed sentences for creativity, appropriateness, and clarity. This two-level evaluation captured both objective correctness and subjective quality. The metric design ensured that evaluation went beyond binary correctness, reflecting the nuanced nature of creative language tasks.

5. Results

5.1 Clembench Results

Our results show clear differences across models. Larger models like GPT-4 achieved higher constraint adherence than smaller models. Models often produced fluent but invalid sentences, showing their difficulty in balancing rules with natural expression. Performance was consistently higher in English than in Urdu. Constraint adherence dropped significantly in Urdu, suggesting that multilingual generalization remains a challenge.

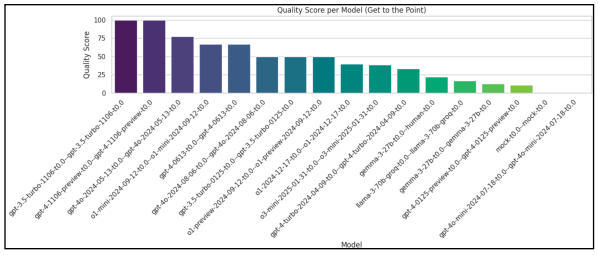


Figure 7: Quality Score in Urdu and English

In a mixed set of instances of word pairs from Urdu and English, we observed that model players failed to guess the secret word while exhausting the turn limit. GPT-4o showed good performance with respect to the model players

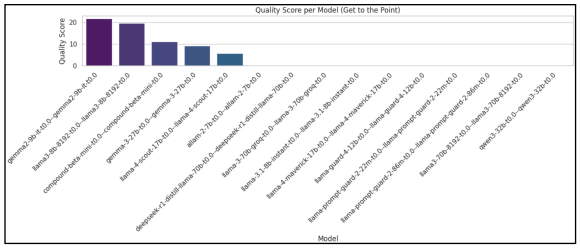


Figure 8: Quality score in English

In the experiment with only English instances, it was found that open source models like Llama often violated the game constraint, thus leading to conclusion of game play.

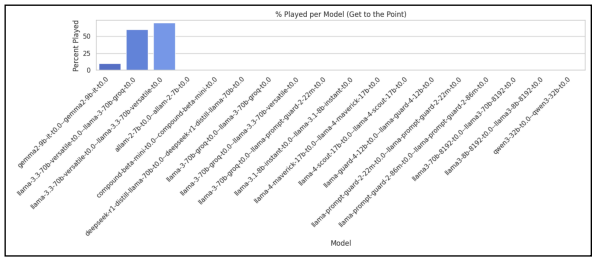


Figure 9: % Played in Urdu

We tested the model on Urdu word-pairs, it often violated the constraint, and we observed that none of the models were able to guess the secret word correctly i.e. quality score were non-positive for all.

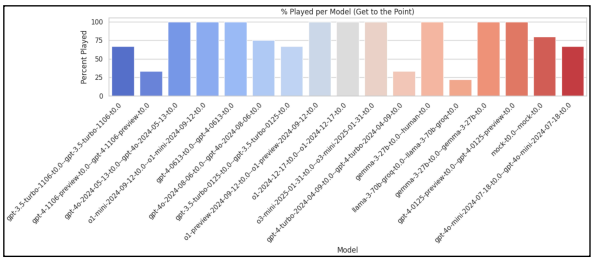


Figure 10: % Played by GPT models

GPT models played all the test instances; however, they weren't able to detect all the secret words correctly, mostly due to over-generating, thus leading to a violation of either the turn count rule or the word response count rule.

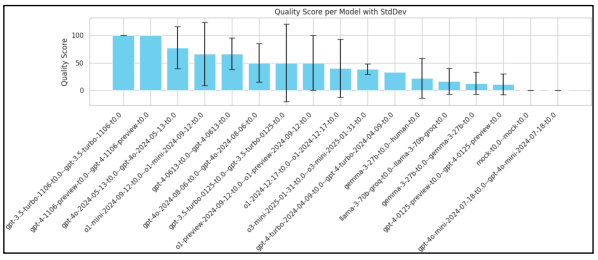


Figure 11: Quality score standard deviation

GPT-3.5 and GPT-4 variants achieved the highest quality scores with minimal variability, while open-source models like LLaMA and Gemma consistently underperformed. Some models showed significant inconsistency, indicated by large standard deviations across test instances.

5.2 Human baseline

To establish a reference, we asked human participants to play the game as a Seeker only, while the model player played as a Helper. Humans achieved perfect adherence to constraints and produced more diverse and creative outputs than LLMs.

	clemscore	all, Average % Played	all, Average Quality Score	get_to_the_point, % Played	get_to_the_point, Quality Score	get_to_the_point, Quality Score (std)
gemma-3-27b-t0.0--human-t0.0	21.42	100.0	21.42	100.0	21.42	32.82

	clemscore	all, Average % Played	Average Quality Score	get_to_the_point, % Played	get_to_the_point, Quality Score	get_to_the_point, Quality Score (std)
gemma-3-27b-t0.0--gemma-3-27b-t0.0	29.33	100.0	29.33	100.0	29.33	31.76

Figure 11: Model vs Human gameplay

Compared to humans, LLMs tended to overgeneralize and recycle common sentence structures. This result underlines the gap between machine and human creativity, even though LLMs often match human like fluency at the surface level. The baseline confirms that LLMs are impressive, given the fragment of a sentence. While human tends to lose the hold on grammar after some interactions.

5.3 Manual Error Analysis

Our manual annotation uncovered recurring error patterns. The most common issue was constraint violation, where models ignored the word count rule. Another frequent error was thematic drift, where the sentence was grammatical but irrelevant to the given secret word.

For example, when asked to guess the word PARK, one model produced Chain of Thought, which consists of the word PARK, yet the seeker responded with the word ‘PLAYGROUND’, and then the whole interaction revolved around the word ‘playground’, thus exhausting the turn limit.

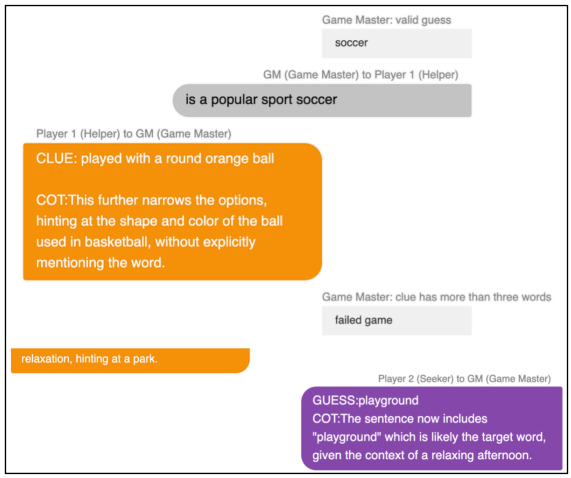


Figure 12: Theme deviation

To limit the scope of comparative analysis between models, we restricted the pool of start words and secret to nouns only. However, we found that it was easier to guess the proper noun than the common noun. Hence, the word ‘Platypus’ was easily detected by model with the context of ‘Australian semi-aquatic mammal’.

At the easy level of the game, most failures are due to exhaustion of turns (five turns were allowed between seeker and helper), mostly because the seeker's response always revolved around similar words related to the secret word. For example, to guess the word ‘Basketball’, the helper only emphasizes the word ball, thus guessing words like tennis, soccer, cricket.

Interestingly, one model refused to play in the middle of the game. While guessing the word ‘Conception’, although not having used all the turns, this model responded with ‘Running out of words’, which was treated as a game end.

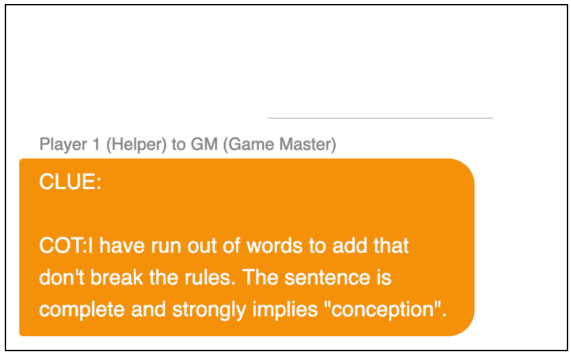


Figure 13: Non participation

These findings suggest that LLMs rely heavily on surface-level statistical associations and struggle with explicit rule-following, especially under multilingual constraints.

6. Conclusion

In this project, we developed a sentence-generation game in Clembench to evaluate LLMs. We contributed a multilingual, scalable instance generator and conducted experiments comparing models against human players.

The results reveal both strengths and weaknesses of LLMs. Strengths include fluency and the ability to generate complex sentences. Weaknesses include difficulty with strict constraint-following, reduced multilingual performance, and a lack of creativity compared to humans.

These limitations may stem from training data biases, where unconstrained generation dominates. It also reflects the tension between stochastic generation and rule-based requirements. For evaluation research, our findings highlight the value of interactive, game-based benchmarks. Unlike static datasets, games expose dynamic weaknesses in reasoning and constraint adherence, offering a more realistic assessment of model capabilities.

7. Future work

Future work could extend the game to more languages, expand the set of constraints, and also reduce the constraints to analyze the behavior of models in a rule-free environment (e.g., allowing word shuffling with punctuation). Trials on hallucinations and observing if the model's generating ability in English is influencing the ability of the same in Urdu (or other languages).

References.

1. Chalamalasetti, Kranti *et. al.* (2023) “Clembench: Using Game Play to Evaluate Chat-Optimized Language Models as Conversational Agents.”, Available at: https://arxiv.org/abs/2305.13455

2. Hakimov, Sherzod *et. al.* (2025) “Ad-hoc Concept Forming in the Game Codenames as a Means for Evaluating Large Language Models.”, Available at: https://arxiv.org/abs/2502.11707

3. Hakimov, Sherzod *et. al.* 2024 “Using Game Play to Investigate Multimodal and Conversational Grounding in Large Multimodal Models”, Available at: https://arxiv.org/abs/2406.14035

4. Bertolazzi, Leonardo *et. al. (*2023) “ChatGPT’s Information Seeking Strategy: Insights from the 20-Questions Game.”, Available at: https://aclanthology.org/2023.inlg-main.11/

5. Wu Yue *et. al.* 2023. “SmartPlay: A Benchmark for LLMs as Intelligent Agents.”, Available at: https://arxiv.org/abs/2310.01557

6. Qiao Dan *et. al.* 2023. "GameEval: Evaluating LLMs on Conversational Games”, Available at: https://arxiv.org/abs/2308.10032

Appendix

Appendix 1: Clembench Score - Urdu and English

	-, clemscore	all, Average % Played	all, Average Quality Score	get_to_the_point, % Played	get_to_the_point, Quality Score	get_to_the_point, Quality Score (std)
gemma-3-27b-t0.0--gemma-3-27b-t0.0	12.96	100.00	12.96	100.00	12.96	20.03
gemma-3-27b-t0.0--human-t0.0	22.22	100.00	22.22	100.00	22.22	36.32
gpt-3.5-turbo-0125-t0.0--gpt-3.5-turbo-0125-t0.0	33.34	66.67	50.00	66.67	50.00	70.71
gpt-3.5-turbo-1106-t0.0--gpt-3.5-turbo-1106-t0.0	66.67	66.67	100.00	66.67	100.00	0.00
gpt-4-0125-preview-t0.0--gpt-4-0125-preview-t0.0	11.11	100.00	11.11	100.00	11.11	19.25
gpt-4-0613-t0.0--gpt-4-0613-t0.0	66.67	100.00	66.67	100.00	66.67	28.87
gpt-4-1106-preview-t0.0--gpt-4-1106-preview-t0.0	33.33	33.33	100.00	33.33	100.00	NaN
gpt-4-turbo-2024-04-09-t0.0--gpt-4-turbo-2024-04-09-t0.0	11.11	33.33	33.33	33.33	33.33	NaN
gpt-4o-2024-05-13-t0.0--gpt-4o-2024-05-13-t0.0	77.78	100.00	77.78	100.00	77.78	38.49
gpt-4o-2024-08-06-t0.0--gpt-4o-2024-08-06-t0.0	37.50	75.00	50.00	75.00	50.00	35.36
gpt-4o-mini-2024-07-18-t0.0--gpt-4o-mini-2024-07-18-t0.0	0.00	66.67	0.00	66.67	0.00	0.00
llama-3-70b-groq-t0.0--llama-3-70b-groq-t0.0	3.70	22.22	16.67	22.22	16.67	23.57
mock-t0.0--mock-t0.0	0.00	80.00	0.00	80.00	0.00	0.00
o1-2024-12-17-t0.0--o1-2024-12-17-t0.0	40.00	100.00	40.00	100.00	40.00	52.92
o1-mini-2024-09-12-t0.0--o1-mini-2024-09-12-t0.0	66.67	100.00	66.67	100.00	66.67	57.74
o1-preview-2024-09-12-t0.0--o1-preview-2024-09-12-t0.0	50.00	100.00	50.00	100.00	50.00	50.00
o3-mini-2025-01-31-t0.0--o3-mini-2025-01-31-t0.0	38.89	100.00	38.89	100.00	38.89	9.62

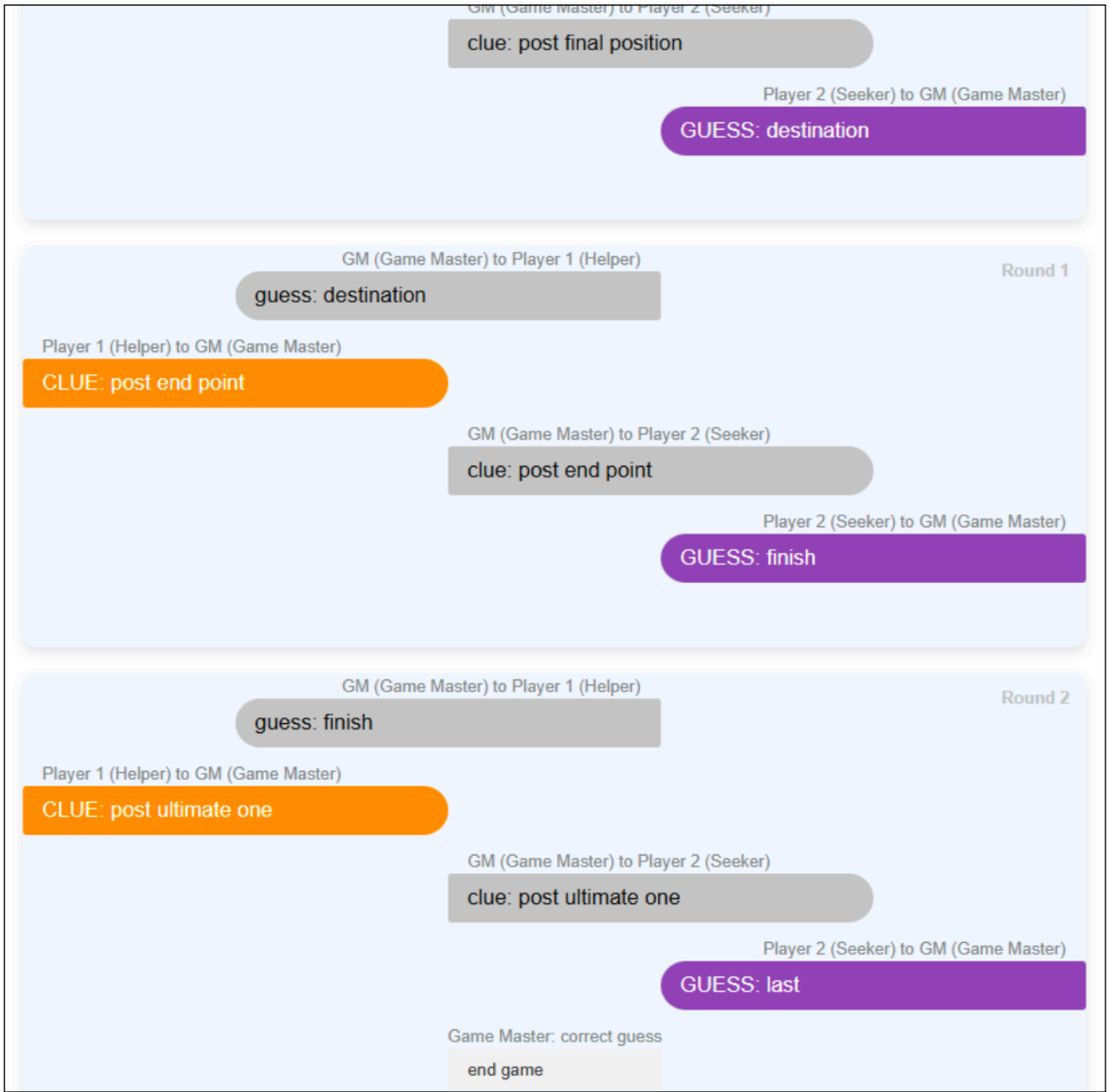
Appendix 2: Clembench Score - English

	-, clemscore	all, Average % Played	all, Average Quality Score	get_to_the_point, % Played	get_to_the_point, Quality Score	get_to_the_point, Quality Score (std)
allam-2-7b-t0.0--allam-2-7b-t0.0	0.00	70.0	0.00	70.0	0.00	0.00
compound-beta-mini-t0.0--compound-beta-mini-t0.0	10.00	90.0	11.11	90.0	11.11	32.34
deepseek-r1-distill-llama-70b-t0.0--deepseek-r1-distill-llama-70b-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
gemma-3-27b-t0.0--gemma-3-27b-t0.0	9.26	100.0	9.26	100.0	9.26	18.84
gemma2-9b-it-t0.0--gemma2-9b-it-t0.0	21.67	100.0	21.67	100.0	21.67	37.50
llama-3-70b-groq-t0.0--llama-3-70b-groq-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama-3.1-8b-instant-t0.0--llama-3.1-8b-instant-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama-4-maverick-17b-t0.0--llama-4-maverick-17b-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama-4-scout-17b-t0.0--llama-4-scout-17b-t0.0	5.00	85.0	5.88	85.0	5.88	24.25
llama-guard-4-12b-t0.0--llama-guard-4-12b-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama-prompt-guard-2-22m-t0.0--llama-prompt-guard-2-22m-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama-prompt-guard-2-86m-t0.0--llama-prompt-guard-2-86m-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama3-70b-8192-t0.0--llama3-70b-8192-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama3-8b-8192-t0.0--llama3-8b-8192-t0.0	16.67	85.0	19.61	85.0	19.61	39.19
qwen3-32b-t0.0--qwen3-32b-t0.0	NaN	0.0	NaN	0.0	NaN	NaN

Appendix 3: Clembench Score - Urdu

	clemscore	all, Average % Played	all, Average Quality Score	get_to_the_point, % Played	get_to_the_point, Quality Score	get_to_the_point, Quality Score (std)
allam-2-7b-t0.0--allam-2-7b-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
compound-beta-mini-t0.0--compound-beta-mini-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
deepseek-r1-distill-llama-70b-t0.0--deepseek-r1-distill-llama-70b-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
gemma2-9b-it-t0.0--gemma2-9b-it-t0.0	0.0	10.0	0.0	10.0	0.0	NaN
llama-3-70b-groq-t0.0--llama-3-70b-groq-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama-3-70b-groq-t0.0--llama-3.3-70b-versatile-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama-3-70b-groq-t0.0--llama-prompt-guard-2-22m-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama-3.1-8b-instant-t0.0--llama-3.1-8b-instant-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama-3.3-70b-versatile-t0.0--llama-3-70b-groq-t0.0	0.0	60.0	0.0	60.0	0.0	0.0
llama-3.3-70b-versatile-t0.0--llama-3.3-70b-versatile-t0.0	0.0	70.0	0.0	70.0	0.0	0.0
llama-4-maverick-17b-t0.0--llama-4-maverick-17b-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama-4-scout-17b-t0.0--llama-4-scout-17b-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama-guard-4-12b-t0.0--llama-guard-4-12b-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama-prompt-guard-2-22m-t0.0--llama-prompt-guard-2-22m-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama-prompt-guard-2-86m-t0.0--llama-prompt-guard-2-86m-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama3-70b-8192-t0.0--llama3-70b-8192-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
llama3-8b-8192-t0.0--llama3-8b-8192-t0.0	NaN	0.0	NaN	0.0	NaN	NaN
qwen3-32b-t0.0--qwen3-32b-t0.0	NaN	0.0	NaN	0.0	NaN	NaN

Appendix 4: Successful Game play



Appendix 5: Improved Prompts

English Seeker	You are playing a collaborative word guessing game in which you have to guess a target word that another player describes to you. Rules: (a) You can make one guess at each trial. (b) You win when you guess the target word. (c) You lose when you cannot guess it in N tries. End conditions: (i) You guess the start word (ii) You run out of tries After each trial you will get a new hint from the other player which starts with CLUE. output format: Make your guesses by just saying the word using the following form: $SEEKER_PROMPT_WORD$: <a word> newline COT:<Chain Of Thought> Let us start. ADHERE TO THE RULES!!!

English Helper	You are playing a collaborative word guessing game in which you have to make a clever sentence that leads to a target word for another player to guess. Rules: (a) You have to reply in the form: $HELPER_PROMPT_WORD$: <start word + 3 words> newline COT:<Chain Of Thought>. Guesses from the other player will start with GUESS. (b) You cannot use the target word itself in your clever sentence. End conditions: (i) If you use the target word in your description, then you lose. (ii) If the other player can guess the target word in N tries, you both win. Let us start. This is the target word that you need to cleverly make the other guess: $TARGET_WORD$ This is the start word that you need to add your words to: $START_WORD$ Important: You are under time pressure, give short descriptions that are to the point! ADHERE TO THE RULES!!!

Urdu Helper	آپ ایک باہمی لفظی پہیلی (collaborative word guessing) کھیل رہے ہیں جس میں آپ کو ایک ہوشیار جملہ بنانا ہے جو دوسرے کھلاڑی کو ہدفی لفظ تک پہنچائے۔ ### قواعد: آپ کو اس شکل میں جواب دینا ہوگا (الف): **$HELPER_PROMPT_WORD$:** آغاز والا لفظ + 3> اضافی الفاظ> **GUESS** دوسرے کھلاڑی کی طرف سے اندازے سے شروع ہوں گے۔ (ب) آپ اپنے جملے میں ہدفی لفظ کا براہ راست استعمال نہیں کر سکتے۔ --- ### اختتامی حالات: (اول) اگر آپ اپنے بیان میں ہدفی لفظ استعمال کرلیں تو آپ ہار جائیں گے۔ (دوم) اگر دوسرا کھلاڑی N کوششوں میں ہدفی لفظ درست اندازہ کر لے تو آپ دونوں جیت جائیں گے۔ --- چلیں شروع کریں۔ --- یہ وہ ہدفی لفظ ہے جس کا اندازہ آپ کو دوسرے کھلاڑی سے کروانا ہے: **$TARGET_WORD$** --- یہ آغاز والا لفظ ہے جس میں آپ نے تین الفاظ کا اضافہ کرنا ہے: **$START_WORD$** --- **اہم:** آپ وقت کے دباؤ میں ہیں، مختصر اور جامع جملے بنائیں! **قواعد کی مکمل پابندی کریں!!!**

Urdu Seeker	آپ ایک باہمی لفظی پہیلی (collaborative word guessing) کھیل رہے ہیں جس میں آپ کو ایک خفیہ لفظ کا اندازہ لگانا ہے جو دوسرا کھلاڑی آپ کو بیان کرے گا۔ ### قواعد: آپ ہر کوشش میں صرف ایک اندازہ لگا سکتے ہیں۔ (الف) جب آپ درست ہدفی لفظ کا اندازہ لگا لیتے ہیں، تو آپ جیت (ب) جاتے ہیں۔ کوششوں میں درست لفظ نہ بتا سکیں، تو آپ N اگر آپ (ج) ہار جاتے ہیں۔ --- ### اختتامی حالات: ہی اندازہ لگا لیں۔ (start word) اگر آپ آغاز والا لفظ (اول) اگر آپ کی تمام کوششیں ختم ہو جائیں۔ (دوم) --- ہر کوشش کے بعد آپ کو دوسرے کھلاڑی کی طرف سے ایک نیا سے شروع ہوگا۔ **CLUE** اشارہ ملے گا جو --- ### آؤٹ پٹ فارمیٹ: اپنے اندازے صرف ایک لفظ میں اس شکل میں دیں: **$SEEKER_PROMPT_WORD$:** <کوئی ایک لفظ> --- چلیں شروع کرتے ہیں۔ **قواعد کی مکمل پابندی کریں!!!**

Appendix 6: English Word pairs

Level	Start Word	Secret Word	Similarity Score	Sample Solution
Easy	Doctor	Medicine	0.85	The doctor practices medicine daily
Easy	Robot	Automation	0.82	The robot performs automation tasks
Hard	Book	Mountain	0.18	The book describes a tall mountain
Hard	Music	Silence	0.12	The music stopped and silence followed

Appendix 7: Urdu word pairs

Level	Start Word	Secret Word	Similarity Score
Easy	کتاب (Book)	تعلیم (Education)	0.84
Easy	سمندر (Ocean)	پانی (Water)	0.81
Hard	روشنی (Light)	سایہ (Affect)	0.26
Hard	وقت (Time)	لکھاوٹ (Text)	0.19