ソルバー付き大規模言語モデルによる麻雀局面に対する多事実列挙の検証

土田 蒼大[1,2,a)]　小橋 洋平[2,3]

概要：麻雀の局面について盤面から一意に決まる事実（打牌後の向聴数・有効牌・待ちの形）を，大規模言語モデル（LLM）がどこまで誤りも漏れもなく日本語で列挙できるかを，LLM も人手も用いない決定論的な判定器で測定した．本稿では，打牌と打牌確率を返す凍結した麻雀 AI（Mortal）をソルバーと呼び，向聴数や有効牌は麻雀のルールに従うルール計算器で計算する．両者の出力を入力に与えた Qwen3-1.7B（LoRA・1 seed）は，未使用のテスト用 1,500 局面のうち 1,427（95.1%）で単回生成のまま合格し，必須の事実の数が増えても合格率は大きく下がらなかった．これらの出力は，学習した射影層で soft tokens として与えるよりテキストで与える方が合格数が多かった（1 seed）．また，Mortal の中間表現から反実仮想の打牌を予測する MLP プローブを学習し，その予測を言語モデルに attention bias として加えると，他家のリーチが無い場合の Mortal の打牌を言語化できた（開発用・3 seed）．さらに状況に応じて要点から話す解説へ拡張し，牌の名称を 1 トークンとする専用トークンを導入した．正確さの差は検出されず，生成ステップ数は 1.9〜9.3% 減った（3 seed）．

キーワード：麻雀，説明生成，忠実性検証，大規模言語モデル，評価プロトコル

Verifying Multi-Fact Enumeration by a Solver-Augmented Large Language Model on Mahjong Positions

Souta Tsuchida[1,2,a)]　Yohei Kobashi[2,3]

Abstract: We measure how completely and faithfully a large language model (LLM) can enumerate, in Japanese, board-determined facts about Mahjong positions (shanten, useful tiles, waiting shapes), using a deterministic judge that involves neither an LLM nor a human. We use a frozen Mahjong agent (Mortal) as the solver for moves and move probabilities, and a rule-based calculator for the facts. Given their outputs, Qwen3-1.7B with LoRA passed 1,427 of 1,500 held-out positions in a single greedy generation (one seed), without a sharp drop as the number of required tiles grew. Passing the outputs as text outperformed passing them as soft tokens from a learned projector (one seed). An MLP probe trained on the agent's intermediate representations, whose predictions bias the LLM's attention, let the model verbalize the agent's counterfactual move without the opponent's riichi (three seeds). We also extend the task to situation-dependent commentary with dedicated single-token tile names (three seeds; no detectable accuracy difference, 2–9% fewer generation steps).

Keywords: Mahjong, explanation generation, faithfulness verification, large language models, evaluation protocol

[1] N 高等学校
N High School
[2] 東京大学松尾研 LLM コミュニティ LLMATCH
LLM Community "LLMATCH", Matsuo–Iwasawa Laboratory, The University of Tokyo
[3] 東京大学大学院工学系研究科技術経営戦略学専攻
Department of Technology Management for Innovation, Graduate School of Engineering, The University of Tokyo
[a)] nynanekinoaka@gmail.com

1. はじめに

麻雀 AI は人間を上回る打牌を返すが，その根拠を言葉で示さない．大規模言語モデル（LLM）に解説させると流暢な文は得られるが，向聴数や有効牌のような計算で決まる事実を誤る．その文を人手や LLM で判定すると，評価

© 2026 Information Processing Society of Japan

そのものが揺らぐ．

本研究の目的は，計算済みの事実を言語モデルに文章化させる際の誤りと欠落を，正解を機械的に定められる麻雀局面で測定することである．本稿では，凍結した麻雀 AI の Mortal[27] をソルバーと呼ぶ．ソルバーは局面に対する打牌と打牌確率を返し，盤面に介入して再推論させた結果も返す．向聴数・有効牌・待ちの形・役と点数などは，麻雀のルールに従うルール計算器で計算する．将来的には打ち手らしい語り口を伴う解説を目指すが，本稿はその前段として事実の正確さを測る．

本稿の貢献は次の四つである．第一に，誤った文と必須の欠落を許容しない合格条件を持つ決定論的な判定器と，組成を明示した評価セットを整備し，その監査結果を示す．第二に，ソルバーとルール計算器の出力を入力に与えれば 1.7B のモデルでも与えた事実をほぼ漏れなく写せ，事実の数が増えても合格率が大きく下がらないこと，これらの出力は学習した射影層による soft tokens よりテキストで与える方が合格数が多いことを示す（5.1 節・5.2 節）．第三に，Mortal の中間表現から反実仮想の打牌を予測する MLP プローブ（MLP probing）を学習し，その予測を言語モデルに attention bias として加えると，反実仮想の打牌を言語化できることを示す（5.3 節）．第四に，状況に応じて要点から話す解説へ拡張し，牌の名称を 1 トークンとして扱う専用トークンを導入したモデルを構築する（5.4 節）．なお，対象の事実は定型文で直接出力すれば原理的に正しく書けるので，本稿は LLM を介する構成の応用上の優位を主張しない．

2. 関連研究

ゲームの解説生成では，将棋 [2], [16], [17] とチェス [3], [9] で，指し手や探索の結果を言語モデルに与えて解説を生成する研究がある．麻雀では，AI の判断を代理モデルや寄与度で説明する研究 [1], [6], [7] と，指示学習で解説を生成し LLM を判定者とする研究 [5] がある．

外部の表現を言語モデルに渡す方法としては，画像やグラフの特徴を射影して soft tokens とする構成 [18], [19]，ゲームエンジンの潜在表現を渡す構成 [10]，言語モデル自身の隠れ表現を言語化する研究 [11], [20] がある．介入により AI の振る舞いを説明する研究 [13], [14], [21] は，本稿の反実仮想と考え方が近い．

生成文の事実性は，構造化データとの照合 [8], [22] や原子的な主張への分解 [4], [12] で評価されてきたが，いずれも判定にモデルや人手を介する．学習の面では，検証器で選別したデータでの再学習 [23] や，新しいトークンを元のサブワードの平均で初期化する語彙の拡張 [24], [25], [26] が本稿に関係する．本稿は，ソルバーとルール計算器の出力を正解として生成文を文ごとに決定論的に照合し，判定に LLM も人手も用いない点でこれらと異なる．

3. 手法

本稿の構成を図 1 に示す．ソルバーとルール計算器の出力を LLM の入力と判定の両方に用いる構成を「ソルバー付き大規模言語モデル」と呼ぶ．生成文を 1 文ごとにこれらの出力と照合し，真偽と必須の欠落を返すプログラムを判定器と呼ぶ（学習も乱数も LLM も用いない）．凍結はパラメータを更新しないこと，層別は局面を必須の牌の種類数などで群に分けて集計することを指す．最終潜在表現 phi は Mortal が行動価値を出す直前の 1,024 次元のベクトル，色別 4 トークンは有効牌の集合（34 種）を萬子・筒子・索子・字牌の 4 本に分けて LLM の入力空間に射影したもの（手牌も同様）である．

3.1 Mortal をソルバーとして使う

本稿では，凍結した麻雀 AI である Mortal[27] を，打牌の正解を返すソルバーとして扱う．Mortal からは三つを得る．第一は打牌と，合法手だけで正規化した打牌確率である．推奨の強さ（はっきり・やや・わずか）と「迷う」の区分は，この確率で決める．第二は内部表現（最終潜在表現 phi と ResBlock の出力）である．第三は，他家のリーチ宣言を牌譜から除いた同じ局面で再推論させた結果である．これにより「リーチが無ければ Mortal は Y を切る」という反実仮想の文の真偽が，人の判断を介さずに決まる．向聴数・有効牌・待ちの形・役と点数・順位など盤面から一意に決まる事実は，ルール計算器で計算する．Mortal について書く文は，意図ではなく打牌確率に現れた評価として述べる（例「Mortal は一萬切りをやや推す」）．

凍結した Mortal の打牌と内部表現，ルール計算器が計算した事実を言語モデルに渡し，LoRA[29] で学習させて解説文を生成する．Mortal の重みは Mortal の作者以外が訓練したものを用いた．本稿で扱う三つのモデルの対応を表 1 にまとめる．

3.2 言語モデルと入力の渡し方

射影層（Projector）は，Mortal の打牌確率（3 本），最終潜在表現 phi と盤面の CNN 特徴をまとめた 1 本，打牌，ドラ表示牌，Q 値の上位の差（1 本），有効牌と手牌（萬子・筒子・索子・字牌に分けて各 4 本）から計 15 本のベクトルを作り，言語モデルの入力の先頭に並べる（soft tokens）．学習するのは Projector と LoRA で，Mortal は凍結する．言語モデルは Qwen3-1.7B[28] である．

3.3 学習データの作り方

規則で作る正解文．v1 の課題と拡張課題では，ルール計算器とソルバーの出力を定型文に当てはめて正解文を作る

© 2026 Information Processing Society of Japan

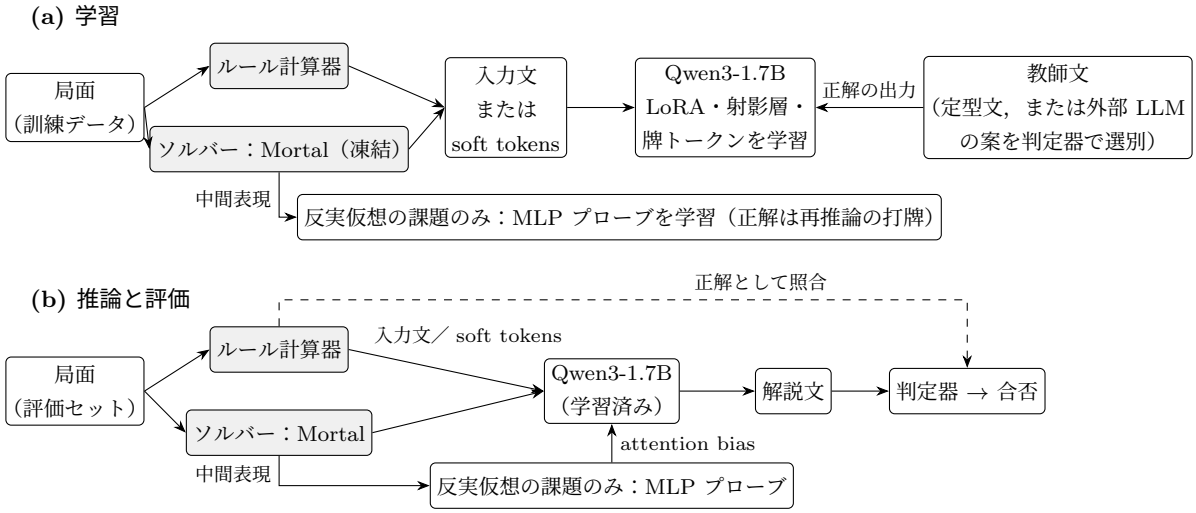


図 1 本稿の構成．(a) 学習：ルール計算器とソルバー（凍結した Mortal）の出力を入力文または soft tokens として与え，教師文を正解として LoRA などを学習する．MLP プローブは反実仮想の課題（5.3 節）でのみ用い，Mortal の中間表現から反実仮想の打牌を予測する．他の課題では使わない．(b) 推論と評価：学習済みのモデルが生成した解説文を，判定器がルール計算器の結果と照合する（破線．Mortal の評価を述べる文はソルバーの出力と照合する）

表 1 本稿で扱う三つのモデルの構成（言語モデルはいずれも Qwen3-1.7B）

	事実の列挙 v1（5.1 節）	拡張課題 v2.1（約 9 文の解説・診断のみ）	状況に応じた解説（5.4 節）
学習	LoRA 末尾 16 層	同左（v1 の採用モデルから継続）	LoRA 全 28 層・牌トークン
入力	soft tokens 15 本＋打牌の one-hot・打牌後 13 枚＋文字の 1 行（打牌後の向聴・聴牌時の待ち）	同左＋計算した事実の入力文	計算した事実の入力文（soft tokens なし）
seed	1 seed	1 seed	3 seed

（課題と評価仕様は 4 章）．文型の数が限られるので，学習したモデルの出力も同じ文型の列になる．

LLM が書いた教師文．状況に応じて要点から話す解説（5.4 節）では，教師文を外部の LLM に書かせ，判定器で選別した（他の課題では教師 LLM を用いていない）．事実の一覧を並べた入力文と，局面の状況から規則で決めた「先に言う事実の順」を渡して書かせ，順番の規則は教師にだけ渡し学習するモデルには渡さない．判定器が偽とした文，入力文に無い牌名や数字，意図や理由の言い方，判定器が照合できない文のいずれかを含む案は捨てた．まず GPT-6 Luna に書かせた教師文の表記の偏りを規則で直し，接続詞（「なお」「一方」など）を加えた．次に，追加した 6 状況（4 章）の局面と誤りの多かった型の局面の教師文を GPT-6 Astra と Claude（Opus 5.5・Sonnet 5.5）に同じ条件で書かせ，同じ選別を通した．最終の教師文は 10,518 局面（14,566 行）で，教師ごとの局面数は GPT-6 Luna 6,611・Claude Opus 2,368・Claude Sonnet 1,039・GPT-6 Astra 500 である．全行を判定器で照合し，偽は 0 だった．

入力: 手牌 一萬五萬六萬八萬八萬九萬二筒六筒六筒一索五索七索九索九索／ドラ表示 四筒／ 3 巡目／ Mortal の打牌 一索
出力: 「打一索．五萬六萬で四萬・七萬の両面受け．五索七索で六索のカンチャンの受け．」
判定: 打牌一致・受けの 2 文とも真・必須の受け牌（四萬・七萬・六索）を全て網羅 → 合格

図 2 入力・出力・判定の 1 組（事実の列挙 v1 の課題）

4. 実験計画

4.1 課題

入出力の実例．図 2 に，入力（打牌前の手牌 14 枚・ドラ表示牌・Mortal の打牌），出力の文，判定の 1 組を示す（開発用の局面から選んだ）．出力は句点で区切った短文の列で，1 文目は打牌の宣言，以降は受けの形または待ちの形を述べる．判定は文ごとに真偽を返し，最後に必須の受け牌がすべて述べられたかを調べる．

四つの課題．本稿は次の四つの課題を扱う．(1) 事実の列挙（v1）：打牌と必須の受け牌・待ちの形を偽なく漏れ

© 2026 Information Processing Society of Japan

なく書く．(2) 拡張課題（v2.1）：守り・向聴・枚数・役・点数・Mortal の評価などを含む約 9 文の解説を書く．(3) 反実仮想：他家のリーチがある局面で，リーチが無ければ Mortal の打牌が変わるかを 1 文で書く（正解は再推論で決まる）．(4) 状況に合わせて要点から話す解説：局面を 6 つの状況（他家リーチと自分の聴牌が重なる／自分のリーチ／聴牌／他家リーチ／ Mortal が迷う／それ以外）に分け，状況ごとに先に言う事実（それぞれ最大 3 つ）を規則で決める．最終の教師文では，リーチの絡まない局面向けに，アガれない待ち（フリテン・役なし）／オーラスの順位／鳴いて聴牌／鳴いた手の役の形／七対子／向聴を戻す打牌の 6 つを加えた．たとえば他家リーチ中なら打牌の安全さ，聴牌なら待ちを先に言う．解説は 3〜6 文の自由な語順で書かせる．

4.2 判定器と監査

評価仕様 v1.0．正解文と判定器はこの仕様に従う．使う情報は自分の手牌・ドラ表示牌・Mortal の打牌・有効牌だけで，推測は入れない．有効牌は，引くと向聴数が下がる牌（聴牌なら和了牌）のうち見えている牌を除いて残りが 1 枚以上ある牌とし，向聴数は七対子と国士無双を含む最小値を取る．聴牌の待ちの形は手全体の面子分解から決める．合格は「1 文目の打牌が一致し，誤った文が一つも無く，必須の受け牌が全てどれかの正しい文で挙げられている」局面で，文の順番は問わず，正解の一解に固定しない．必須は，聴牌では待ち牌の全部，非聴牌では打牌後の手に無い有効な数牌のうち手にある 2 枚と両面・カンチャン・ペンチャンを作れる牌とする（手にある牌を引く受け，孤立牌にくっつく受け，七対子の受け，字牌の受けは必須にしない）．必須が 0 の局面は「打 X.」だけで合格とし，natural_dev では 72 局面ある．

判定器の監査は，(a) 向聴数と有効牌の計算が外部ライブラリ（mahjong 1.2.1[30]）と一致すること（向聴は全件，有効牌は 899/900），(b) 全正解文が判定を通ること，(c) 意図的に作った誤文が全て不合格になること，の 3 点で行った．これは誤りが無いことの保証ではない（6.1 節）．

4.3 評価セットと使用履歴

評価セット v2．未使用の対局から 1 対局につき 1 局面を取り，対局の内容のハッシュで 4 分割した．自然セットは対局内の適格局面から一様に 1 つ選ぶ．challenge セットは排他的な 3 層（打牌後聴牌，非聴牌で他家リーチあり，非聴牌・リーチなし・受けの形 4 つ以上）を同数ずつ集めた．いずれも門前の局面に限り，受けの形が無い局面も含める．分割間と既存の全訓練データ（計 99,715 局面）との重なりは 0 であった．評価セットは，モデルの選択や設計に用いる開発用（natural_dev・challenge_dev）と，設計の決定に用いないテスト集合に分けた．本稿ではこのテスト集合をテスト集合 A と呼び，5.4 節で新たに作ったテスト集合をテスト集合 B と呼ぶ．各集合の組成を表 2 に示す．自然分布の巡目は 1〜18 巡目に広がる（開発用 600 のうち 6 巡目以内が 374）．

表 2 評価セット v2 の組成（局面数）

集合	n	聴牌	リーチ	必須牌数 0/1/2/3/4/5/6
natural_dev	600	16	79	72/36/104/137/117/71/63
challenge_dev	300	100	134	17/59/48/24/31/47/74
natural_test	1,500	55	234	164/87/250/321/310/208/160
challenge_test	900	300	384	46/152/196/88/91/106/221

聴牌は打牌後聴牌，リーチは他家リーチありの局面数．必須牌数は評価仕様 v1.0 で数え，6 は 6 以上．上 2 つは開発用，下 2 つはテスト集合 A．

表 3 主結果：テスト集合 A（1,500・900 局面）での単回生成の合格数（学習 seed 42）．下 2 行は採用モデルのみ合格／対照のみ合格の局面数と McNemar 検定の p 値

モデル	natural_test	challenge_test
採用（Qwen3-1.7B）	**1,427**	**827**
対照 Qwen3-1.7B	1,281	697
対照 Qwen2.5-1.5B	1,309	728
対 Qwen3	201 ／ 55（p=1.0e-20）	173 ／ 43（1.1e-19）
対 Qwen2.5	169 ／ 51（5.8e-16）	145 ／ 46（3.9e-13）

テスト集合 A は，事前に固定した計画どおり採用モデルと対照 2 本を一度だけ評価した（5.1 節の主結果）後，計 13 本の探索的な比較に再利用した．初回以外は「テスト集合 A を再利用した探索的評価」と表示する．

4.4 指標

主指標 v1 合格率@1 は，書き直しなしの 1 回の貪欲生成（以下，単回生成）で評価仕様 v1.0 の合格を満たす局面の割合である．部分点は与えない．打牌の宣言だけを出力した場合の合格数（下限）は，natural_dev 600 局面で 72 である．

5. 結果

5.1 事実の列挙

主結果（v1）．表 3 に，テスト集合 A での単回生成の合格数を示す．すべて単一の学習 seed（42）の結果である．採用モデルは，有効牌を色別の soft tokens で，打牌後の向聴と聴牌時の待ちを文字の 1 行で受け取る（表 1）．対照の 2 種は事前登録した前回の設定の Qwen3-1.7B と Qwen2.5-1.5B で，採用モデルとは文字の 1 行（打牌前の値を打牌後の値に修正），打牌後 13 枚と打牌の one-hot の有無，補強データ，受けを書く順が異なる．差はこれらの複合効果で，個々の寄与は分離していない．McNemar 検定は固定した重み同士の対応比較で，訓練 seed のばらつきを含まない．

採用モデルは両集合で対照 2 本をいずれも上回った

© 2026 Information Processing Society of Japan

（表 3）．生成中に判定器で誤りを検出して巻き戻す方式では 1,450・834，単回生成で不合格の局面だけ書き直す方式では 1,470・859 であった（テスト集合 A を再利用した探索的評価）．

参考として，テスト集合 A の部分集合 600 局面で，外部 LLM に v1 の規則を文章と例 5 つで渡して 1 回生成させた．合格数は GPT-6 Astra 600・Claude Fable 598・GPT-5.6 Sol 591・GPT-6 Luna 379（採用モデルの単回生成は 567）であった．プロンプト 1 通りの参考値である．

必須牌数による層別．主結果の出力を，必須の受け牌の種類数（0，1，…，5，6 以上）で層別して再採点した．合格数は natural_test で 164/164・79/87・241/250・305/321・292/310・197/208・149/160，challenge_test で 46/46・136/152・184/196・80/88・81/91・97/106・203/221 であった．必須牌数が 1 以上のどの層でも合格率は 89〜96% で，事実の数が増えても大きく下がらなかった．

5.2 入力の渡し方

同じ訓練レシピで入力の構成だけを変えて比較した（natural_dev 600・各 1 seed）．Mortal の内部表現・打牌後の手牌・有効牌を Projector（soft tokens）で渡す条件 P0 は 578，同じ事実を文字だけで渡す条件 T は 594 で，T が高かった（T だけ合格 22，P0 だけ合格 6，$p = 0.0037$）．有効牌を渡さない文字だけの条件 T0（494）に Mortal の内部表現・文脈・打牌を Projector で加えた T0M（517）は，自然分布で高く（T0M だけ合格 52，T0 だけ合格 29，$p = 0.014$），challenge_dev では向きが逆で差がなかった（合格 259 対 268，$p = 0.16$）．この上乗せは内部表現だけに帰せない（文脈・打牌も含む）．渡した事実は正解の有効牌そのものなので，これは正解をテキストで与えた際の転記性能の比較である（Mortal の評価を含む比較は本節後半）．

Mortal の中間マップ（最後の ResBlock の出力，256 次元 ×34 牌）を牌ごとの 34 本の soft tokens として加えても，別局面のマップに入れ替えた対照との差はなかった（テスト集合 A を再利用した探索的評価・1 seed．natural_test で加えない 1,427 ／マップ 1,442 ／入れ替え 1,447）．色別 4 本の有効牌・手牌トークンを牌ごとのトークンに置き換えても，採用モデルの 1,427 を上回らなかった（無作為初期化 1,367，元のトークンの埋め込みの平均で初期化 1,409）．

拡張課題 v2.1 の診断．v1 の採用モデルから継続学習し，約 9 文の解説を書かせる拡張課題 v2.1 で出力文を種類別に採点した．入力は v1 の構成に計算した事実の入力文を加えた構成で，受けの形と Mortal の評価の文は入力文に含めない．再利用したテスト集合 A natural_test 1,500 の出力について，出力文のうち判定器が正しいとした割合を種類別に数えた．明示的に渡した事実の文は 6,512/6,516（99.9%），手牌から読む受け・待ちの形の文は 4,243/4,579（92.7%），Mortal の評価の文は 1,151/1,557（73.9%）であった．challenge_test では順に 4,722/4,722・2,469/2,621・819/1,197 であった．局面単位の合格は 662/1,500・332/900 である．盤面由来の入力を全て除いた対照は 0/1,500・0/900 であった．

Mortal の評価の文の誤りの原因を開発用 900 局面で調べると，Projector に渡していた確率は合法でない行動まで含めた softmax で，判定器の定義（合法な打牌だけで正規化）と推奨の強さの区分が 642/900 で食い違っていた．そこで同一の初期重みから 2 構成を学習した．P′ は合法手でマスクし，宣言後のリーチ比較も渡す．T′ はさらに手牌・ドラ・有効牌・Mortal の打牌確率（%・小数 1 桁）まで全て文字で渡し，soft tokens を外す．合格は修正前 402，P′ 502，T′ 689（/900），Mortal の評価の文の正答は 74.9%／ 78.8%／ **96.5%** で，P′ だけ合格と T′ だけ合格の局面は 66 対 253（$p = 5.9 \times 10^{-27}$）であった．T′ の評価の文の誤り 37 はすべて区分の閾値の境目（確率差 0.05 未満）にあった．P′ と T′ は batch が異なり（8 と 4），その効果は分けていない．

5.3 反実仮想

Mortal の中間層について，Mortal を追加訓練せずに三つの分析を行った．(a) は中間層にどの程度の情報があるか，(b) はそれを soft tokens として言語モデルに加えたときの効果，(c) は中間層から反実仮想の打牌を予測する MLP プローブの予測を言語化に用いたときの効果を調べる．(b) と (c) は別々の問いで，互いを比べるものではない．

(a) 層別の線形プローブ．対象は他家リーチありの打牌決定 8,405 局面（287 対局）で，目的変数はリーチ宣言を牌譜から除いて Mortal に再推論させると打牌が変わるか（変わる割合 0.460）である．比較基準（巡目・向聴・他家の河・手牌・Q 値など．AUC 0.747）に各層の特徴を加えたときの AUC の差を，対局単位の 5 分割で測った．差は Conv 直後の +0.006 から ResBlock を深く進むほど単調に増え，最後の ResBlock（54 番目，以下 ResBlock 54）で +0.084（5 分割すべて正），最終潜在表現 phi では +0.010 であった．

(b) soft tokens として加えた場合（1 seed・開発用・事前登録）．反実仮想の事実を入力文から取り除き，代わりに ResBlock 54 の出力を牌ごとの 34 本の soft tokens として与えて書かせた．開発用のリーチ局面 213（多数派の答えの割合 60.1%）での二値正答は，何も加えない条件 157，ResBlock 54 を加える条件 151，別局面の ResBlock 54 に入れ替えた対照 159 で，加える条件だけ正答と入れ替

© 2026 Information Processing Society of Japan

表 4　反実仮想の言語化（開発用 2,590 局面で正解と一致した局面の数・seed ごと．最下行は CT だけ正解／ CR だけ正解の局面数）

	seed 42	seed 43	seed 44
CT（文字だけ）	1,488	1,030	1,167
CR（MLP プローブ）	2,026	1,920	1,947
（対応）	158 ／ 696	100 ／ 990	124 ／ 904

えだけ正答の局面は 19 対 27（$p = 0.30$）であった．射影後のベクトルからはこの二値を線形プローブで AUC 0.883 で予測できたが，生成位置の隠れ状態では何も加えない条件と同じ値（0.836 対 0.837）で，情報は生成位置まで伝播しなかったと考えられる．

(c) MLP プローブの予測を用いた場合（3 seed・開発用）． ResBlock 54 の牌ごとの出力から候補の牌ごとにスコアを出す小さな MLP プローブ（256 → 64 → 1）を設け，反実仮想の打牌を予測するよう学習させた．このスコアを，言語モデルの層 12〜27 で手牌の牌名に向かう注意スコアに attention bias として加える．訓練用 10,848 局面の先頭 6,000 で学習し，訓練と別の対局から取った反実仮想用の開発用 2,590 局面（表 2 とは別．打牌が変わる 1,232・同じ 1,358）で，出力した文（打牌が変わるか否かと，その打牌）が正解と一致した局面を数えた（表 4）．入力文だけで書く条件 CT は 3 seed の平均 1,228.3（SD 235.1．下限 1,358 を下回る），attention bias を加える条件 CR は 1,964.3（SD 55.1）で，3 seed とも CR が上回った．常に「同じ牌を切る」と書いた場合の正答数（下限）は 1,358 である．同じ重みのまま attention bias を外すと，どの seed も「変わる」か「同じ」の片方ばかり出力するようになり（1,342・262・163），入力を別の局面の ResBlock 54 に入れ替えると 238・234・211 に低下した．Mortal と同じ構造で訓練していない（無作為初期化の）ネットワークの ResBlock 54 を入力にすると 1,681 だった（seed 42 のみ）．seed 42 では MLP プローブの正答 2,043 と文の一致 2,026 がほぼ同じであり，言語モデルは MLP プローブの予測を言語化していると考えられる．ただし CR は反実仮想の答えで教師付き学習したプローブを含み，訓練していないネットワークでも下限を上回ったので，上乗せの一部はプローブの学習による．(a) の比較基準の特徴で学習したプローブとは比べていない．訓練済みの Mortal の中間層から MLP プローブで予測した反実仮想の打牌を，言語モデルが言語化できることが示された（この課題・1.7B・3 seed・開発用）．

5.4　状況に合わせて要点から話す解説

実験設定． 比較の起点（以下，ベースライン）は，計算済みの事実をテキストの入力文として与え，soft tokens を用いない構成（5.2 節後半の T′ と同じ入力形式，LoRA は末尾 16 層）を，修正前の教師文（GPT-6 Luna のみ，状況 6 種）で学習したモデルである．ベースラインは seed ごとに学習し（3 seed），各 seed のベースラインから継続して，最終の教師文の同じ行・同じ設定で次の 2 条件を学習した．ベースラインとの差には，教師文・LoRA の層数（16 から 28）・学習量の変更がまとめて含まれる．

表 5　状況に応じた解説で誤りを含む局面の数（単回生成，3 seed の平均．計は平均 ± SD．少ないほど良い．テスト集合 B ではベースラインを評価していない）

	開発用				テスト集合 B
	natural	challenge	鳴き	計 /1,200	計 /1,200
ベースライン	124.3	40.7	79.0	244.0±14.5	–
牌トークンなし	4.3	7.3	30.0	41.7±9.2	38.7±5.1
牌トークンあり	5.7	6.3	27.0	39.0±3.6	41.0±4.4

牌トークンあり（本稿のモデル）． 牌 37 種（数牌 27，字牌 7，赤五 3）の名称をそれぞれ 1 トークンとして扱う専用トークンを，既存の語彙とは別に追加した．元の語彙では数牌の名称は 2 トークン，赤五は 3 トークンに分割される．埋め込みと出力層の重みは，元のトークン列の埋め込みの平均で初期化した．学習は 2 段階で行った．第 1 段階では言語モデル本体と LoRA を固定し，追加したトークンの重みのみを教師文の先頭 2,000 行で学習した（ウォームアップ）．第 2 段階では，LoRA（全 28 層，rank 8）と追加トークンの重みを残りの 12,566 行で 1 エポック学習した．同じ文字が牌と座席の両方を表すため，東南西北と中は，座席・局（「東家」「東 3 局」）や語の一部（「途中」）として現れる場合と，入力文の座席の欄では牌トークンに変換しない．

牌トークンなし（対照条件）． ウォームアップを行わず，同一の 12,566 行で LoRA（全 28 層）のみを学習した．したがって両条件の差は，牌トークンの追加とウォームアップ段階の両方を含む．

評価． 開発用データとして natural_dev 600，challenge_dev 300，副露を含む局面 300（以下，鳴き）を用いた．テスト集合 B として，それまでの実験で一度も使用していない天鳳の対局から natural 600，challenge 300，鳴き 300 を新たに構築した（3 集合間で対局は重複しない）．いずれも単回生成を評価した．主指標は，次のいずれかを含む局面の数（少ないほど良い）とした：判定器が偽と判定した文，入力文に存在しない牌名・数値，意図や理由を述べる表現，副露の種類の混同，同一文の反復，途中で途切れた文，主語の不整合，前提を省いた「ツモでしかアガれない」．誤った文は判定器で，それ以外の 7 種は教師文の選別に用いたのと同じ規則のパターン照合で検出した．選別と評価に同じ規則を用いるため，この指標が数えるのはその規則で検出できる誤りに限られる．副指標として，先に言うべき 1 番目の事実が 2 文目までに出た局面の数を，事実ごとの語の一致で近似して数えた．

結果を表 5 に示す．誤りを含む局面の数は，ベースライ

© 2026 Information Processing Society of Japan

ンの平均 244.0 から 39.0〜41.7 に減少した．この減少は牌トークンの有無にかかわらず生じており，教師文と学習設定の変更をまとめた効果である（個々の寄与は分離していない）．牌トークンあり対なしは，seed ごとに 36 対 31，43 対 47，38 対 47 である．平均の差 2.7 は牌トークンなしの seed 間の標準偏差 9.2 より小さく，牌トークンによる正確さの差は検出されなかった．テスト集合 B（未使用の対局から新たに作った 1,200 局面，1 回だけ採点）でも 41.0 対 38.7 で，差は seed 間の標準偏差（4.4・5.1）の範囲内であった．開発用で見られた牌トークンありのばらつきの小ささ（3.6 対 9.2）は，テスト集合 B では再現しなかった．残る誤りの多くは鳴きの集合に集中していた．先に言うべき 1 番目の事実が 2 文目までに出た局面（3 seed の平均．natural ／ challenge ／鳴き）は，ベースライン 423.3 ／ 262.3 ／ 108.0，牌トークンなし 504.3 ／ 282.3 ／ 187.7，牌トークンあり 502.7 ／ 278.3 ／ 188.0 で，両条件はほぼ同じであった．

生成された解説の長さは，文字数では両条件で −3.2〜+2.9% の範囲でほぼ同等であった．一方，牌の名称が 1 トークンで表されるため，1 件の解説の生成に要するステップ数は，3 seed × 3 集合の 9 組すべてで牌トークンありの方が少なかった（開発用で 3.0〜9.3%，テスト集合 B で 1.9〜8.8% 減）．これは主に符号化から生じる差で，正確さの改善を示すものではない．

6. 考察

6.1 自由入力での学習と評価

判定器の限界．本稿の合格率は判定器が受け入れた出力の割合であり，判定器が照合できない文の誤りは数えていない．判定器が検証できなかった文は，誤りとも無害とも断定できない．

自由な文の評価．本稿の判定器は規則で文を照合するので，学習と評価は判定器が読める書き方の範囲に留まる．5.4 節の解説は語順と接続を自由にしたが，文の中身は判定器が照合できる主張に限った．規則を外して自由記述させた予備実験（開発用 30 局面）では，未学習の Qwen3-1.7B と Qwen3.5（4B）で照合できない文が半分を超えた（57%・70%）．多くは入力文の欄の意味を取り違えた理由づけである．自由な文を評価するには，LLM で主張を抽出する処理と，抽出した主張を決定論的に照合する処理を分け，抽出の取りこぼしと誤りを別に測る必要がある．

今後．文字で渡せる事実の範囲を河から一意に決まる事実（現物・筋・壁・手出し・宣言牌）と一局の時系列へ広げること，他家の手牌に関する推測を確からしさ付きの主張として書かせ牌譜の隠れた情報で較正を測ること（聴牌予測には AUC 0.778 の先行例がある [15]）が残る．推測の確からしさを「〜だろう」「〜かもしれない」で言い分けさせる設計は，日本語のモダリティ研究と接する．

6.2 ローカル LLM で同様の学習

外部の大きな LLM は，規則を文章と例で渡すだけで v1 の課題をほぼ満たした（GPT-6 Astra 600/600，採用モデルの単回生成は 567/600．5.1 節）．本稿で 1.7B のモデルを学習する意義は正確さではなく，ローカル実行の低コスト性と，入力形式や内部表現を操作して比較できる点にある．本稿の教師文の作り方（入力文と順番の規則を渡して書かせ，判定器で選別する）は，教師を手元で動く LLM に置き換えても成り立ち，教師の大きさと学習後の正確さの関係を同じ判定器で測れる．ただし上記の予備実験では 4B のモデルが入力の項目を誤読したため，より大きなモデルか多数の候補からの選別を要する．

7. おわりに

ソルバーとルール計算器の出力を写す部分は，定型文の出力に近い水準まで言語モデルに書かせられた（テスト集合 A 1,500 局面で 1,427・1 seed）．合格数は，ルール計算器の事実も Mortal の打牌確率も，テキストで渡す構成の方が soft tokens で渡す構成より高かった（開発用・1 seed．batch の違いは分離していない）．Mortal の中間表現から反実仮想の打牌を予測する MLP プローブの予測を attention bias として加えると，その打牌を言語化できた（3 seed・開発用）．状況に応じた解説への拡張では，牌トークンを導入したモデルで誤りを含む局面を開発用 1,200 局面中平均 39.0，テスト集合 B で 41.0 とした（3 seed）．牌トークンの有無による正確さの差は検出されず，生成ステップ数は開発用で 3.0〜9.3%，テスト集合 B で 1.9〜8.8% 少なかった．

限界．(1) 数値は，5.3 節 (c) と 5.4 節の 3 seed を除き学習 seed 42 の 1 本である．(2) テスト集合 A を 4.3 節のとおり再利用した．初回評価以外は探索的評価である．(3) v1 でも非聴牌の受けの文は手全体を面子に分けておらず，完成した順子の中の 2 枚を受けとして書く文が受けの文の 10.2%（1,374 中 140）にあった（開発用 600 の正解文）．示したのは規則が定める受けの文を偽なく漏れなく書けることであり，盤面の理由を正しく説明できることではない．(4) 外部 LLM との比較はプロンプト 1 通りである．(5) 5.3 節 (b) と入力の渡し方の比較は 1 seed である．(6) 5.4 節の教師文は 4 つの LLM の混成で，新しく追加した状況の教師文は GPT-6 Luna 以外の LLM だけが書いた．教師の違いと状況の違いは分離していない．

参考文献

[1] Li, Lu, Wang, Zheng, Li: Mxplainer: Explain and Learn Insights by Imitating Mahjong Agents, Algorithms, Vol.18, No.12, 738 (2025).

© 2026 Information Processing Society of Japan

[2] Kameko, H., Mori, S. and Tsuruoka, Y.: Learning a Game Commentary Generator with Grounded Move Expressions, IEEE CIG 2015, pp.177–184 (2015).

[3] Lee, A., Wu, D., Dinan, E. and Lewis, M.: Improving Chess Commentaries by Combining Language Models with Symbolic Reasoning Engines, arXiv:2212.08195 (2022).

[4] Min, S. et al.: FActScore: Fine-grained Atomic Evaluation of Factual Precision in Long Form Text Generation, EMNLP 2023, pp.12076–12100 (2023).

[5] Fang, Z., 李吉屹，吉岡真治: 説明可能な麻雀 AI に向けた 2 段階 Dual-LoRA を用いたニューロシンボリック指示学習，言語処理学会第 32 回年次大会（NLP2026）発表論文集，P6-12, p.2828 (2026).

[6] 髙野智椰, 宮本友樹, 片上大輔: 深層学習麻雀 AI の打牌選択への寄与度可視化による対話型牌譜検討支援システムの提案, 第 41 回ファジィシステムシンポジウム（FSS2025）講演論文集，2A3-4，pp.334–338 (2025).

[7] 岡本龍馬，永井秀利，中村貞吾: Kernel SHAP を用いた麻雀の打牌選択モデルにおける判断根拠の説明，2024 年度電気・情報関係学会九州支部連合大会，09-1P-01, p.175 (2024).

[8] Wiseman, S., Shieber, S. M. and Rush, A. M.: Challenges in Data-to-Document Generation, EMNLP 2017 (2017).

[9] Kim, J., Goh, J., Hwang, I., Cho, J. and Ok, J.: Bridging the Gap between Expert and Language Models: Concept-guided Chess Commentary Generation and Evaluation, NAACL 2025 (2025).

[10] Harini S I, Singh, S., Singla, Y. K., Shah, R. R., Doermann, D. and Krishnamurthy, B.: Exploring Collaboration between a Language and a Non-language Agent, EMNLP 2026 (2026).

[11] Pan, A., Chen, L. and Steinhardt, J.: LatentQA: Teaching LLMs to Decode Activations Into Natural Language, ICLR 2026 (2026).

[12] Hebbar, S. A., Sheng, P., Oh, S. and Viswanath, P.: Hallucinations on the Board: Tool-Augmented Evaluation of LLM Chess Commentary, arXiv:2608.04240 (2026).

[13] Gupta, P., Puri, N. et al.: Explain Your Move: Understanding Agent Actions Using Specific and Relevant Feature Attribution, ICLR 2020, arXiv:1912.12191 (2020).

[14] Gyevnár, B., Lucas, C. G., Albrecht, S. V. and Cohen, S. B.: Integrating Counterfactual Simulations with Language Models for Explaining Multi-Agent Behaviour, arXiv:2505.17801 (2025).

[15] Mizukami, N. and Tsuruoka, Y.: Building a Computer Mahjong Player Based on Monte Carlo Simulation and Opponent Models, IEEE CIG 2015 (2015).

[16] 山内悠輔，河原大輔: 手順のテキスト化による将棋解説文生成，言語処理学会第 30 回年次大会発表論文集，P11-5, pp.3197–3202 (2024).

[17] 池田大雅，鶴岡慶雅: Policy-Value Monte-Carlo 木探索を用いた将棋の解説文に出現する手順の予測，言語処理学会第 31 回年次大会発表論文集，P3-23, pp.1175–1179 (2025).

[18] Liu, H., Li, C., Wu, Q. and Lee, Y. J.: Visual Instruction Tuning, NeurIPS 2023 (2023).

[19] Perozzi, B., Fatemi, B., Zelle, D., Tsitsulin, A., Kazemi, M., Al-Rfou, R. and Halcrow, J.: Let Your Graph Do the Talking: Encoding Structured Data for LLMs, arXiv:2402.05862 (2024).

[20] Ghandeharioun, A., Caciularu, A., Pearce, A., Dixon, L. and Geva, M.: Patchscopes: A Unifying Framework for Inspecting Hidden Representations of Language Models, ICML 2024 (2024).

[21] Kim, H., Chen, H., Li, C. and Lee, J. M.: TalkToAgent: A Human-centric Explanation of Reinforcement Learning Agents with Large Language Models, arXiv:2509.04809 (2025).

[22] Kasner, Z. and Dušek, O.: Beyond Traditional Benchmarks: Analyzing Behaviors of Open LLMs on Data-to-Text Generation, ACL 2024, pp.12045–12072 (2024).

[23] Singh, A. et al.: Beyond Human Data: Scaling Self-Training for Problem-Solving with Language Models, TMLR (2024).

[24] Kim, S., Choi, S. and Jeong, M.: Efficient and Effective Vocabulary Expansion Towards Multilingual Large Language Models, arXiv:2402.14714 (2024).

[25] Fujii, K. et al.: Continual Pre-Training for Cross-Lingual LLM Adaptation: Enhancing Japanese Language Capabilities, arXiv:2404.17790 (2024).

[26] Nakash, I., Calderon, N., Ben-David, E., Hoffer, E. and Reichart, R.: AdaptiVocab: Enhancing LLM Efficiency in Focused Domains through Lightweight Vocabulary Adaptation, COLM 2025 (2025).

[27] Equim-chan: Mortal: A Fast and Strong AI for Riichi Mahjong, GitHub, `https://github.com/Equim-chan/Mortal`.

[28] Qwen Team: Qwen3 Technical Report, arXiv:2505.09388 (2025).

[29] Hu, E. J. et al.: LoRA: Low-Rank Adaptation of Large Language Models, ICLR 2022 (2022).

[30] MahjongRepository: mahjong (Python library) 1.2.1, `https://github.com/MahjongRepository/mahjong`.

© 2026 Information Processing Society of Japan