自己検証 LLM による日本司法試験短答式試験合格

Shin Andrew
慶應義塾大学 理工学部 情報工学科
shin@ics.keio.ac.jp

概要

日本の司法試験は、その複雑な回答形式ゆえに LLM にとって依然として難関である。問題を単純化する既存研究とは異なり、本稿では実際の試験形式を忠実に再現したデータセットと自己検証 (self-verification) モデルを提案する。本モデルは、問題構造や採点規則を変更することなく公式合格点を上回ることに成功した初の事例である。本結果は、形式に忠実な学習と一貫性検証の重要性を裏付けるものである。[1)2)]

1 はじめに

大規模言語モデル（LLM）は、質問応答 [1, 2] や要約 [3] に加え、数学 [4, 5] やプログラミング [6, 7] といったドメイン特化型の推論においても驚くべき能力を示している。しかしながら、高度な専門性と構造化された形式を持つ法律分野、特に日本の司法試験においては、依然として性能にばらつきが見られる。法的推論には、単なる言語能力を超えて、法令の正確な解釈や相互作用する複数の条件の慎重な評価が求められるからである。中でも日本の司法試験（短答式）は、複数の命題を結合的に評価し、かつ厳格な回答形式を遵守しなければならないという点で、極めて困難なベンチマークである。構成要素のわずかな誤りが回答全体を無効にするこの評価体制下では、ベースモデルの性能は限定的であり、一般的な言語理解能力と試験レベルの法的能力との間には大きな隔たりが存在する。

近年の研究では、この課題に対してデータセットの構築やタスクの再定式化によるアプローチが試みられている。代表的な JBE-QA [8] は、複雑な問題を独立した正誤判定（True-False）に分解することで学習を単純化し、性能向上を報告している。しかし、こうした分解アプローチは本来の試験構造を根本的に変更するものであり、実際の試験形式や厳密な採点基準の下で、モデルが真に通用するかどうかという疑問は未解決のままである。

そこで本稿では、タスクをモデルに合わせて変更するのではなく、実際の試験形式と評価尺度を忠実に再現したデータセットを構築し、それを用いて訓練された自己検証（self-verification）モデルを提案する。我々のモデルは、問題構造や採点規則を変更することなく、2024 年の実際の司法試験において公式の合格点（93 点）を上回る 96 点を獲得した。これは我々の知る限り、LLM が本来の試験形式で合格を達成した初の実証である。本結果は、形式に忠実な教師あり学習と一貫性検証のアライメントが重要であることを強調しており、慎重に設計された単一モデルによるアプローチが、ハイステークスな専門的推論タスクにおいて、より複雑なシステムを凌駕し得ることを示唆している。

2 関連研究

日本の司法試験は、法的推論の難関ベンチマークとして近年注目されている。COLIEE [9] 等の初期の研究は情報検索や含意認識等のサブタスクに焦点を当てていたが、近年の研究 [10] では試験独自の構造を扱う重要性が認識されつつある。

中でも JBE-QA データセット [8] は、過去問を独立した正誤判定に分解することで学習効率を向上させ、高い性能を報告している。しかし、この分解アプローチは、複数の命題を結合して評価するという本来の試験構造を捨象しており、厳格な採点ルール下での通用性は保証されない。本研究は、実際の試験形式と尺度を用いてモデルを評価することで、このギャップに直接対処するものである。

1） https://huggingface.co/datasets/shinysup/JBE-MC-original-format
2） https://github.com/shinandrew/self_verification

表 1　JBE-QA[8] データセットとの比較.

Dataset	Question	Answer	#Questions
JBE-QA	憲法第３１条の定める法定手続の保障は、直接には刑事手続に関するものであるが、行政手続にも及ぶと解すべき場合があり、その場合には行政処分の相手方に常に事前の告知、弁解、防御の機会を与える必要がある。	False	2,770
	憲法第３５条は、住居、書類及び所持品について、侵入、捜索及び押収を受けることのない権利を規定しているが、この規定の保障対象には、住居、書類及び所持品に準ずる私的領域に侵入されることのない権利が含まれる。	True	
	憲法第３８条第１項は、自己が刑事上の責任を問われるおそれのある事項について供述を強要されないことを保障するものであり、氏名の供述も、これによって自己が刑事上の責任を問われるおそれがあることから、原則として保障が及ぶ。	False	
Ours (実際の試験形式と一致)	刑事手続上の権利に関する次のアからウまでの各記述について、最高裁判所の判例の趣旨に照らして、それぞれ正しい場合には１を、誤っている場合には２を選びなさい。 ア．憲法第３１条の定める法定手続の保障は、直接には刑事手続に関するものであるが、行政手続にも及ぶと解すべき場合があり、その場合には行政処分の相手方に常に事前の告知、弁解、防御の機会を与える必要がある。 イ．憲法第３５条は、住居、書類及び所持品について、侵入、捜索及び押収を受けることのない権利を規定しているが、この規定の保障対象には、住居、書類及び所持品に準ずる私的領域に侵入されることのない権利が含まれる。 ウ．憲法第３８条第１項は、自己が刑事上の責任を問われるおそれのある事項について供述を強要されないことを保障するものであり、氏名の供述も、これによって自己が刑事上の責任を問われるおそれがあることから、原則として保障が及ぶ。	2,1,2	460

3　手法

3.1　データセット構築

日本の法務省より過去 6 年間（2019 年〜2024 年）の実際の試験問題を収集し、2024 年（令和 6 年/R6）をテストセットとして分離した。日本の司法試験の形式と評価基準を忠実に再現したデータセットを構築した。問題を独立した正誤判定文に分解する先行研究とは異なり、本データセットの各インスタンスは、すべての構成要素となる記述と元の選択肢を含む完全な試験問題に対応している。回答は、各記述の正誤を示す数字の連結など、試験で要求される通りの形式で表現される。

各問題には、科目カテゴリ（憲法、民法、刑法）、実施年度、および配点のアノテーションが付与されている。これにより、単なる正解率だけでなく、試験で使用される公式の配点方式に基づいた評価が可能となる。データセットは年度ごとに分割し、過去の年度（2019〜2023 年/R1〜R5）を訓練用、2024 年（R6）を評価用とした。受験者が過去問を参考に学習するという実際の試験対策を反映している。

各記述を独立した True/False クエリに還元する分解形式とは異なり、実際の試験における正解はすべての記述の結合評価に依存する。例えば、記述の真偽値の特定の組み合わせに対応する単一の選択肢を選ぶ必要があったり、複数の記述に対応させて「112」のような数字の連結を出力する必要があったりする。意味的な判断または形式のいずれかに違反した場合、回答は不正解とみなされる。表 1 に、本データセットと JBE-QA の対比を示す。

3.2　自己検証 (Self-Verification)

ファインチューニング: 我々のアプローチは、教師ありファインチューニングと回答条件付き自己検証を組み合わせたものである。訓練中、モデルは問題をより単純な部分問題に分解することなく、完全な問題を与えられた状態で正しい試験形式の回答を生成するようにファインチューニングされる。一連の記述 $\{s_{i1}, s_{i2}, \ldots, s_{in}\}$ と司法試験で定義された有効な回答形式の集合からなる問題 q_i が与えられたとき、モデルは意味的な正しさと厳密な形式制約の両方を満たす単一の回答 a_i を生成することが求められる。ここで a_i は、実際の試験と同様に複数の整数を含む場合がある。

自己検証: 推論時において、モデルが自身の予測回答を元の問題の文脈で再評価する検証ステップを導入する。重要な点は、この検証が同一のファインチューニング済みモデルによって行われるが、検証指向の振る舞いを誘発する異なるプロンプトの下で実行されることである。

形式的には、$f_\theta(q)$ を問題 q に対するモデルの初期予測とする。次に、問題と初期予測回答との整合性を評価することで修正回答を生成する検証関数 $g_\theta(q, f_\theta(q))$ を定義する。最終的な出力は $\hat{a} = g_\theta(q, f_\theta(q))$ で与えられる。f_θ と g_θ は同一のパラメータを共有するが、一方は回答生成を促し、他方は保守的な修正を促すという異なるプロンプトでインスタンス化される。この手順は推論時に 1 回の追加フォワードパスを要するのみであるが、形式上の誤りや局所的な推論ミスに対するロバスト性を大幅に向上させる。図 1 に本アプローチの全体的なワークフローを示す。

プロンプト設計: 表 2 に本システムで使用したプロンプトをまとめる。回答形式の指示プロンプトは、要求される厳密な形式を強制するように調整されており、これにより司法試験の問題において頻出する多様な出力形式に対応するための正規化処理が不要となる。検証用プロンプトは、明らかな矛盾が検出されない限り元の回答を維持するようモデルに

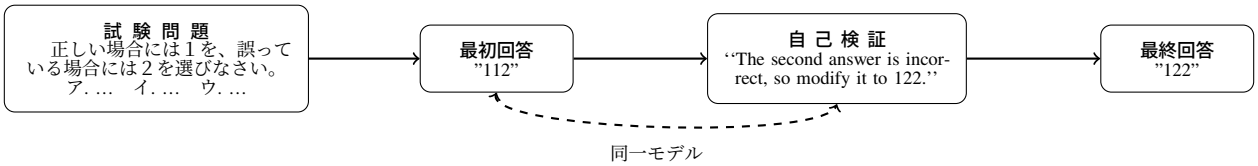

図 1　共有モデルを用いた自己検証手法の概要

表 2　システム役割・回答形式・自己検証に対するプロンプト

Purpose	プロンプト
システム役割	あなたは日本の司法試験を受験する受験者である。
回答形式	【回答形式の厳守】必ず「答えのみ」を出力せよ。理由・説明・記号は一切不要。 1) 選択肢が番号で与えられている場合（例：1. アOイOウO、2. アOイOウX…）→ 正しい選択肢の番号のみ出力（例：2） 2) 各記述（ア・イ・ウ…）について 1 / 2 を答える問題の場合→ 数字列のみ出力（例：112） 禁止：　- OOX　- アOイOウX　- ア1イ1ウ2　- 説明文
自己検証	あなたは法律試験の答案を最終確認する役割である。以下の【問題】と【あなたの解答】を照らし合わせ、選択肢番号または数値の形式として最も正しい最終解答を一つだけ出力せよ。 ・問題文の条件に照らして明らかに誤っている場合のみ修正すること ・元の解答が正しい場合は、そのまま同じ解答を出力すること ・理由や説明は一切出力せず、最終的な数字のみを出力せよ

明示的に指示しており、これは不要な修正を防ぐ上で重要であることがわかった。

4　実験

4.1　実験設定

令和 6 年（R6）の司法試験問題を用いて評価を行い、それ以前の年度（R1〜R5）の問題のみをファインチューニングに使用した。ベースモデルとして GPT-4.1 [11] を使用し、ゼロショットおよびフューショット設定の両方を検証した。フューショット設定では、訓練セットから 5 つのサンプルデモンストレーションを選択した。また、我々のデータセットと JBE-QA のそれぞれを用いて、個別の GPT-4.1 モデルをファインチューニングした。自己検証の効果を検証するため、自己検証の有無による結果を報告する。すべてのモデルに対して同一のプロンプトを使用し、実験は 3 回繰り返された。

評価指標として、完全一致（Exact-match）正解率に加え、試験の採点ルールに従って部分点を与える公式の試験得点を報告する。部分点の仕組みは以下の通りである。3 つ以上の設問がセットで n 点とされている場合、1 つの設問を間違えると $n-2$ 点となるが、2 つ以上の設問を間違えると 0 点となる。例えば、5 つの設問がセットで 4 点の場合、4 つの設問に正解すれば 2 点が得られるが、正解が 3 つの場合は、正解率が 50%を超えているにもかかわらず 0 点となる。試験は憲法 50 点、民法 75 点、刑法 50 点の計 175 点満点で構成される。2024 年（令和 6 年/R6）の試験の場合、実際の合格点は 93 点であった。また、各科目で少なくとも 40%の得点を達成しなければならないという追加要件もある。

4.2　結果と分析

表 3 に、検証した各モデルの結果をまとめる。

ベースモデル: ゼロショット設定での性能は低く、事前学習された法的知識だけでは試験レベルのタスクには不十分であることが明確に示された。フューショット設定も、ゼロショットに比べて性能をほとんど向上させなかった。これは、少数のサンプル提示では法的知識の補完や試験特有の厳格な形式への誘導が十分になされないためと考えられる。一方で、自己検証を行うことで明確な性能向上が見られた。この効果はモデルの選択にかかわらず一貫しており、自己検証が効率的かつモデル非依存の手法であることが実証された。

JBE-QA: 大規模なデータセットで訓練されているにもかかわらず、JBE-QA モデルは実際の司法試験において著しく低い性能を示した。これは、単に知識を増やすだけでは、より複雑なタスクでの性能向上には直結しないことを示唆している。彼らの分解戦略は、学習を単純化するために記述間の結合制約を取り除いてしまう。二値分類への再定式化は、組み合わせルールに基づく制約付き選択という本来のタスクからの乖離（分布シフト）を引き起こし、断片化された知識表現を助長する。その結果、推論時にそれらを再構成して全体として矛盾のない回答を導くことが困難となる。これらの知見は、ハイステークスな試験においては、訓練中に本来の問題形式を維持し、局所的な知識と大局的な決定の一貫性を整合させることが不可欠であることを示している。なお、本モデルにおいても自己検証による有意な性能向上が確認された。

提案手法（Ours）: 我々のデータセットによる

表 3　司法試験（令和 6 年）における性能比較。Accuracy は完全一致の正解率を示し、Points は部分点を含む公式の採点方式に従う。また、憲法、民法、刑法の科目別平均スコアを示す。

Model	Accuracy	Points (Avg/Min/Max)	憲法	民法	刑法
Passing Score for Examinees	N/A	93 (out of 175)	20	30	20
Base (Zero-Shot)	0.4036	67.0 / 65 / 68	8.0	32.0	27.0
Base (Few-Shot)	0.3896	68.3 / 63 / 71	8.0	33.3	27.0
Base (Few-Shot) + Self-Verification	0.4156	76.3 / 76 / 77	9.7	36.7	**30.0**
Fine-Tuned w/ JBE-QA	0.3766	64.0 / 62 / 66	8.0	30.0	26.0
Fine-Tuned w/ JBE-QA + Self-Verification	0.4226	80.7 / 78 / 82	21.0	32.7	27.0
Fine-Tuned w/ Ours	0.4675	92.3 / 91 / 93	20.3	42.0	**30.0**
Fine-Tuned w/ Ours + Self-Verification	**0.4935**	**94.7 / 94 / 96**	**22.3**	**42.3**	**30.0**

表 4　実際の司法試験形式における定性的な評価例。モデルの「+V」は自己検証の実行を示す。各出力の括弧内は獲得点数を表す。太字は正解および満点を示す。

Question	次のアからオまでの各記述を判例の立場に従って検討し、正しい場合には１を、誤っている場合には２を選びなさい。 ア．甲は、宝くじの当せん金を得るため、外れた宝くじに印字された番号を当せん番号に改ざんした。 この場合、甲に有印私文書変造罪が成立する。 イ．甲は、事情を知らない乙に対し、偽造通貨を真正な通貨のように装って代金として交付し、乙から商品を購入した。この場合、甲に詐欺罪及び偽造通貨行使罪が成立し、両罪は観念的競合となる。 ウ．甲は、乙から、乙の代わりにＡ大学の入学試験を受けてほしいと頼まれ、これを引き受け、乙に成り済まして同入学試験を受け、氏名欄に乙の氏名を記載し、乙名義で答案を作成した。この場合、甲に有印私文書偽造罪が成立する。 エ．甲は、行使の目的で、他人が振り出した額面１００万円の小切手の金額欄に「０」を加え、額面１０００万円の小切手に改ざんした。この場合、甲に有価証券偽造罪が成立する。 オ．甲は、乙から金銭の借入れとして１万円札１０枚を受け取った際、それらの中に偽造の１万円札が含まれていることに気付かず、その後、偽造の１万円札の存在に気付いたが、行使の目的でそのまま保持した。この場合、甲に偽造通貨収得罪は成立しない。

Model	Exam(GT)	Base(ZS)	Base(FS)	Base+V	JBE-QA	JBE-QA+V	Ours	Ours+V	MA(Same)	MA(Sep)
Output (Pts)	**22121(4)**	21222(0)	21222(0)	21122(0)	21212(0)	21211(0)	21121(2)	**22121(4)**	21222(0)	21212(0)

ファインチューニングは、自己検証の有無にかかわらず、他のアプローチを明らかに上回っている。特筆すべきは、自己検証なしでも合格点付近のスコアを獲得している点である。回答の組み合わせ空間（例：「11221」など）の広大さを考慮すると、これは形式の暗記や当て推量によるものではない。むしろ、訓練中に本来の複数命題形式に触れることで、各構成記述を内部的に推論し、結合的に評価する能力が誘導されたためと考えられる。

さらに推論時に自己検証を導入することで、一貫した性能向上が観察された。検証ステップにより、モデルは元の問題に対する自身の予測の内部整合性を再評価することが可能になる。これは、正しい複合回答の中に含まれる単一の誤判断といった局所的な不整合を修正するのに有効であり、公式の採点方式において直接的なスコア向上をもたらす。重要な点は、自己検証が外部データに頼らず、第 2 パスの推論フェーズにおいてモデルの既存の法的知識を活用している点である。

これらの結果は、形式特化のファインチューニングと自己検証の両方が、モデル内に既に存在する潜在的な知識を引き出す「触媒」として機能していることを示唆している。これらは、比較的小規模なデータセットであるにもかかわらず大幅な性能向上を実現した要因としても説明がつく。すなわち、形式学習が知識の活用法をモデルに教示し、自己検証が大局的な一貫性を促進することで、この引き出しプロセスをさらに強化しているのである。

表 4 に定性的な例を示す。我々のモデルは、他のモデルが苦戦する単一回答形式と複合回答形式の双方に正しく対応している。また、正解率が 50%を超えていても 0 点となるケースが多い事実は、分解された命題での好成績が実際の試験形式での性能を保証するものではなく、真の成功には複合的な推論が不可欠であることを裏付けている。

5　おわりに

本稿では、本来の問題形式と採点基準の下で日本の司法試験に合格した初の LLM システムを提示し、データセット構築および学習において試験の複数命題構造を維持することが不可欠であることを実証した。問題を分解する既存アプローチが現実的な条件下で十分な性能を発揮できないのに対し、軽量な自己検証を備えた本モデルは、小規模なデータセットであっても、外部リソースや問題の分解に頼ることなく合格ラインを達成した。この結果は、本試験における成功が推論の分散ではなく、密結合した命題間での大局的な一貫性の維持に依存していることを示唆しており、形式特化の学習と自己検証が、モデル内の潜在的な知識を効果的に引き出す触媒として機能していると結論付けられる。

謝辞

本研究は 2025 年度栢森情報科学振興財団（https://www.kayamorif.or.jp/）の研究助成を受けたものです.

参考文献

[1] Murong Yue. A survey of large language model agents for question answering. ArXiv, Vol. abs/2503.19213, , 2025.

[2] Jens Lehmann, Antonello Meloni, Enrico Motta, Francesco Osborne, Diego Reforgiato Recupero, Angelo Salatino, Sahar Vahdati, TU ScaDS.AI, Dresden, and De. Large language models for scientific question answering: An extensive analysis of the sciqa benchmark. In Extended Semantic Web Conference, 2024.

[3] Yixin Liu, Alexander R. Fabbri, Pengfei Liu, Dragomir R. Radev, and Arman Cohan. On learning to summarize with large language models as references. In North American Chapter of the Association for Computational Linguistics, 2023.

[4] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Yu Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. ArXiv, Vol. abs/2402.03300, , 2024.

[5] An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. ArXiv, Vol. abs/2409.12122, , 2024.

[6] Juyong Jiang, Fan Wang, Jiasi Shen, Sungju Kim, and Sunghun Kim. A survey on large language models for code generation. ACM Transactions on Software Engineering and Methodology, 2024.

[7] Ahmed El-Kishky, Alexander Wei, Andre Saraiva, Borys Minaev, Daniel Selsam, David Dohan, Francis Song, Hunter Lightman, Ignasi Clavera, Jakub W. Pachocki, Jerry Tworek, Lorenz Kuhn, Lukasz Kaiser, Mark Chen, Max Schwarzer, Mostafa Rohaninejad, Nat McAleese, o3 contributors, Oleg Murk, Rhythm Garg, Rui Shu, Szymon Sidor, Vineet Kosaraju, and Wenda Zhou. Competitive programming with large reasoning models. ArXiv, Vol. abs/2502.06807, , 2025.

[8] Zhihan Cao, Fumihito Nishino, Hiroaki Yamada, Nguyen Ha Thanh, Yusuke Miyao, and Ken Satoh. Jbe-qa: Japanese bar exam qa dataset for assessing legal domain knowledge. 2025.

[9] Nguyen Ha Thanh, Hai-Yen Thi Vuong, Phuong Minh Nguyen, Binh Dang, Quan Minh Bui, Sinh Trong Vu, Chau Nguyen, Vu Tran, Ken Satoh, and Minh Le Nguyen. Jnlp team: Deep learning for legal processing in coliee 2020. ArXiv, Vol. abs/2011.08071, , 2020.

[10] Hoang-Trung Nguyen, Tan-Minh Nguyen, Xuan-Bach Le, Tuan-Kiet Le, Khanh-Huyen Nguyen, Ha Thanh Nguyen, Thi-Hai-Yen Vuong, and Le-Minh Nguyen. Nowj@coliee 2025: A multi-stage framework integrating embedding models and large language models for legal retrieval and entailment. ArXiv, Vol. abs/2509.08025, , 2025.

[11] OpenAI. Introducing gpt-4.1 in the api. https://openai.com/index/gpt-4-1/, 2025.

[12] Qizheng Zhang, Changran Hu, Shubhangi Upasani, Boyuan Ma, Fenglu Hong, Vamsidhar Reddy Kamanuru, Jay Rainton, Chen Wu, Mengmeng Ji, Hanchen Li, Urmish Thakker, James Zou, and Kunle Olukotun. Agentic context engineering: Evolving contexts for self-improving language models. ArXiv, Vol. abs/2510.04618, , 2025.

[13] Li Zhang and Kevin Ashley. Mitigating manipulation and enhancing persuasion: A reflective multi-agent approach for legal argument generation. ArXiv, Vol. abs/2506.02992, , 2025.

[14] Jingyun Sun, Chengxiao Dai, Zhongze Luo, Yangbo Chang, and Yang Li. Lawluo: A multi-agent collaborative framework for multi-round chinese legal consultation. 2024.

表 5　令和 6 年短答式試験におけるマルチエージェント設定の性能。設定は表 3 と同様。

Model	Accuracy	Points (Avg/Min/Max)	憲法	民法	刑法
マルチエージェント（同一モデル）	0.4026	75.7 / 74 / 79	19.3	30.7	25.7
マルチエージェント (別々のモデル　)	0.3969	71.0 / 66 / 77	12.7	34.7	25.6

A　マルチエージェント設定の検証

マルチエージェント推論は多くの複雑なタスクにおいて成功を収めており [12]、法領域においても試みられている [13, 14]。相互作用するエージェントへの明示的な分解が日本の司法試験における性能を向上させるかを評価するため、我々は検索、検証、知識抽出、および最終回答を担う異なるエージェントから構成されるマルチエージェントパイプラインを実装した。均質なエージェントや非形式的な連携を仮定する先行研究とは異なり、我々の実装は明確に分離された機能的役割を割り当て、共有モデルと個別にファインチューニングされたエージェント構成の双方を、実際の試験採点方式の下で評価する。

表 6　それぞれのエージェントへのプロンプト。

エージェント	プロンプト
Retriever	以下の問題に関連すると考えられる過去問とその回答を選択せよ。選択の基準は、扱われている法分野、論点、条文、または判例の種類が共通しているかどうかである。最大で数問まで選んでよい。
Verifier	以下の問題に対して参考になる過去問と回答のみを選別してください。
Extractor	以下の問題と正解から、将来の類似問題に使える一般化可能な法的知識を抽出せよ。
Reasoner	以下は関連する法的知識である。上記を踏まえて、次の問題に答えよ。

A.1　マルチエージェントアーキテクチャ

パイプラインは 4 つの直列的なエージェントからなり、[12] によって提案されたアーキテクチャに緩やかに着想を得ている。

- 検索エージェント *(Retriever Agent):* テスト問題 q が与えられたとき、検索エージェントは訓練セット（R1–R5）から、q に関連すると判断される過去問と回答の候補セットを選択する。
- 検証エージェント *(Verifier Agent):* 検証エージェントは、テスト問題と検索された過去問候補を受け取り、関連性があるとみなされるもののみを残すようにフィルタリングを行う。その役割は、表面的には類似しているが法的に無関係な問題を破棄し、それによって知識抽出の前のノイズを低減することである。
- 知識抽出エージェント *(Knowledge Extraction Agent):* 検証された各過去問に対して、抽出エージェントは問題と回答のペアから一般化可能な法的原理を抽象化する。エージェントは、再利用可能な基準、条件、またはパターンのみを箇条書き形式で出力するように指示される。
- 最終推論エージェント *(Final Reasoning Agent):* 推論エージェントは、元のテスト問題と集約された抽出知識を受け取り、試験で要求される厳格な形式で最終回答を生成する。このエージェントは、すべての命題の結合評価および形式制約の遵守に対して単独で責任を負う。

我々はこのアーキテクチャについて 2 つの構成を評価した。「共有モデル（shared-model）」設定では、すべてのエージェントが前の実験と同様に我々のデータセットでファインチューニングされた同一モデルからインスタンス化され、役割分担と相互作用の効果を分離して検証した。「個別ファインチューニング（independently fine-tuned）」設定では、機能的な専門性と多様性を高めることを目的として、各エージェントが同じ訓練データを用いて役割固有のプロンプトで個別にファインチューニングされた。各エージェントのプロンプトは 表 6 に示す通りである。なお、回答形式の指示は前の実験と同一である。

A.2　結果と分析

表 5 に示すように、いずれの構成も単一のファインチューニング済みモデルと比較して大幅に低い性能となった。共有モデルによるマルチエージェントシステムは平均 75.7 点であり、個別ファインチューニングによるマルチエージェントシステムはさらに性能を落とし 71.0 点となった。これらはいずれも合格点を大幅に下回っている。推論の複雑さが著しく増したにもかかわらず、これらの結果は単一モデルのアプローチを大きく下回るものであった。

我々の結果は、マルチエージェントシステムが強力な単一モデルのベースラインを自動的に上回るわけではないことを強調している。それどころか、日本の司法試験のような制約の厳しいタスクでは、個々のエージェントによる誤りがパイプライン全体に伝播し複合する傾向があるため、推論をエージェント間で分散させることは有害になり得る。特に、抽象化と検証の段階で微細な不整合が生じる可能性があり、最終推論エージェントは厳格な形式制約と結合的一貫性の制約の下でそれらを整合させなければならないという課題が生じる。

さらに、独立したファインチューニングを通じてエージェントの多様性を高めても性能は向上せず、むしろ調整の失敗（coordination failures）を悪化させることがわかった。独立して訓練されたエージェントは表面的なバリエーションは大きいものの、出力を確実に統合するための共有された表現空間を欠いている。対照的に、共有された表現は、特に複数の相互依存する命題間で大局的な一貫性を維持することが成功の鍵となる設定において、効果的なエージェントの振る舞いにとって重要であるようである。