卒業論文

# LLMマルチエージェント弁証法的推論による質問応答

指導教官　　村上 陽平 教授

立命館大学情報理工学部
先端社会デザインコース

増尾 柚希

2026年1月1日

# LLM マルチエージェント弁証法的推論による質問応答

増尾 柚希

## 内容梗概

大規模言語モデル（LLM）を用いた質問応答システムは，さまざまなアプリケーション，システムで実用化が進んでいる．そして近年では，LLM に複数の価値観や立場を付与したマルチエージェント対話が提案され，模擬的な議論や意思決定を生成する技術が登場している．多文化共生社会においては, 質問応答は単なる情報検索にとどまらず, 異なる文化的・宗教的・倫理的背景をもつ人々の相互理解を促進する役割が求められている．しかしながら，既存の LLM ベースのマルチエージェント対話は，対立する立場をもとに勝敗を決めるディベート的議論が多く，最終的な結論も表層的な折衷案の提示に留まる．たとえば,「学校清掃を生徒が行うべきか職員が行うべきか」といったトピックにおいて，現状の LLM は「日常清掃は生徒，専門清掃は職員」といった妥協案を返す傾向があるが，これは価値観の本質的な対立を踏まえ，清掃を教育活動として再定義した生徒と職員の共同清掃のような止揚的な解決の回答生成は依然として実現できていない．そこで，本研究では，多様な価値観の対立を扱うトピックを対象に，LLM を用いたマルチエージェント対話によって新たな価値・制度・概念の再構成（止揚）を導く弁証法的な対話プロトコルを提案する．具体的には，kido，kurihara[1] らの記号論理に基づく既存の弁証法議論，および統合のアルゴリズムの手続きを参照し，その形式的プロセスを LLM マルチエージェントが自然言語で模倣できるように再構成した段階的対話モデルを設計する．本手法の実現にあたり，取り組むべき課題は以下の 2 点である．

議論の収束

質問応答のシステムに適用される推論のプロトコルは，その推論過程の終了を保証できなければならない．こと弁証法的対話プロトコルにおいては，エージェント同士の議論を収束させることで，推論過程の終了を保証する．記号論理を用いて表現されたエージェントの知識ベースや，それらを用いた論理は手動で定義されるため，有限であり，同じ論理の再使用を許可しない限り，それらを用いて行う議論は必ず収束する．しかし，自然言語を用いて議論を行う LLM のエージェント同士の議論では，意味的に同義である論理の再使用を制限する必要があり，その制御はプロンプトによる指示だけではそれを保証できない．

高次な止揚論証の生成

統合（止揚）は折衷ではなく価値の再構成による高次な概念の創出を意味するが，その成立条件及び構築アルゴリズムは自然言語では明確化されておらず，形式的定義だけでは意味的統合として妥当かを判断できない．また，kido, kurihara[1] の統合アルゴリズムでは，入力としてお互いに defeated された論証のみを入力としていたが，LLM では議論の収束が難しく，議論に決着をつけることがほとんどできないため，統合アルゴリズムをそのまま適用することは実用的に困難である．

本研究では，以上の課題を解決するために，プロトコル内で反復的にエージェントの議論が行われる「反論」フェーズにおいて，同一内容の主張・反論の再生成をプロンプトによって明示的に制限するとともに，反論回数の上限を設けた．上限に達した議論はその時点で終了し，それまでに生成された Argument 群から，対立している評価軸をエージェント自身に抽出させる．これにより，お互いの議論に決着がついていない場合であっても，対立構造を明示化した上で統合処理に移行することが可能となる．止揚論証の構築においては，Kido, Kurihara [1] による統合アルゴリズムの形式的手続きを参照し，それを自然言語プロンプトとして再構成した．具体的には，対立する主張の前提（support）を背景知識のもとで拡張する処理，統合可能な評価軸に基づいて論証を特化する処理（Specialization），および両者を包摂する上位概念を導出する汎化処理（Generalization）を段階的に実行させることで，単なる折衷ではなく，価値観の前提そのものを再構成した止揚論証を生成できることを確認した．加えて本研究では，提案プロトコル組み込んだ弁証法的推論による質問応答システムを実装し，kido, kurihara[1] らの止揚論証の例との比較を行った．

本研究の貢献は以下の通りである．

議論の収束

実装したシステムを用いて 10 個のトピックについて議論を行わせ，対話プロトコルは必ず収束することを確認した．

高次な止揚論証の生成

実装した質問応答システムを用いて kido, kurihara[1] らの対話例を実際に行い，最終的な止揚論証が意味的に一致することを確認した．

# LLM Multi-agent Dialectical Reasoning for Question Answering

Yuzuki MASUO

**Abstract**

Question answering systems using large language models (LLMs) are being increasingly deployed across various applications and systems. In recent years, multi-agent dialogue approaches that assign multiple values and perspectives to LLMs have been proposed, enabling the generation of simulated debates and decision-making processes. In multicultural societies, question answering is expected not only to serve as information retrieval but also to facilitate mutual understanding among people with diverse cultural, religious, and ethical backgrounds by generating consensus-based responses.

However, existing LLM-based multi-agent dialogues often rely on debate-style arguments that determine winners and losers based on opposing positions, with final conclusions limited to superficial compromises. For instance, when addressing topics such as "Should students or staff be responsible for school cleaning?", current LLMs tend to return compromise solutions like "students for daily cleaning, staff for specialized cleaning." Such responses fail to achieve dialectical synthesis that, by addressing the fundamental conflict of values, redefines cleaning as an educational activity and proposes collaborative cleaning by both students and staff.

Therefore, this study proposes a dialectical dialogue protocol that enables LLM-based multi-agent dialogue to guide the reconstruction (synthesis) of new values, systems, and concepts on topics involving diverse value conflicts. Specifically, we refer to the existing dialectical argumentation and integration algorithms based on symbolic logic by Kido and Kurihara [1], and redesign the formal process into a stepwise dialogue model that LLM multi-agents can replicate using natural language.

To realize this approach, two key challenges must be addressed:

**Convergence of Argumentation:** The reasoning protocol applied to question answering systems must guarantee the termination of the reasoning process. In dialectical dialogue protocols, this is ensured by converging the

arguments between agents. When using symbolic logic, knowledge bases and logical rules are manually defined and thus finite; as long as reuse of the same logic is not permitted, arguments using them will necessarily converge. However, in LLM-based agent dialogues using natural language, it is necessary to restrict the reuse of semantically equivalent logic, and this control cannot be guaranteed by prompt instructions alone.

**Generation of Higher-Order Synthetic Arguments:** Synthesis (Aufhebung) refers not to compromise but to the creation of higher-order concepts through value reconstruction. However, the conditions for its establishment and construction algorithms are not clearly defined in natural language, and formal definitions alone cannot determine semantic validity as integration. Furthermore, Kido and Kurihara's [1] integration algorithm accepts only mutually defeated arguments as input. Since LLMs struggle to achieve argumentative convergence and rarely reach definitive conclusions, directly applying the integration algorithm is practically difficult.

To address these challenges, this study explicitly restricts the regeneration of identical claims and rebuttals via prompts in the "rebuttal" phase, where agents engage in iterative argumentation within the protocol, and imposes an upper limit on the number of rebuttals. When this limit is reached, the dialogue terminates, and agents are prompted to extract the conflicting evaluation axes from the generated argument set. This enables transition to integration processing even when the arguments have not reached definitive conclusions, by explicitly articulating the conflict structure.

For constructing synthetic arguments, we refer to the formal procedures of Kido and Kurihara's [1] integration algorithm and reformulate them as natural language prompts. Specifically, we confirmed that synthetic arguments reconstructing the premises of values themselves, rather than mere compromises, can be generated by sequentially executing processes such as expanding the premises (support) of conflicting claims based on background knowledge, specializing arguments based on integrable evaluation axes (Specialization), and deriving superordinate concepts that subsume both (Generalization).

Additionally, we implemented a question answering system incorporating

the proposed protocol and compared the results with the synthetic argument examples by Kido and Kurihara [1].

The contributions of this study are as follows:

**Convergence of Argumentation:** Using the implemented system, we conducted dialogues on 10 topics and confirmed that the dialogue protocol always converges.

**Generation of Higher-Order Synthetic Arguments:** Using the implemented question answering system, we reproduced the dialogue examples by Kido and Kurihara [1] and confirmed that the final synthetic arguments are semantically consistent.

# LLM マルチエージェント弁証法的推論による質問応答

# 目次

# 第1章　はじめに

近年，大規模言語モデル（LLM）を用いた質問応答は，さまざまなアプリケーション，システムで実用化が進んでいる．同時に世界では，多文化共生がますます進んでおり，さまざまな価値観を持つ人々が共に暮らす社会ができている．質問応答には情報検索としての役割に加えて，そのような社会に適応するために，異なる価値観を持つ人々の相互理解を促進する，合意可能な回答を生成する役割も求められている．この要請に対し，複数の役割や価値観を付与したLLMエージェント同士の対話により結論を導くマルチエージェント議論の枠組みが提案されている．しかし既存手法の多くは，対立する立場をもとに勝敗を決めるディベート的議論，あるいは双方の主張を並列した表層的な折衷案に収束しやすい．その結果，対立を保持したまま共通項を抽出し，新たな価値・制度・概念の再構成として解決を導く止揚的な回答生成には至りにくいという課題がある．

そこで本研究では，記号論理で形式化された弁証法的推論の枠組みに基づき，複数のLLMエージェントが論証構築・相互反論・統合を段階的に行う対話プロトコルを提案する．具体的には，Kido, Kurihara らの弁証法議論形式 [1] を参照し，記号論理で示された対話プロトコルと統合の手法を，2体のLLMエージェントを用いた弁証法的推論による質問応答システムに適用する．プロトコルは論証構築フェーズ，反論フェーズ，統合フェーズから構成され，論証構築とそれに対する反論は，2体の対話エージェントによって相互に行われる．

本提案手法を実現するにあたり，取り組むべき課題は以下の2点である．

**議論の収束**

LLMの自由発話による議論は，発話の役割や論点が明示的に制約されないため，話題の逸脱や同一主張の反復が生じやすく，対立が解消されないまま議論が循環するという問題がある．また，どの主張がどの前提に基づき，どの反論によって否定されたのかが不透明となり，収束の判定が困難である．本研究ではこの問題に対し，発話の役割を明示した構造化メッセージスキーマを導入し，各ターンを結論と最小限の前提からなる論証として表現する．さらに，反論が生成不可能となった時点を終了条件として定義することで，議論の進行を形式的に管理し，無限ループを抑制する．これにより，議論の可視性を保ちながら収束性を担保する．

**高次な止揚論証の生成**

Kido, Kurihara によって提案された統合のアルゴリズム（Reasoning Method）は，記号論理に対する操作は可能であるものの，自然言語に対する操作は不可能であり，これをそのまま用いて止揚論証を構築することは困難である．本研究では，統合のための論証スキーマも提案した上で，Reasoning Method を意味的に解釈し，指示プロンプトに反映することで，LLM 上で特化（specialization）および一般化（generalization）の操作を擬似的に実装する．これにより，対立する論証の前提構造を再編成し，単なる折衷ではなく，上位概念に基づく高次な止揚論証の生成を可能とする．また，Kido, Kurihara が実際に行っていた対話例を本提案システム上でも実行し，同様の止揚論証が得られることを確認した．

以下，本論文では 2 章において関連研究を示した上で本研究の位置付けを行い，3 章では実際に提案するシステム，プロトコルについて詳細に説明する．続いて 4 章では評価として，Kido, Kurihara らの対話例を実際に行わせ，5 章でそれらを評価した．最後に，6 章で評価結果から考えられる考察と，本研究の将来的な展望を述べる．

# 第2章　関連研究

## 2.1　記号論理に基づく弁証法的議論とその限界

対立する主張を形式的に扱う枠組みとして，弁証法議論に関する研究が行われてきた．

Sawamura，Umeda ら [2][3] は，Prakken, Sartor ら [4] の拡張論理プログラミングに基づき，弁証法議論の枠組み（Computational Dialectics）を提案している．この枠組みでは，論証の構築と，相手の論証に対する攻撃（rebut, undercut）を段階的に行い，複数のエージェントが対話を通じて弁証法的に統合を行う議論フレームワークを形式定義している．ただし，Sawamura，Umeda らは文献の中で，具体的な止揚論証の導出方法は神託（a sort of oracle）であると述べており，形式定義される止揚論証の意味的実体や，その構築方法は未実装であった．

これに対して，Kido, Kurihara ら [1] は，Specialization および Generalization によって，形式定義された止揚論証を構築する手法を提案した．この手法により，対話を通じて双方の立場を包含する統合を行うための形式的操作が明確化された．

一方で，これらの手法はいずれも記号論理による知識表現を前提としており，議論に用いられる事実や規則，およびそれらの組合せは，あらかじめ定義された知識ベースに強く依存する．そのため，統合可能な概念や評価軸は限定的であり，現実の社会的・倫理的議論に見られるような，文脈に強く依存し，必ずしも明確に定義できない価値観や，議論の過程で新たな視点や概念が創発されるような高次の止揚を，十分に扱うことが難しい．また，形式的に導出された止揚論証が，どのような意味的解釈を持ち，どのような制度や価値の再構成を示すのかは，論理式そのものからは直接的に読み取ることができず，依然として課題として残されている．

## 2.2　大規模言語モデルによる弁証法的議論とその限界

そこで近年では，大規模言語モデル（LLM）を複数のエージェントに分担させ，異なる視点から推論を進めるマルチエージェント対話が注目されている．

Anghel ら [5] は，複数の LLM を用いて弁証法的推論を模倣する枠組みを提案し，(1) 主張（thesis）の生成，(2) 反論（antithesis）の生成，(3) 統合（synthesis）の生成という三段階の推論フローを設計している．同手法では，主張，反論，そ

して統合を行うエージェントを別々に定義し，自由発話によってそれら構築する．さらに生成された推論過程や最終的な統合結果に対して，別の評価エージェントがルーブリックに基づく評価を行う点に特徴がある．

一方で，エージェントは発話を一度だけしか行わず，議論によってエージェント同士のスタンスや価値観の理解などを促進させることはしていない．これは，形式定義された弁証法議論には則っておらず，あくまで擬似的に弁証法的推論を行ったに過ぎない．また，このシステムを多ターンに展開するとしても，自由発話によって構築された論証では，各発話の役割や攻撃関係，および前提の継承関係が構造的に管理されないため，論点の逸脱や同一主張の反復が生じやすく，議論の深化や収束が保証されない．その結果，対立の整理は行われても，前提の再編成や新たな概念形成を伴う止揚には至らず，表層的な対立調整にとどまる可能性が高い

以上より，止揚（統合）を弁証法的に妥当な形で生成するためには，統合に先立って多ターンの議論を通じて争点を段階的に明確化し，各主張の前提構造と，それらの前提間の衝突関係を，攻撃関係および依存関係として構造的に表現・管理する枠組みが必要である．

# 第3章　弁証法的推論による質問応答システム

## 3.1　概要

本システムは，ユーザから与えられたトピック（Issue）と、対話を行うエージェントのスタンスを入力とし，2体のLLMエージェントによる対話的推論を通じて，最終的に止揚論証（統合された回答）を生成するマルチエージェント推論システムである．本研究では，弁証法的推論の規則や進行手順を「弁証法プロトコル」として抽象化し，それを実行するための計算基盤としてシステムアーキテクチャを設計する．本章では，提案プロトコルの内容には立ち入らず，システムを構成する各コンポーネントと，それらの間の情報の流れに焦点を当てる．

## 3.2　システム構成

本システムは，図1に示すように，入力，議論制御，弁証法プロトコル，LLM Agents，出力の5つの主要コンポーネントから構成される．

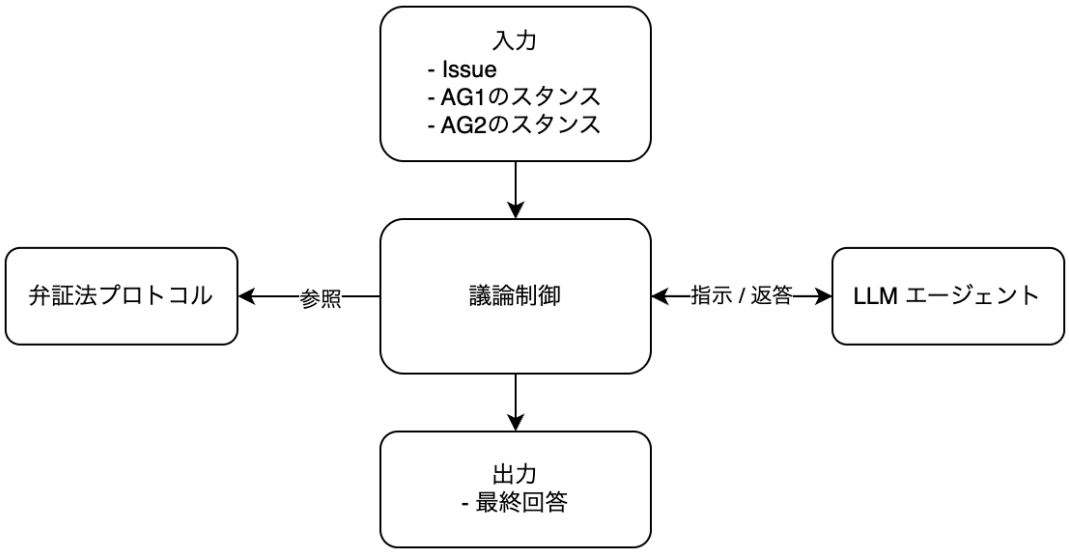


図1: 提案システムの全体アーキテクチャ

入力コンポーネントはトピックと各エージェントのスタンスをユーザーから受け取り，議論制御コンポーネントに渡す．

以下に本システムの入力例を示す．(図2)

Issue: Which camera should we buy?

AG1's stance:
a is a camera.
c is a camera.
a is compact.
a is light.
c has long battery life.
c is user-friendly.
b is over budget.
If a camera is compact and light, we should buy it.
If something is over budget, we should not buy it.
If something is compact and light, it is user-friendly.

AG2's stance:
b is a camera.
a is out of stock.
b has long battery life.
b has high image quality.
If a camera has high image quality and long battery life, we should buy it.
If something is out of stock, we should not buy it.

図 2: 入力例

今回は，Kido, Kurihara の対話例を再現することを目的として，文献中のカメラ選択問題をもとに，各エージェントのスタンスを事実および規則の集合として与えている．AG1 は「コンパクトで軽く，ユーザーフレンドリーなカメラ」を重視する立場，AG2 は「高画質かつバッテリー性能に優れたカメラ」を重視する立場をそれぞれ表しており，両者の価値基準が対立する設定となっている．

このように，各エージェントのスタンスを明示的な命題および推論規則として与えることで，議論フェーズにおいて，どの前提に基づいて主張が構成され，どの規則が衝突しているのかを構造的に追跡可能とする．以降では，本入力を用いて，多ターンの議論を通じて対立点がどのように明確化され，最終的に止揚論証がどのように生成されるかを示す．

次に，弁証法プロトコルコンポーネントは，議論の状態と，エージェントが行う操作を定義したものである．議論制御コンポーネントは弁証法プロトコルを参照しながら議論を進行し，エージェントが行う操作を LLM エージェント

に指示する．LLM エージェントは指示に対して返答を行い，議論制御コンポーネントに返す．使用するモデルは GPT 5-mini である．プロトコルにおいて議論の終了状態に遷移した時，議論制御コンポーネントは議論を終了し，最終回答を出力する．

次章では，本研究の主題である弁証法プロトコルの詳細設計について説明する．

### 3.2.1 プロトコル設計

本研究で提案する弁証法プロトコルは，**論証構築フェーズ**，**反論フェーズ**，**統合フェーズ**の 3 段階から構成される．各フェーズでは，LLM エージェントの出力をあらかじめ定義した論証スキーマに従って制約することで，議論の構造化と収束性，および高次な止揚論証の生成を両立させる．

**論証構築フェーズ**　論証構築では，各エージェントはシステム入力として与えられた Issue および自身のスタンスに基づき，最初の主張（main argument）を 1 つ生成する．この際，主張は単なる自然文ではなく，推論規則の列（rules）から構成される論証として表現される．各規則は，結論を導くための最小限の前提（antecedent.strong）および，証拠不在を仮定する弱否定（antecedent.weak_negation）から，結論（consequent）を導く形式を取り，規則列は結論に至る推論の依存関係を明示する順序構造を持つ．また，各論証は，結論集合（Conc）および仮定集合（Ass）を明示的に保持する．これにより，各エージェントの主張がどの前提に基づいて構成されているかを構造的に把握可能とする．

**反論フェーズ**　反論フェーズでは，以下の手続きを反復的に実行することで，エージェント間の対立を段階的に掘り下げる．

1. 相手の直前の論証（初期主張または再反論）に対して，反論が可能かを判定する．反論が可能な場合は，*rebut*（結論への反駁）または *undercut*（推論規則の前提への攻撃）のいずれかを選択し，新たな論証を構築する．反論が不可能な場合，相手の論証が正当化されたとみなし，議論を終了する．
2. 反論が生成された場合，今度は先に主張したエージェントが，同様にその反論に対して再反論可能かを判定し，可能であれば再反論を構築し，不可能であれば当該エージェントの論証が *defeated*（敗北）したものとして議論を終了する．
3. 上記の手続きを，あらかじめ設定した最大反論回数に達するまで反復する．

反論においても，出力はすべて論証スキーマに従い，攻撃方法（rebut / undercut），前提（strong, weak_negation），結論（consequent），および Conc,

Ass を明示する．これにより，どの前提がどのように攻撃され，どの仮定が否定されたのかを，論証間の関係として形式的に追跡可能とする．

**統合フェーズ**　統合フェーズでは，反論フェーズで得られた議論履歴（複数の Argument）を入力とし，以下の段階を経て最終回答（止揚論証）を構築する．

1. 性質化（Characterization）：対立する 2 つの代表的な論証について，それぞれの前提（antecedent.strong）と結論（consequent）を「性質レベル」に抽象化する．ここでは，特定の対象を指す表現を除去し，どのような条件や価値基準のもとで主張が成立しているかを明示する．
2. 一般化（Generalization）：性質化された 2 つの論証（C1, C2）に基づき，両者が受け入れ可能な合意核（E）を構築する．このとき，共通する前提が存在する場合は必ず含めるとともに，元の前提をそのまま再利用するのではなく，両者を包摂する上位概念として再構成する．これにより，対立していた評価軸を同時に満たしうる条件集合と結論を導出する．
3. 最終回答の生成（Answer）：得られた合意核 E に基づき，E の前提条件を可能な限り満たす具体的な解決策を提示する．この際，元の論証に含まれていた個別の結論をそのまま再利用することは避け，合意核に基づく新たな視点から，両者が納得可能な形で再構成された結論として最終回答（止揚論証）を出力する．

以上のプロトコルにより，本研究では，多ターンの議論を通じて対立する前提構造を明示化し，それらを性質化・一般化することで，単なる折衷ではなく，上位概念に基づく高次の止揚論証を体系的に生成することを可能とする．

### 3.2.2　メッセージスキーマ

本研究では、Kido, Kurihara の統合アルゴリズム及び議論形式に準拠させつつ，LLM を用いた幅広い議論を可能にするため，図 3 に示すようなメッセージスキーマを定義するこれにより，エージェントは論証の構造を把握し，反論のフェーズにおいて，rebut では Conc を，undercut では Ass を明示的に対象にした上で，反論を行うことができる．

```
{
  "Argument": {
    "rules": [
      {
        "id": "r1",
        "antecedent": {
          "strong": [
            "結論を導くための最小限の前提条件1",
            "結論を導くための最小限の前提条件2",
            ...
          ],
          "weak_negation": [
            "ある論証が成り立つという証拠がないという仮定（事実の否定ではない）の前提1",
            "ある論証が成り立つという証拠がないという仮定（事実の否定ではない）の前提2",
            ...
          ]
        },
        "consequent": "strong, weak_negationから導かれる結論（結論のみ）の自然文",
      }
    ],
    ...
    "Conc": [
      "r1のconsequent",
      "r2のconsequent",
      ...
    ],
    "Ass": [
      "r1のweak_negation",
      "r2のweak_negation",
      ...
    ]
  }
}
```

図 3: メッセージスキーマ (JSON)

最後に，弁証法プロトコルの全体像をシーケンス図で表す（図 4）．これの図はこの図では，それぞれの最初の主張をそれぞれ A1，A2 で示す．

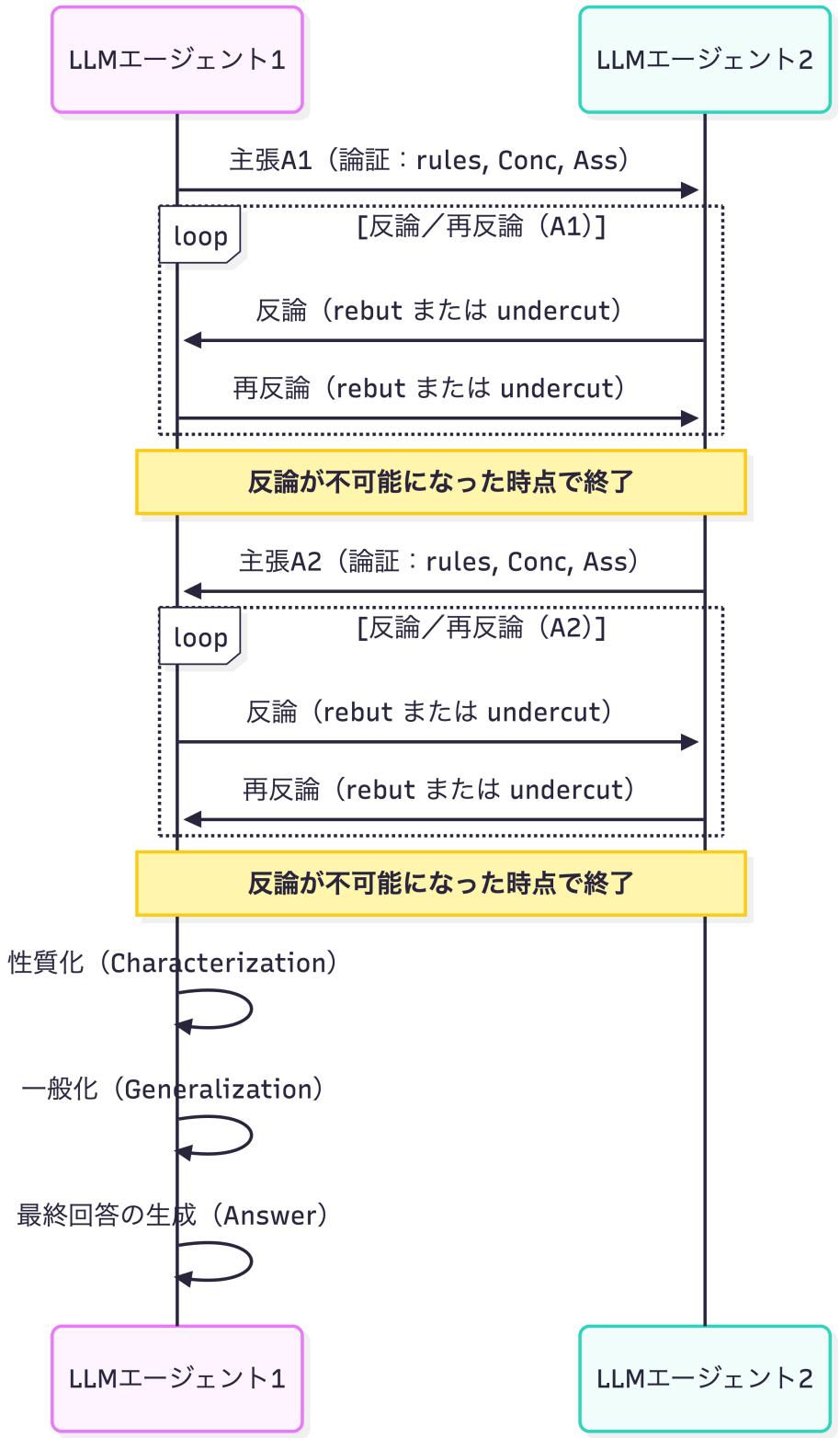


図 4: 弁証法プロトコルを用いたシーケンス図

# 第4章　実験

## 4.1　実験の目的

本章では，前章で定義した弁証法プロトコルを実際の問題設定に適用し，どのような対話が生成され，最終的な止揚論証がどのような過程を経て導出されるのかを，具体的な対話例を通して示すことを目的とする．

また，本章で得られた対話ログおよび最終回答は，次章において，既存研究との整合性および統合結果の妥当性を評価するための基礎データとして用いる．

## 4.2　対話の実行と結果

### 4.2.1　問題設定

題材には，Kido，Kurihara[1] 第6章で用いられているカメラ購買問題を採用した．議題（Issue）は「どのカメラを購入すべきか」とし，2体のエージェントには，それぞれ異なる事実および推論規則からなるスタンスを入力として与えた．本問題設定では，一方のエージェントが「携帯性・操作性」を重視する立場，他方のエージェントが「画質・バッテリー性能」を重視する立場をとることで，評価基準の異なる2つの観点が意図的に対立するよう設計されている．

入力の具体例については，3章で示した図2の通りである．このように，各エージェントの知識および判断規則を明示的な命題と推論規則として与えることで，対話の過程において，どの前提がどの結論を支え，どの規則同士が衝突しているのかを，構造的に追跡可能な設定とした．

### 4.2.2　実行方法

本実験では，前章で定義した論証スキーマ（rules，Conc，Ass）および弁証法プロトコルをそのまま適用した．各ターンにおけるエージェントの出力はすべて構造化された論証として生成・記録され，推論規則の列，結論集合（Conc），および仮定集合（Ass）が明示的に保持される．

これにより，どの前提がどの結論を導いているのか，反論においてどの結論が否定され，あるいはどの仮定が攻撃されているのか，さらに統合フェーズにおいて，対立する前提構造がどのように再編成されているのかを，対話の進行とともに追跡可能とした．

なお，本実験では，新たな規則や外部知識を途中で導入することは行わず，入力として与えたスタンスと，提案プロトコルに基づく生成結果のみを用いて議

論を進行させた．これにより，提案手法そのものの振る舞いを純粋に観察できる実験設定とした．

#### 4.2.3　対話ログの可視化

以下では，主張，反論，再反論，および統合の各段階で生成された論証を，対話ログの画像として提示する（図 5）．

各ログには，当該ターンで用いられた推論規則（rules），結論集合（Conc），仮定集合（Ass），および攻撃の種別（rebut / undercut）が併記されており，論証間の攻撃関係，どの前提が争点となっているか，統合段階で前提構造がどのように抽象化・一般化されているかを視覚的に確認できる．

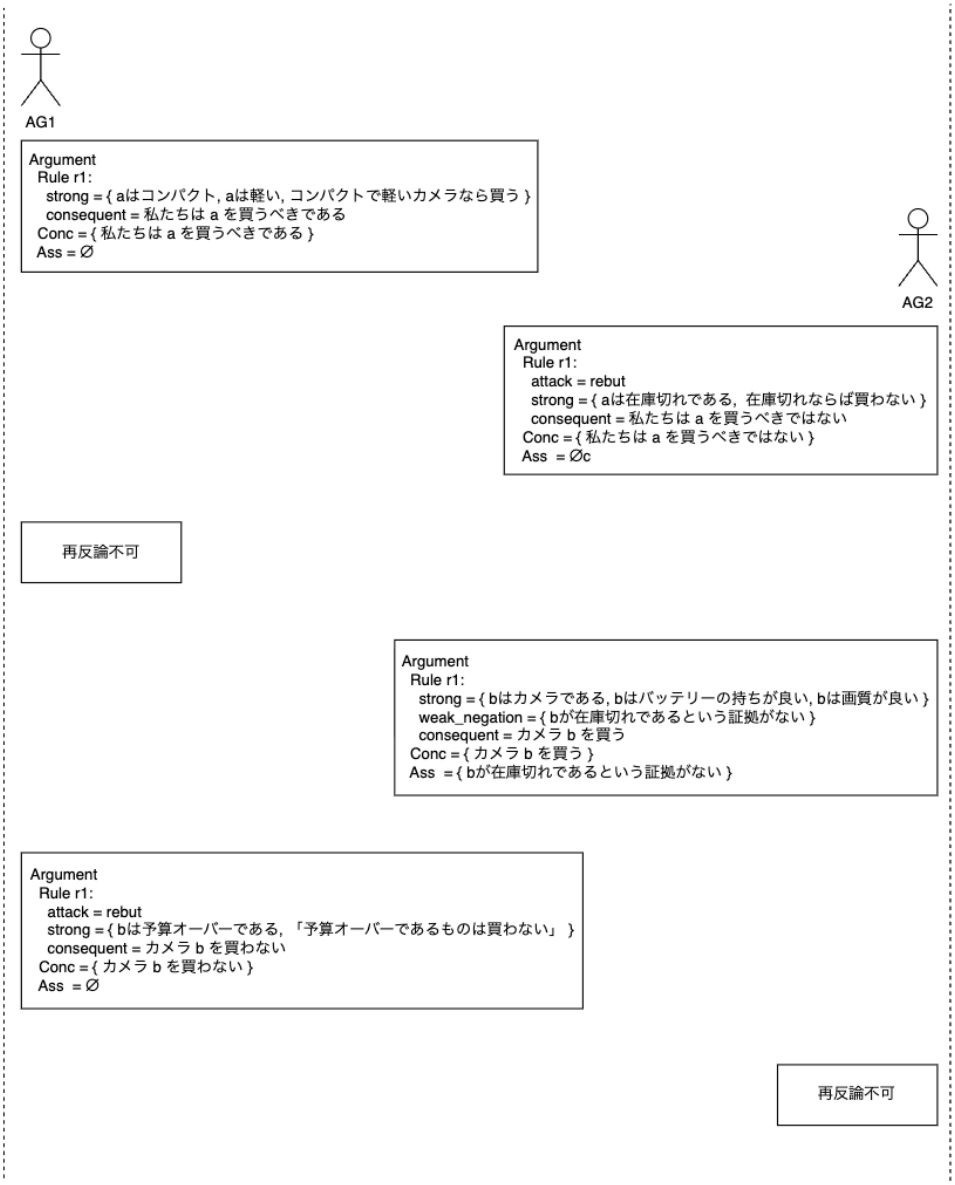


図 5: 対話実行のログ可視化

本図に示すように，本研究のプロトコルでは，多ターンの議論を通じて対立点が段階的に明確化され，最終的に性質化および一般化を経て，単なる折衷ではない止揚論証が構築されていることが確認できる．次章では，これらの対話結果および最終回答を，既存の形式例と比較することで，提案手法の妥当性について評価を行う．

#### 4.2.4　止揚論証の生成

対話の結果，いずれの主張も単独では正当化されない状態となり，反論フェーズで得られた議論履歴に対して統合フェーズが適用された．図 6 は，性質化および一般化を経て得られた合意核に基づき，最終的な止揚論証が構築される過程を示したものである．この結果は，個別の結論の単なる折衷ではなく，対立していた評価軸を上位概念のもとで再構成した解として与えられている．

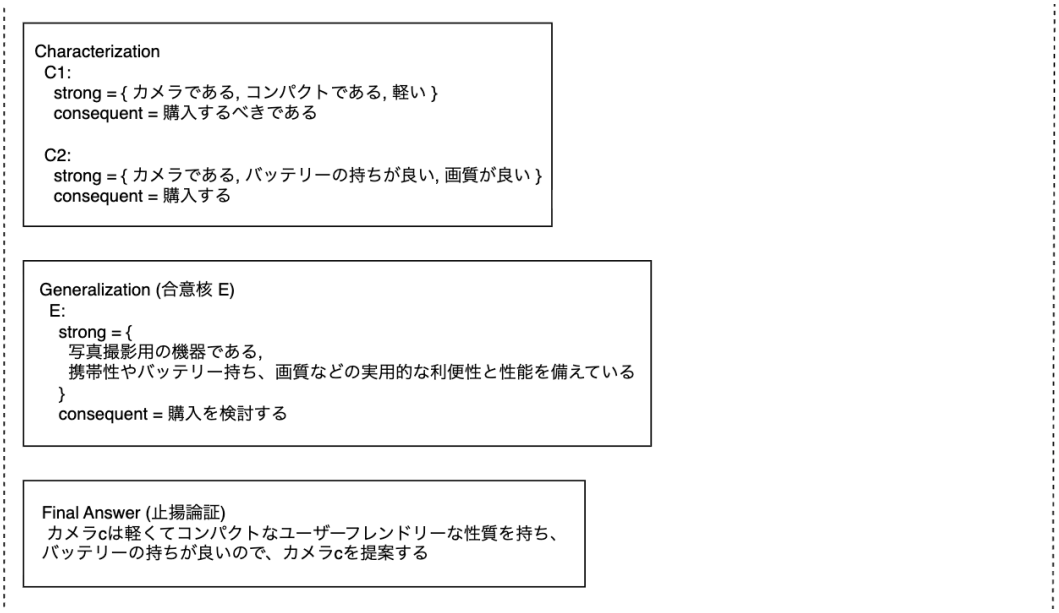


図 6: 統合フェーズによる止揚論証の生成

# 第5章　評価

## 5.1　評価手法

本章では，提案する弁証法プロトコルの妥当性を検証するための評価手法について述べる．本研究における評価の主目的は，提案プロトコルによって生成される議論過程および止揚論証が，既存の計算的弁証法モデルに基づく形式例と，構造および意味の両面において対応しているかを確認することである．

従来の質問応答システムの評価では，単一の正解を持つタスクに対する正解率が主な指標として用いられることが多い．しかし，弁証法的議論に基づく推論では，議題に対する唯一の正解を定義することが困難であり，議論の進行構造，反論関係の形成，および最終的な統合の構成そのものが評価対象となる．本研究では，提案プロトコルの動作が既存研究で示されている形式例をどの程度再現できているかに着目し，生成過程の対応関係をトレースすることによって妥当性を検証する．

本研究における評価方針は，以下の二点に基づく．

- 構造的一致:提案プロトコルによって生成される議論構造が，Kido，Kurihara によって定義された計算的弁証法の形式例と対応しているかを検証する．具体的には，論証の構築順序，反論関係（defeat 関係）の形成，および議論終了後に統合が生成される過程について，形式例と照合しながら確認する．
- 統合の再現性：生成される最終回答が，既存の形式例における統合結果と意味的に対応しているかを検証する．ここでは，新規性や一般性の評価は行わず，対立構造の整理や判断根拠の構成が，既存の統合例をどの程度再現できているかに着目する．

なお，本章の評価は，提案プロトコルが既存の計算的弁証法モデルの振る舞いをどの程度正確にトレースできているかを確認することを目的としており，多様なトピックに対する汎用的性能や定量的な比較評価は対象外とする．

## 5.2　評価に用いる比較データ

本研究では，提案する弁証法プロトコルの妥当性を検証するための比較対象として，Kido，Kurihara によって提案されたカメラ購買問題に関する対話例（文献 [1] 第 6 章）を用いる．本節では，評価の基準となる形式例を明確にするため，文献中で用いられている問題設定および対話過程を省略せずに示す．

まず，文献では次の議題と 2 つの知識ベースが与えられる．

$$Issue = buy(x) \wedge camera(x)$$

$$\begin{aligned} S_1 = \{ & camera(a),\ camera(c),\ compact(a),\ light(a), battery(c, \text{long}), \\ & userFriendly(c),\ overTheBudget(b), compact(x) \wedge light(x) \\ & \wedge camera(x) \rightarrow buy(x), overTheBudget(x) \rightarrow \neg buy(x) \\ & compact(x) \wedge light(x) \rightarrow userFriendly(x) \} \end{aligned}$$

$$\begin{aligned} S_2 = \{ & camera(b),\ outOfStock(a),\ battery(b, \text{long}), resolution(b, \text{high}) \\ & resolution(x, \text{high}) \wedge battery(x, \text{long}), \quad \wedge camera(x) \rightarrow buy(x), \\ & outOfStock(x) \rightarrow \neg buy(x) \} \end{aligned}$$

ここで，$S_1$ および $S_2$ は，それぞれエージェント 1 およびエージェント 2 に与えられる知識ベースを表す．$S_1$ は携帯性や使いやすさを重視する立場に対応しており，$S_2$ は高画質およびバッテリー性能を重視する立場に対応している．両者は評価基準が異なるため，購入すべきカメラについて対立した判断を導く．

次に，文献中で示されている対話の進行を原表記に基づいて示す．ここで $\text{Arg}_i^j$ は,「エージェント $i$ が構築した $j$ 番目の論証」を表す.

まず，エージェント 1 が主張を構築し，エージェント 2 がこれを反論する．

$$\begin{aligned} \text{Arg}_1^1 = [ & compact(a), light(a), camera(a), \quad compact(a) \wedge light(a) \wedge camera(a) \\ & \rightarrow buy(a)] \end{aligned}$$

(I want to buy 'a' Since it is a compact and light camera.)

$$\text{Arg}_2^2 = [outOfStock(a),\ outOfStock(a) \rightarrow \neg buy(a)]$$

(It is out of stock.)

この反論に対して，エージェント 1 はこれを打ち負かす論証を構築できない．

続いて，エージェント 2 が新たな主張を行い，エージェント 1 がこれを反論する．

$$\mathrm{Arg}_2^3 = [resolution(b, \mathrm{high}),\ battery(b, \mathrm{long}),\ camera(b), resolution(b, \mathrm{high}) \\ \wedge battery(b, \mathrm{long}) \wedge camera(b) \rightarrow buy(b)]$$

(Would you like 'b?' Because it is a high-resolution camera with a long battery life.)

$$\mathrm{Arg}_1^4 = [overTheBudget(b),\ overTheBudget(b) \rightarrow \neg buy(b)]$$

(It exceeds the budget.)

この結果，$\mathrm{Arg}_1^1$ および $\mathrm{Arg}_2^3$ のいずれも正当化されず，両エージェントはこれ以上の主張を構築できない状態となる．そこで，エージェント 1 は，両者の根拠（warrant）を統合した新たな主張を構築する．

$$\mathrm{Arg}_1^5 = [userFriendly(c),\ battery(c, \mathrm{long}),\ camera(c), userFriendly(c) \\ \wedge battery(c, \mathrm{long}) \wedge camera(c) \rightarrow buy(c)]$$

(I will then buy "c." Since this is a user-friendly camera with a long battery life.)

この $\mathrm{Arg}_1^5$ は，携帯性・使いやすさとバッテリー性能という異なる評価軸を同時に満たす対象として $c$ を選択するものであり，$\mathrm{Arg}_1^1$ と $\mathrm{Arg}_2^3$ の根拠を包摂した統合的な論証である．

## 5.3 評価結果

本節では，前節で示した Kido，Kurihara の形式例と本研究の実験結果を比較し，提案プロトコルが弁証法的議論および統合過程をどの程度再現できているかを検証する．

まず，議論の進行構造について，本研究のプロトコルでは，各エージェントが主張を構築し，相手がそれに対して反論を行い，いずれかが反論を構成できなくなるまで再反論を反復する手続きを明示的に実装している．この流れは，文

献例における $\mathrm{Arg}_1^1$ と $\mathrm{Arg}_2^2$，および $\mathrm{Arg}_2^3$ と $\mathrm{Arg}_1^4$ の相互攻撃構造と対応しており，提案プロトコルが計算的弁証法における基本的な議論構造を再現していることを示している．

次に，議論終了後の統合過程に関して，本研究では，反論フェーズで得られた論証を性質化および一般化によって抽象化し，両立場を包摂する合意核を導出した上で最終回答を構築する．これは，Kido，Kurihara の例において，$\mathrm{Arg}_1^1$ と $\mathrm{Arg}_2^3$ の warrant を基に $\mathrm{Arg}_1^5$ を構成している操作と意味的に対応している．

さらに，本研究の実験結果においても，最終的に得られた止揚論証は,「a を買う」「b を買う」といった個別の主張の単なる折衷ではなく，携帯性・使いやすさと性能・持続性という異なる評価軸を上位の判断基準のもとで再構成した解として生成されている．この点から，提案プロトコルは，既存の計算的弁証法モデルに基づく統合過程と整合した形で，止揚論証を構築できているといえる．

以上より，本研究の手法は，Kido，Kurihara によって示された形式例と構造的および意味的に対応する議論構造と統合過程を再現しており，弁証法的推論に基づく質問応答の計算モデルとして妥当であることが確認された．

# 第6章　考察

# 第7章　おわりに

本研究では，大規模言語モデル（LLM）を用いたマルチエージェント対話に，計算的弁証法の理論的枠組みを導入することで，対立する主張を形式的に扱い，その上で止揚論証を生成する質問応答システムを提案した．論証を構造化スキーマとして表現し，rebut および undercut に基づく反論フェーズと，性質化および一般化に基づく統合フェーズからなる弁証法プロトコルを設計・実装した点に本研究の特徴がある．さらに，Kido，Kurihara の形式例を評価基準として用いることで，提案手法が既存の計算的弁証法モデルと意味的に対応する議論構造および統合過程を再現できることを示した．

本研究の意義は，LLM を用いた対話型質問応答を，単なる自然言語生成にとどめず，推論過程を明示的に扱う弁証法的推論システムとして位置づけた点にある．対立する立場を単に併記するのではなく，論証構造を明確化し，その上位概念に基づいて統合的な判断を導出する枠組みを与えたことは，価値観や評価基準の異なる立場間の対話を計算的に支援する一つの方法を示すものである．

一方で，本研究にはいくつかの課題も残されている．評価は既存文献の形式例をトレースすることに主眼を置いており，多様なトピックに対する汎用的な性能やスケーラビリティの検証には至っていない．また，議論の進行は外部の制御モジュールに依存しており，エージェント自身が議論状態を把握して次の操作を選択する自律性については十分に検討できていない．

今後の発展としては，主に三つの方向性が考えられる．第一に，本研究では評価対象を特定の形式例に限定したが，本プロトコルをさまざまなトピックに適用し，社会的・倫理的問題や価値観の対立を含む多様な議題において，議論構造および統合過程がどの程度一貫して再現されるかを検証することが重要である．これにより，本手法の適用範囲と限界を体系的に明らかにできると考えられる．

第二に，本研究では特定の LLM を用いてエージェントを構成したが，今後はモデル規模や学習方針の異なる複数の LLM を用いたエージェント間対話を通じて，議論の進行特性や統合結果の差異を比較することが課題である．また，プロンプト設計や内部表現の調整（チューニング）を通じて，反論の生成傾向や統合の抽象度がどのように変化するかを分析することで，弁証法的推論に適したエージェント設計指針の確立が期待される．

第三に，議論過程におけるエージェントの発言の一貫性に対する評価枠組みの構築が挙げられる．具体的には，各エージェントが初期の立場や評価基準を議論の進行中にどの程度維持しているか，また統合段階においてそれらがどのように再構成されているかを定量的に測定する指標の設計が求められる．これにより，最終的な出力結果のみならず，推論過程そのものの安定性や妥当性を評価することが可能になると考えられる．

本研究が，弁証法的推論と大規模言語モデルの融合に向けた基盤的な試みとして，今後の計算的対話システムの発展に寄与することを期待する．

## 謝辞

本研究を行うにあたり，熱心なご指導，ご助言を賜りました村上陽平教授に深く感謝申し上げます．また，普段からお世話になっている社会知能研究室の皆様にも心より感謝申し上げます．

# 参考文献

[1] Sawamura, H. and Umeda, Y.: Computational Dialectics for Argument-based Agent Systems, *Proc. ICMAS 2000*, pp. 271–278 (2000).

[2] Kido, H. and Kurihara, M.: Computational Dialectics Based on Specialization and Generalization, *Proc. JSAI 2008*, pp. 228–241 (2008).

[3] Anghel, C. et al.: Multi-Model Dialectical Evaluation of LLM Reasoning Chains, *Informatics*, Vol. 12, No. 3, p. Art. 76 (2025).

# 付録

## A.1　プロンプトテンプレート

本付録では，提案システムにおいて使用される 5 つのプロンプトテンプレートの詳細を示す.

### A.1.1　主議論構築プロンプト (MAIN_ARGUMENT)

あなたは論理的な議論を構築する専門家です。
以下の質問に対して、事実を基に論理的な議論を構築してください。

質問: {question}

利用可能な事実:
{facts}

以下の形式で議論を構築してください:
- Claim（主張）: 質問に対する答え
- Premises（前提）: 主張を支持する事実のリスト
- Inference Rule（推論規則）: 前提から主張を導く論理的規則

出力形式:
Claim: [主張]
Premises: [前提 1，前提 2，...]
Inference Rule: [推論規則]

### A.1.2　反駁議論プロンプト (DEFEATING_ARGUMENT)

あなたは批判的思考の専門家です。
以下の議論に対する反駁を構築してください。

元の議論:
Claim: {claim}
Premises: {premises}
Inference Rule: {inference_rule}

利用可能な事実:
{facts}

反駁の種類:
{attack_type}

以下の形式で反駁議論を構築してください:
- Claim（主張）: 反駁する内容
- Premises（前提）: 反駁を支持する事実のリスト
- Inference Rule（推論規則）: 前提から反駁を導く論理的規則
- Attack Type（攻撃タイプ）: rebut または undercut

出力形式:
Claim: [反駁主張]
Premises: [前提1, 前提2, ...]
Inference Rule: [推論規則]
Attack Type: [攻撃タイプ]

### A.1.3 特性化プロンプト (CHARACTERIZATION)

あなたは議論評価の専門家です。
以下の議論の特性を評価してください。

議論:
Claim: {claim}
Premises: {premises}
Inference Rule: {inference_rule}

以下の観点から0.0から1.0のスコアで評価してください:
1. Acceptability（受容性）: 前提の信頼性と妥当性
2. Relevance（関連性）: 前提が主張とどの程度関連しているか
3. Sufficiency（十分性）: 前提が主張を支持するのに十分か

出力形式:

Acceptability: [0.0-1.0 のスコア]
Relevance: [0.0-1.0 のスコア]
Sufficiency: [0.0-1.0 のスコア]
Explanation: [評価の理由]

#### A.1.4 一般化プロンプト (GENERALIZATION)

あなたは論理的推論の専門家です。
以下の議論フレームワークから最終的な結論を導いてください。

議論フレームワーク:
{argumentation_framework}

各議論のステータス:
{argument_status}

元の質問: {question}

すべての議論を考慮し、最も支持される結論を導いてください。
複数の有効な議論がある場合は、それらを統合してください。

出力形式:
Conclusion: [最終結論]
Supporting Arguments: [結論を支持する議論のリスト]
Reasoning: [結論に至った理由]

#### A.1.5 回答生成プロンプト (ANSWER)

あなたは質問応答システムの専門家です。
以下の情報を基に、質問に対する最終的な回答を生成してください。

質問: {question}

導出された結論: {conclusion}

議論の詳細:

{argumentation_details}

以下の要件を満たす回答を生成してください:
1. 簡潔で理解しやすい
2. 結論を明確に述べる
3. 必要に応じて根拠を示す
4. 自然な日本語/英語で記述

出力形式:
Answer: [質問に対する回答]
Confidence: [0.0-1.0 の信頼度スコア]
Justification: [回答の根拠]