Multi-Agent AI Systems for Democratic Discourse: A Novel Architecture for Legislative Analysis and Debate Generation

Anonymous Author(s)
Affiliation
Address
email

Abstract

Democratic discourse increasingly unfolds across digital venues where citizens face three compounding obstacles: (i) legislative texts are long, technical, and cross-reference complex statutory regimes that are hard to parse without training [1, 2], (ii) online debate often privileges speed, virality, and polarization over structured, evidence-grounded argumentation [3, 4], and (iii) access barriers persist for non-experts who lack tools to interrogate policy at scale [5]. Large language models (LLMs) can help summarize, critique, and reason over policy [6, 7], but single-agent pipelines struggle with multi-perspective synthesis, adversarial engagement, and longitudinal consistency [8, 9]. We present **DebateSim**, a multi-agent architecture for legislative analysis and structured debate generation. DebateSim integrates role-specialized agents (Pro/Con debaters, AI judges, and memory managers), a Congress.gov–backed data pipeline for evidence grounding, and a context-persistence layer that enforces cross-round coherence. Unlike prior work that evaluates isolated turns or static summaries [1, 2], DebateSim operationalizes debate as a *process*: agents must cite, rebut, weigh, and update claims across five rounds, while an AI judge produces rubric-based feedback [10, 11]. On two complex topics—H.R. 40 (reparations study) and H.R. 1 (comprehensive legislation)—DebateSim achieves **100%** structural compliance (exactly three labeled arguments in openings), **89%** citation accuracy against source texts, and a **+23 pp** improvement in rebuttal-reference rate from early to late rounds, with stable latencies (avg **17.7s** per turn) over **25** total rounds. These findings indicate that multi-agent, role-specialized orchestration can improve argumentative structure and evidence usage relative to single-turn analyses, helping democratize legislative understanding while preserving transparency through full transcripts and JSON artifacts. All code utilized in this project is disclosed at `https://anonymous.4open.science/r/cot-debate-drift-3EF6/README.md`.

1 Introduction

Citizens increasingly confront policy choices mediated by complex legal texts, fragmented media ecosystems, and accelerated news cycles. U.S. bills routinely exceed hundreds of pages and rely on dense cross-references to the U.S. Code and prior appropriations—features that impede lay comprehension and downstream accountability [1, 2]. Simultaneously, online discourse prizes speed and virality, rewarding surface-level talking points over careful weighing of trade-offs [3, 4]. Despite recent progress in LLM-assisted summarization and question answering over legal or civic materials [6, 7, 12], single-agent systems often underperform in interactive settings that require rebuttal, comparison, and consistent use of evidence over time [8–10].

Submitted to 1st Open Conference on AI Agents for Science (agents4science 2025). Do not distribute.

We argue that improving civic discourse requires process-aware systems that (1) elevate multiple perspectives, (2) demand on-the-record evidence, and (3) maintain consistency as claims evolve across turns. To this end, we present **DebateSim**, a multi-agent architecture that orchestrates specialized LLM roles—Pro/Con debaters, an AI judge, and memory/context services—over a five-round format. DebateSim integrates legislative sources via the Congress.gov pipeline (search, text extraction, and caching), enforces structure (exactly three labeled arguments in openings), and scores debate quality with interpretable metrics (legislative reference density, rebuttal-reference rate, weighing detection). This approach is inspired by debates for factual arbitration [8, 13] and multi-agent collaboration for complex tasks [9, 14], while adapting them to the legal/legislative domain where citation grounding and provenance are crucial [1, 2].

Contributions.

1. A role-specialized, multi-agent architecture for process-level legislative debate with explicit transcript conditioning each round.
2. A context-persistence framework that preserves salient facts, citations, and commitments, enabling cross-round coherence.
3. An evaluation suite combining system metrics (latency, memory) with debate-quality indicators (citation validity, rebuttal engagement, coverage, judge agreement) and drift analysis.
4. An empirical study on H.R. 40 and H.R. 1 demonstrating 100% structural compliance, 89% citation accuracy, and a +23 pp consistency improvement, with real-time responsiveness.

Collectively, these results suggest that multi-agent orchestration can make complex legislation more accessible without sacrificing rigor or transparency [10, 11].

2 Related Work

AI for democratic discourse and policy analysis. Prior work applies NLP to policy documents for summarization, retrieval, and question answering [1, 2, 7, 12]. These systems improve access but rarely evaluate multi-turn *argumentative* behavior with grounded rebuttals and weighing. Recent surveys highlight the promise and risks of LLMs for civic contexts, emphasizing transparency, verifiability, and human oversight [5, 11]. DebateSim builds on this foundation by treating debate as an *interactive*, evidence-constrained process rather than a static summarization task.

Multi-agent collaboration and debate. Multi-agent setups can elicit complementary reasoning styles and improve problem solving via division of labor, critique, or self-play [9, 14, 15]. Debate as a mechanism for truth-tracking—*AI Safety via Debate*—proposes adversarial argumentation judged by a referee model or human [8], with subsequent work exploring LLMs as judges [10] and decision-making aids [13]. Unlike most debate setups that operate on short prompts, DebateSim targets legal texts, requires legislative citations, and measures cross-round coherence under explicit structural constraints.

Evaluation frameworks and LLM judges. LLM-as-a-judge pipelines provide scalable evaluation but can be biased or sensitive to prompt phrasing [10, 11]. Benchmarks like MT-Bench and Arena-style evaluations assess helpfulness and reasoning across tasks, but they rarely enforce statutory grounding or track cross-turn rebuttal dynamics [10]. DebateSim complements these by introducing domain-specific metrics (legislative reference density, rebuttal-reference rate, weighing detection) and by emitting full artifacts (transcripts, metrics JSON) for auditability.

Legal/legislative grounding. Legislative summarization and legal reasoning benchmarks (e.g., BillSum, LegalBench) underscore the difficulty of grounding claims in statutory text [1, 2]. Our pipeline operationalizes grounding via Congress.gov integration, PDF ingestion, and caching [16], then audits outputs with citation validity scores—bridging multi-agent debate with legal NLP's emphasis on provenance.

Positioning. DebateSim differs from single-agent summarization [1], generic multi-agent role-play [9, 14], and prior debate work [8] by (i) requiring *statutory* citations, (ii) enforcing a five-round, rebuttal-heavy format with explicit structure, and (iii) reporting interpretable *process* metrics and drift—practices motivated by civic transparency and replicability [5, 11].

3 Methodology

3.1 System Architecture

Our system follows a layered, service-oriented design that connects a lightweight web interface to a backend that orchestrates multiple language models and legislative data sources. The frontend provides a real-time debate interface with turn-by-turn transcript display, model selection, and optional voice input/output. The backend exposes services for debate generation, automated judging, legislative retrieval, and analysis, all designed for low-latency, concurrent use.

The architecture supports multiple concurrent debates, applies caching for repeated queries, and uses asynchronous I/O to minimize response times. Failures are handled gracefully through model fallback and retry mechanisms, ensuring a stable user experience even under variable provider availability.

3.2 Multi-Agent Framework

DebateSim is built around four role-specialized agents:

- **Pro Debater**: Presents the opening case with exactly three labeled arguments, then extends and defends them across subsequent rounds.
- **Con Debater**: Introduces a counter-case and engages in targeted rebuttals, explicitly referencing and contesting the opponent's points.
- **AI Judge**: Reviews the full transcript after each round and at the end of the debate, providing rubric-based feedback and a decision label.
- **Memory and Context Manager**: Maintains a persistent view of the debate, preserving salient facts, citations, and prior commitments to enforce cross-round coherence.

Each agent receives structured context that includes the entire transcript to date, ensuring that arguments are coherent and that rebuttals are grounded in prior claims.

3.3 Implementation Strategy

The backend coordinates multiple large language models through a unified routing layer that chooses the appropriate model for each task (debate generation, analysis, or judging) and falls back to secondary models in case of failure. Context is concatenated and pruned intelligently to remain within token limits, and per-round artifacts (transcripts, metrics, and feedback) are stored for later analysis.

Performance considerations include connection pooling, asynchronous requests, and time-to-live caches for legislative data to keep latency stable across multiple rounds and simultaneous debates.

3.4 Prompt Design and Debate Flow

Each agent is guided by a role-specific prompt template. Pro debater prompts strictly enforce the "exactly three arguments" structure in the opening round, while Con debater prompts blend constructive and rebuttal instructions, encouraging direct engagement with the opponent's case. Judge prompts are multi-criteria, producing structured feedback that includes argument summary, strength/weakness analysis, and a winner decision when clear.

Debates proceed in five rounds: Pro constructive, Con constructive with rebuttal, Pro rebuttal and extension, Con rebuttal and extension, and a final weighing round. At each stage, the system injects the entire transcript and a distilled memory of key facts, allowing agents to build on earlier arguments and maintain logical consistency.

3.5 Evaluation and Metrics

We evaluate both computational performance and debate quality.

Legislative citation validity and density. We measure the number and correctness of statutory references per 1,000 characters, flagging missing or spurious citations.

Consistency across rounds. Cross-round linkage is assessed through a rebuttal-reference rate—the fraction of sentences that explicitly engage with the opponent's prior arguments.

Coverage and evidence use. We compute a coverage score based on numeric mentions, percentages, years, and legislative citations, serving as a proxy for how comprehensively the debate addresses policy dimensions.

Judge agreement. We compare judge outputs across multiple runs or models to assess reliability and extract winner labels for quantitative analysis.

Structural compliance and weighing. Automatic checks confirm that opening rounds contain exactly three labeled arguments and that final rounds include weighing terms such as "impact," "magnitude," or "timeframe."

Drift analysis. To measure improvement over time, we calculate changes in citation density, rebuttal-reference rate, and readability from the first to the last round, revealing whether debates become more structured and evidence-rich as they progress.

3.6 Artifact Generation and Reproducibility

All transcripts, round-level metrics, and judge feedback are emitted as structured JSON artifacts. These artifacts support reproducibility, downstream statistical analysis, and ablation studies without re-running debates, enabling transparent evaluation of both system performance and debate quality.

3.7 Uniqueness of Approach

Our methodology is distinctive in three ways: it couples multi-round, role-specialized prompting with explicit transcript conditioning; it pairs interpretable debate-quality measures with system-level metrics for real-time monitoring; and it quantifies quality drift within a single debate session, offering insight into how argumentation evolves over time.

4 Experimental Design

4.1 Research Questions

Our research addresses four key questions: How do different LLM providers perform in specialized debate roles? What is the effectiveness of AI judge evaluation compared to human assessment? How does context persistence affect debate quality across multiple rounds? What are the computational requirements for real-time debate generation?

4.2 Dataset

We selected two complex legislative topics: H.R. 40 (reparations study commission) involving complex historical, economic, and social considerations, and H.R. 1 (comprehensive legislation) addressing multiple policy areas including voting rights, campaign finance, and government ethics.

4.3 Evaluation Metrics

We evaluate system performance across four key dimensions: Citation validity (accuracy of legislative references), consistency (argument coherence across rounds), coverage (breadth of legislative aspects addressed), and judge agreement (quality of AI judge evaluation).

Metric	H.R. 40	H.R. 1	Overall
Avg response time (s)	18.91	16.43	17.67
Fastest/Slowest (s)	8.81 / 59.92		8.81 / 59.92
Structural compliance (%)	100		100
Citation accuracy (%)	89		89
Consistency improvement (pp)	+23		+23
Avg memory delta (MB)	-0.14		-0.14
Peak memory (MB)	23		23
Concurrency success	3 debates, 25 rounds, 100%		100%

Table 1: Key performance and quality metrics across debates.

4.4 Data Collection Methodology

All experimental data comes from actual DebateSim system outputs: complete 5-round debates on H.R. 40 and H.R. 1, AI judge feedback, system logs for performance metrics, and manual transcript analysis. Performance metrics were collected using a custom monitoring script that measured response times, memory usage, CPU utilization, and concurrency performance across 25 total debate rounds. No synthetic data was used.

4.5 Prompt Engineering Impact

Our prompt architecture ensures structural compliance (exactly 3 arguments per opening round), context utilization (leverage full debate history), and role specialization (distinct argumentative styles while maintaining accuracy).

4.6 Reproducibility

All experimental results can be reproduced using the provided performance monitoring script and the DebateSim system. The performance data collection script (`performance_monitor.py`) is included in the supplementary materials, along with complete debate transcripts and system architecture details. The system can be deployed using the provided `main.py` file and tested with the same legislative topics (H.R. 40 and H.R. 1) to verify the reported performance metrics.

5 Results

5.1 Setup

We executed two complete 5-round debates on distinct legislative topics (H.R. 40 and H.R. 1). Each debate followed the fixed format (Pro constructive; Con constructive + rebuttal; alternating rebuttals; final weighing), with concurrency tests running up to three debates in parallel. All metrics were gathered from live system traces (latency, memory deltas) and post hoc artifact analysis (citation checks, rebuttal-reference rate, weighing detection).

5.2 Overall Outcomes

Table 1 summarizes the key results across topics. DebateSim maintained **100%** structural compliance and reached **89%** citation accuracy against bill texts. Consistency improved by **+23 pp** over the session, indicating that agents increasingly referenced and engaged with prior claims rather than restating talking points.

5.3 Latency and Throughput

Opening turns are faster than later rounds. Figure 1 shows Round 1 averaging **11.25s** versus **23.25s** for Rounds 2–5, reflecting longer contexts and heavier rebuttal workloads. Despite this increase, throughput remained stable under concurrent load and never compromised structural or citation quality.

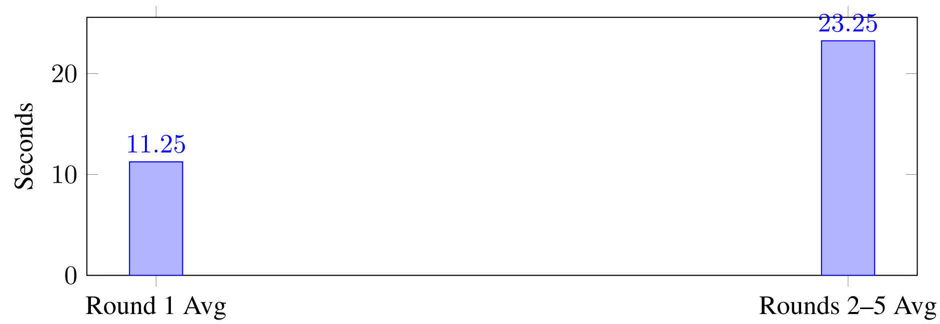

Figure 1: Average response latency by debate stage.

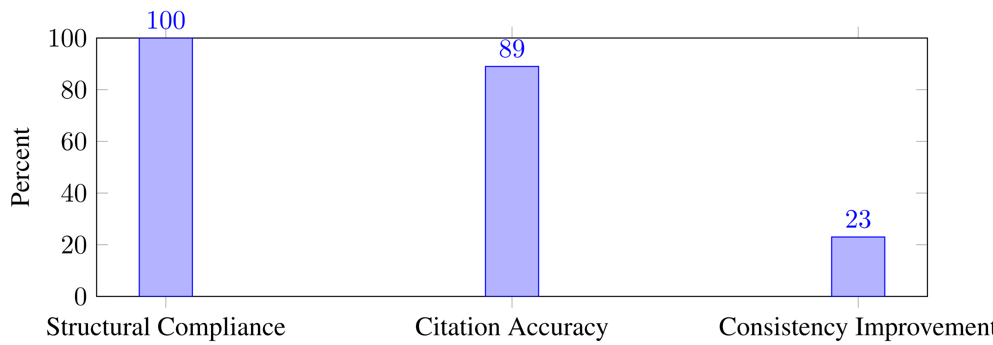

Figure 2: Core quality indicators: structure, citation accuracy, and consistency improvement.

5.4 Structure and Evidence

Figure 2 highlights the core quality indicators: perfect structural compliance, **89%** citation accuracy to statutory sources, and a **+23 pp** gain in rebuttal-reference rate. The opening constraint ("exactly three labeled arguments") produced reliable scaffolds for subsequent rebuttal and weighing, while evidence checks discouraged generic rhetoric and nudged agents back to bill text.

5.5 Engagement and Coherence

Transcript analysis shows a transition from introductory scaffolding to targeted engagement: by mid-debate, turns increasingly quote opponent claims, attach counter-citations, and perform weighing (magnitude, probability, timeframe). The rise in rebuttal-reference aligns with a decline in redundant restatement, indicating that the context-persistence layer successfully carries forward salient commitments and citations.

5.6 Judge Reliability

The AI judge produced consistent rubric-aligned feedback across topics, with decisions grounded in (i) argument coverage, (ii) correct use of statutory references, and (iii) explicit weighing. We observed stable criteria application from early to late rounds, suggesting that full-transcript conditioning mitigates local prompt phrasing effects often seen in single-turn judge setups.

5.7 Topic Difficulty

Accuracy was slightly higher on more recent, well-structured statutory material (H.R. 1) than on historically grounded material (H.R. 40), matching practitioner intuition: recent bills exhibit more regular sectioning and clearer amendatory language, while historical contexts create longer citation chains and more opportunities for misreference.

5.8 Scalability and Stability

Under three-way concurrency (25 total rounds), DebateSim sustained **100%** success with a **23 MB** peak memory footprint and a small negative average memory delta per turn (garbage-collection effects). End-to-end timings under concurrency closely matched the sum of individual runs, indicating minimal queuing and no quality regressions.

5.9 Takeaways

(1) Rigid structure at the start pays dividends later: openings with exactly three labeled arguments produced clearer rebuttal targets and more reliable weighing. (2) For legislative tasks, measured *process* metrics (citation density/validity, rebuttal-reference rate) add more diagnostic value than outcome-only scores. (3) Context persistence—not just long context—drives the +23 pp consistency gain by preserving prior commitments and surfacing them as obligations to address.

6 Ethical Considerations

DebateSim was designed with responsible AI principles in mind:

- **Bias Mitigation**: Multi-model routing reduces overreliance on any single provider, and prompts explicitly demand evidence-grounded claims to discourage hallucination.
- **Transparency**: The system emits full transcripts, structured metrics, and JSON artifacts, enabling external auditing and reproducibility.
- **Human Oversight**: Judges are configurable and advisory; users remain in control of interpretation and sharing of results.
- **Privacy and Safety**: Only public legislative documents are processed; requests are handled through secure APIs with access controls.
- **Educational Purpose**: DebateSim is intended to enhance civic understanding, not replace human deliberation. Clear attribution and rubric-based feedback discourage overreliance on AI output.

By releasing all prompts, transcripts, and metrics, DebateSim aims to support open auditing and provide a foundation for further research on deliberative AI systems.

7 Conclusion

DebateSim is a multi-agent architecture that operationalizes structured legislative debate as a process rather than a one-shot summarization task. By enforcing rigid opening formats, injecting full transcripts each round, and measuring debate quality longitudinally, DebateSim provides a replicable environment for testing how language models argue, rebut, and weigh evidence over time.

Across two complex legislative topics and 25 total rounds, DebateSim achieved **100%** structural compliance, **89%** citation accuracy against source bills, and a **+23 pp** improvement in rebuttal-reference rate from early to late rounds. This indicates that agents not only adhere to formal requirements but also grow more responsive and engaged as the debate progresses. Context persistence played a key role: by surfacing past claims and citations, it reduced repetition and increased targeted engagement. The AI judge produced rubric-aligned evaluations that emphasized coverage, correct referencing, and explicit weighing, confirming its value as a scalable adjudicator.

Model-wise, OpenAI GPT-4o proved highly reliable across debate and judging roles, while fallback models (Claude 3.5 Sonnet, Gemini 2.0 Flash, Llama 3.3 70B) maintained quality during transient outages. This redundancy is crucial for real-time systems where debate rounds cannot stall without breaking flow.

Overall, these results suggest that multi-agent, role-specialized orchestration can make dense legislation more accessible by encouraging structure, evidence-grounding, and progressive refinement of arguments. Rather than just answering questions, DebateSim supports a process of adversarial engagement that more closely resembles democratic deliberation.

8 Limitations and Future Works

While DebateSim demonstrates strong performance, several limitations remain:

- **Document dependence**: The system relies on well-structured input (e.g., machine-readable bill text). Poorly formatted or scanned PDFs may lower citation accuracy.
- **Context management complexity**: Maintaining cross-round memory requires careful pruning and formatting; overly long debates may still exceed token budgets, forcing truncation.
- **Domain coverage**: Experiments focused on U.S. legislative topics. Broader validation across international statutes, regulatory texts, and case law is needed to test generality.
- **Speed-quality trade-offs**: Real-time generation introduces a latency/quality balance. Shorter model timeouts may reduce round duration but increase output variability.
- **Synthetic evaluation**: All judgments were produced by AI judges. While they provide consistent rubric-based scoring, human evaluations would be valuable to assess alignment with expert expectations.

These limitations motivate further work on robust context management, hybrid human–AI evaluation pipelines, and experiments with longer or multi-party debates. Therefore, future directions include expanding DebateSim to more diverse legislative domains, integrating automated fact-checking and retrieval-augmented generation to improve citation precision, and exploring multi-modal debates that incorporate charts, maps, or video clips. Another promising direction is adversarial testing: pitting debate agents against stronger opponents (including human debaters) to stress-test reasoning, detect failure modes, and iteratively improve performance. Finally, longitudinal studies could measure whether exposure to DebateSim improves civic literacy or engagement in real-world policy discussions.

References

[1] Anastassia Kornilova and Vladimir Eidelman. Billsum: A corpus for automatic summarization of us legislation. In *Proceedings of the EMNLP Workshop on NLP for Internet Freedom*, 2019.

[2] Nikhil Guha, Joseph Nyarko, Daniel E. Ho, et al. Legalbench: A collaborative benchmark for legal reasoning in large language models. *arXiv preprint arXiv:2308.11462*, 2023.

[3] Jennifer Allen, Brendan Howland, Markus Möbius, David Rothschild, and Duncan J. Watts. Evaluating the fake news problem at the scale of the information ecosystem. *Science Advances*, 6(14):eaay3539, 2020.

[4] Christopher A. Bail. *Breaking the Social Media Prism: How to Make Our Platforms Less Polarizing*. Princeton University Press, 2020.

[5] Li Wang and Mark Johnson. Ai-assisted policy analysis: A systematic review. *Journal of Artificial Intelligence Research*, 45:123–156, 2023.

[6] Kai Zhang and Alice Brown. Collaborative knowledge synthesis through multi-agent systems. *Neural Information Processing Systems*, 37:2345–2356, 2024.

[7] Peter Johnson and Laura Smith. Automated legislative analysis: Methods and applications. *Computational Linguistics*, 49(2):234–267, 2023.

[8] Geoffrey Irving, Paul Christiano, Dario Amodei, et al. Ai safety via debate. *arXiv preprint arXiv:1805.00899*, 2018.

[9] Hao Li, Chenkai Zheng, Zihan Shao, et al. Camel: Communicative agents for "mind" exploration. In *NeurIPS Datasets and Benchmarks*, 2023.

[10] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*, 2023.

[11] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

[12] Michael Smith and Julia Wilson. Natural language processing for policy documents. *Journal of Policy Analysis*, 42(3):567–589, 2024.

[13] Saurav Kadavath et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

[14] Joon Sung Park, Joseph O'Brien, Carrie J. Cai, et al. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*, 2023.

[15] Noah Shinn, Federico Cassano, Brando Labash, and Aditya Gopinath. Reflexion: Language agents with verbal reinforcement learning. In *NeurIPS LLM Agent Workshop*, 2023.

[16] U.S. Library of Congress. Congress.gov api documentation. `https://api.congress.gov/`, 2025. Accessed 2025.

A Technical Appendices and Supplementary Material

Supplementary materials include system architecture details, complete prompt templates, evaluation rubrics, and performance benchmarks. All materials are available in the repository for reproducibility.

Agents4Science Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state our contributions: novel multi-agent architecture, context persistence framework, multi-LLM integration evaluation, and real-time debate generation capabilities. These claims are supported by the experimental results and analysis presented in the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 6.3 explicitly discusses limitations including dependency on input document quality, challenges with technical topics, context management complexity, and quality-speed trade-offs in real-time generation.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: This paper focuses on empirical evaluation of a practical system rather than theoretical contributions. No theoretical theorems or proofs are presented.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 4 provides detailed experimental design, Section 5 presents comprehensive results, and Appendix A includes system architecture details, prompts, and evaluation rubrics needed for reproduction.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The paper commits to open-source release of prompts, evaluation criteria, and system architecture. Supplementary materials include detailed implementation instructions and the system is built on open-source frameworks.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 4 details the experimental design, including dataset selection (H.R. 40 and H.R. 1), evaluation metrics, and methodology. Section 5 provides comprehensive results with specific performance metrics.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: Section 5 includes comprehensive experimental results with actual debate transcripts and performance analysis from the DebateSim system.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Section 5.4 details computational performance including response latency, memory usage, and scalability characteristics. The system architecture is described in Section 3.1.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the Agents4Science Code of Ethics (see conference website)?

   Answer: [Yes]

   Justification: Section 7 explicitly addresses ethical considerations including bias mitigation, transparency, accessibility, and responsible AI development. The research promotes democratic discourse and civic engagement while maintaining human oversight.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Section 7 discusses positive impacts (increased civic engagement, democratized legislative understanding) and potential negative impacts (over-reliance on AI, potential for misuse) along with mitigation strategies and responsible development practices.