

An LLM-Powered Real-Time Debate Training System with AI Opponent, Adaptive Mentorship, and Multimodal Abuse Detection

Adarsh A
PG Scholar

Department of Computer Applications
Amal Jyothi College of Engineering (Autonomous)
Kanjirappally, Kerala, India
adarsha2026@mca.ajce.in

Anit James
Assistant Professor

Department of Computer Applications
Amal Jyothi College of Engineering (Autonomous)
Kanjirappally, Kerala, India
anitjames@mca.ajce.in

Abstract—The researchers developed a new system for real-time debate training which uses Large Language Models (LLMs) to function as three different systems that compete with users while teaching them privately and detecting multiple types of abuse through one system during live voice interactions. The system provides ongoing feedback during live debates because it differs from Debatrix [1] which only evaluates completed debates through automated judging. The system uses WebRTC-based voice transport [16] to deliver audio through its dual-speech recognition system which combines Web Speech API with Vosk WASM fallback and uses Socket.IO to manage real-time events while the Groq API performs LLM inference. The system uses a dual-layer dynamic health and credibility scoring model which replaces traditional static verdicts to create an educational experience that resembles a video game. The system uses a three-strike moderation system to oversee both live voice transcripts and chat messages with the help of the Hugging Face unbiased-toxic-roberta model [7]. The evaluation process shows that the system can perform real-time adversarial rebuttals while providing private coaching during every spoken sentence and enforcing multimodal safety measures which no other existing debate AI system can accomplish.

Keywords—Large Language Models, debate training, real-time AI, AI opponent, adaptive mentorship, abuse detection, speech recognition, WebRTC, Socket.IO, toxicity detection, credibility scoring

I. INTRODUCTION

The availability of tools that enable live debate practice remains restricted despite their educational value and professional significance. Existing AI systems for debate—such as Debatrix [1] and IBM’s Project Debater [2]—are primarily designed for post-hoc evaluation or structured argument generation, not for real-time interactive training with a live user.

The introduction of Large Language Models [13] together with multiagent debate frameworks [9, 10] has created fresh opportunities for AI-based interactive argumentation. Existing systems use these capabilities to enhance model performance

and execute evaluations, but they do not support human skill development during actual operations.

The paper proposes a system that unifies three distinct functions which operate together during a live debate. First, the system functions as an adversarial AI Opponent which produces simultaneous real-time rebuttals based on opponent actions. Second, the system operates as a private Adaptive Mentor which provides coaching to users about their rhetorical abilities and argument construction while providing guidance for upcoming actions after each spoken statement. In delivering response assessment results to users, the system will maintain total security without disclosing assessment results to its competitors. The system functions as a Multimodal Abuse Detection system which tracks both voice transcripts and chat messages to identify toxic content. The system implements a three-strike moderation system which enforces progressive disciplinary measures for its monitoring operations.

The main achievements of this research include the development of the following: The system enables real-time adversarial debates through LLM technology which functions in live WebRTC voice rooms. The system introduces dynamic health and credibility scores which replace traditional static assessment methods after user sessions. The system operates a private mentorship system which adapts to each user while it functions with the opponent pipeline. The system uses a unified multimodal moderation system which handles both voice and chat streams. The system implements a hybrid speech recognition system which delivers cross-browser compatibility through its use of Web Speech API and Vosk WASM fallback.

II. LITERATURE REVIEW

A. Automated Debate Judging and Evaluation

The automated debate evaluation system known as Debatrix which was introduced at ACL Findings 2024 functions as the most advanced evaluation system. The system uses a pipeline LLM judge which assigns scores according to three specific dimensions that assess both argument quality and source credibility and language proficiency. The system assesses its performance through the PanelBench benchmark which uses DebateArt and British Parliamentary (BP) competition

transcripts as its foundation. The system operates as a post-hoc tool because it requires completed transcripts to generate all verdicts after the debate has ended. The system does not support any realtime features which include adversarial practice and per-user coaching and content moderation capabilities. The researchers present a Multi-Agent Debate (MAD) framework which enables multiple LLM agents to present their arguments through a “titfor-tat” mechanism while a judge supervises the entire debate process. The research shows that LLM-driven debates enhance reasoning skills by making complex tasks easier to solve. The research focuses on improving model performance instead of enhancing human training abilities.

B. Multi-Agent Debate Frameworks

Du et al. [9] prove that multiple LLM systems create better results when they conduct their response evaluation process through three distinct rounds of testing because this method improves their ability to deliver accurate information and solve mathematical problems. The “society of minds” technique produces better results because it decreases hallucinations while increasing accuracy and establishes a basic understanding that supports our system’s ability to handle adversarial AI. Khan et al. [11] demonstrate that more persuasive LLMs show better results during educational debates because LLMs drive adversarial interactions. Irving et al. [12] presented AI safety through debate as a system that enables organizations to supervise their operations at scale while establishing the theoretical basis for controlled adversarial LLM discussions.

C. Computational Argumentation

Stab and Gurevych [3] established the basic principles of argument mining when they developed a method to analyze text arguments through the detection of claims and premises and their supportive connections. The research on argument quality assessment [4] developed a system which evaluates argument strength through automated scoring to establish a framework for assessing individual arguments.

The researchers Chen et al. [14] investigate how LLMs can advance computational argumentation through their ability to generate arguments and extract information and evaluate argument quality in this field of study.

D. Large-Scale Debate AI

The IBM Project Debater system demonstrated its ability to create and present successful debate arguments through its use of argument mining, evidence retrieval, and speech synthesis technologies. The system functions as a speaking agent because it does not provide training functions which would help users learn through detection of their weaknesses and identification of harmful content.

E. Pretrained Language Models for NLP

Devlin et al. [13] developed BERT through a bidirectional transformer which underwent pretraining on unlabelled text and achieved top performance on eleven NLP tasks after finetuning. The unbiased-toxic-roberta model used in our moderation pipeline It is a RoBERTa-based [13] fine-tuned classifier that

can directly benefit from having already pretrained representations.

F. Real-Time Speech Systems and Tutoring

Prior work on spoken dialogue systems [8] and voice-based tutoring [5] established the feasibility of spoken interaction for learning. These systems typically deliver scripted feedback rather than LLM-generated dynamic responses to live user speech. Wachsmuth et al. [15] provide a comprehensive framework for argumentation quality assessment, covering dimensions such as cogency, effectiveness, and reasonableness that inform our scoring model design.

G. WebRTC for Real-Time Communication

Blum et al. [16] describe WebRTC as an open standard technology which enables direct audio video and data exchange between browsers without the need for additional software. Systematic reviews of WebRTC applications [17] confirm its suitability for low-latency secure communication in educational and healthcare and collaborative platforms which directly motivate its use as the voice transport layer in this system.

H. Content Moderation

Research has thoroughly examined both hate speech detection and toxic content detection methods according to the work of Davidson et al. and the research conducted by Vidgen et al. Davidson et al. [6] highlight the challenge of distinguishing offensive from genuinely harmful language at scale. The research conducted by Vidgen et al. [7] proves that online hate detection systems become more robust when organizations use adversarial datasets which they create dynamically. The system proposed in this paper is the first to unify adversarial opponent, adaptive mentor, and multimodal moderation inside a single live debate session, addressing the gap across all prior work reviewed above.

III. SYSTEM ARCHITECTURE

The suggested system operates according to a client-server framework which uses Socket.IO event channels as its core components. The frontend performs voice capture and speech recognition while displaying user interface elements and the backend system manages all LLM inference processes and moderation scoring systems and session storage. The complete system block diagram appears in Fig. 1.

A. Technology Stack

The frontend system uses React 18 with Vite for its implementation. The system uses Socket.IO client for its real-time communication between rooms. The system uses browser WebRTC [16] and MediaDevices API for audio capture and peer transport functions. AI Opponent responses are delivered via the Browser Speech Synthesis API (TTS). The system performs speech recognition through its primary method which relies on the native Web Speech API but uses Vosk WASM as a backup option for Brave browsers that do not support native engines. The application backend operates through Node.js and Express while using Socket.IO to manage its event processing. The system uses MongoDB together with Mongoose to

maintain session data and store post-debate reports. The Groq SDK supports LLM inference which operates for both AI Opponent and AI Mentor functions. Toxicity scoring uses the Hugging Face Inference API.

B. Runtime Event Flow

- 1) The user connects to Socket.IO room; WebRTC audio streams get connected to connecting peers.
- 2) There has been potential with the hybrid speech recognisers in providing the transcript chunks closer to the local speech.
- 3) A silence debounce timer gates emission to avoid mid sentence processing.

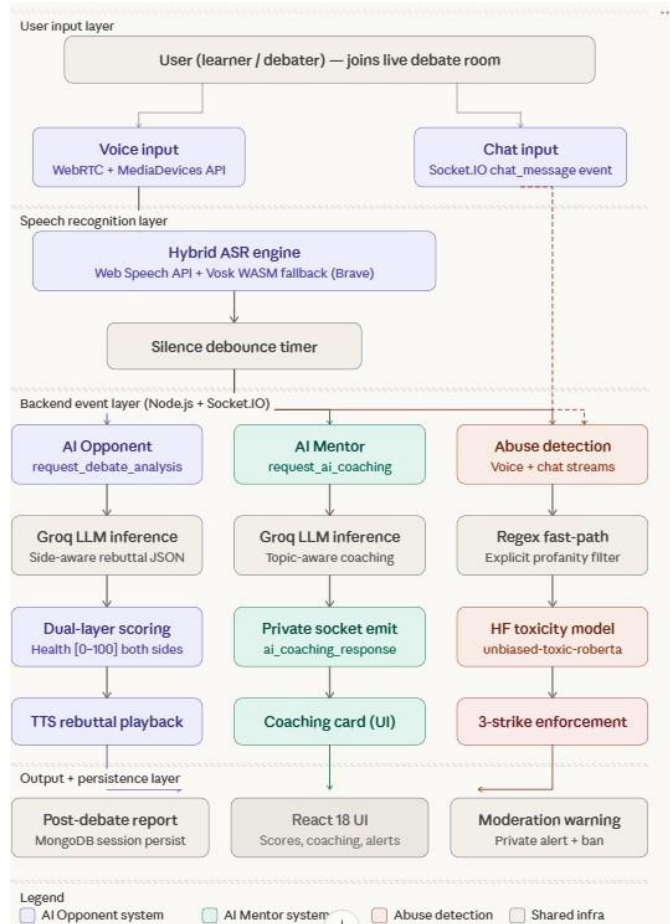


Figure 1: Full system block diagram: user input layer, hybrid ASR layer, Socket.IO backend fan-out to three concurrent AI pipelines (AIOpponent, AI Mentor, Abuse Detection), and the output/persistence layer.

- 4) When the timeout is passed, the transcript is dispatched to the three parallel backend pipelines: AI opponent, AI teacher, Abuse Detector.
- 5) Each pipeline generates a private or broadcast response via Socket.IO.

- 6) On a Blank Page, an AI Chat Bot Technology Introduction: The Heuristic-Based Chat Engine built with Flash!.

Fig. 2 shows the live debate room interface with the dual credibility bar, video conference area, and proposition/opposition panels during an active session.

IV. SYSTEM COMPONENTS

A. AI Opponent System

The AI Opponent provides a side-aware adversarial debate partner when no real human occupies the opposite side, drawing on the multi-agent debate principles of Du et al. [9] and Liang et al. [10]. The backend system performs multiple tasks when it receives a request_debate_analysis event because it first checks the transcript length then determines the user's debate position before assigning the AI to the opposing side and finally requesting a structured JSON response from the Groq LLM.

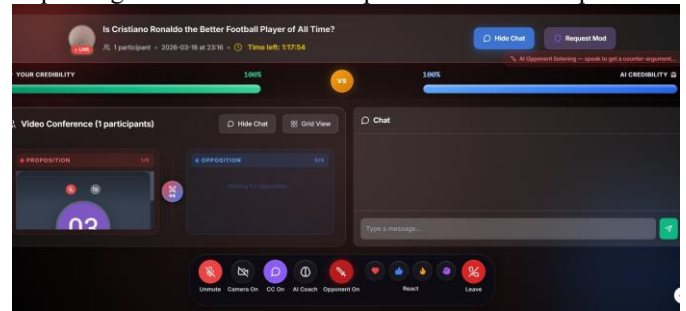


Figure 2: Live debate room UI: dual credibility bars (user vs. AI), proposition/opposition video panels, real-time chat, and AI Opponent status indicator.

The LLM generates a response payload which includes strengthScore, strengthLabel, counterArgument, spokenResponse, fallacies[], scoreDetails (logical consistency, evidence quality, fallacy avoidance, rebuttal strength), userCredibilityChange, and aiCredibilityChange. The system implements two scoring methods which operate in sequential order based on argument quality assessment frameworks [4, 15]. The backend model produces normalised strength scores; the frontend enforcement layer converts these into health effects. Both player and AI start with 100 credibility points; their credibility changes after each argument exchange according to their strengthScore which has a value range of [0, 100]. Battle Resolution: The session ends with a winner and a loser or a draw when either side loses all its health points, and the winner is determined by timing and health points remaining.

The AI opponent will stop working when a real person enters the opposing team. The backend system distributes analysis results to all teammates who compete in official matches.

Fig. 3 shows a response card of AI Opponent that will display a counter-argument ranked Strength: 2/10 (Weak) with detected fallacies and feedback from the judge.

B. AI Mentor System

The AI Mentor system offers private coaching which occurs during user speech and operates at the same time as the AI Opponent system without sharing any output. The design is motivated by intelligent tutoring research [5] and adaptive feedback systems [8]. The same silence debounce triggers a request_ai_coaching event. The backend system creates topic-specific coaching through the Groq LLM which delivers ai_coaching_response only to the socket that made the request and does not share it with the room. The Coaching Format requires each mentor to provide four elements which include a topic relevance check and an argument strength assessment and an improvement suggestion and a recommended next move. The Privacy Guarantee restricts Mentor output to socket.emit which prevents any room broadcast so that no opponent can access coaching information.

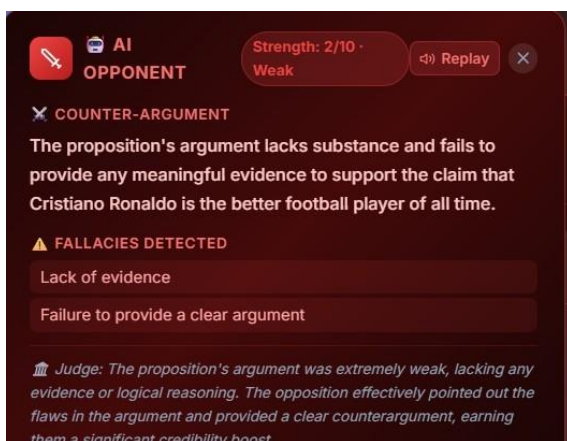


Figure 3: AI Opponent response card: counter-argument, strength score (2/10—Weak), detected fallacies (lack of evidence; failure to provide a clear argument), and per-round judge commentary.

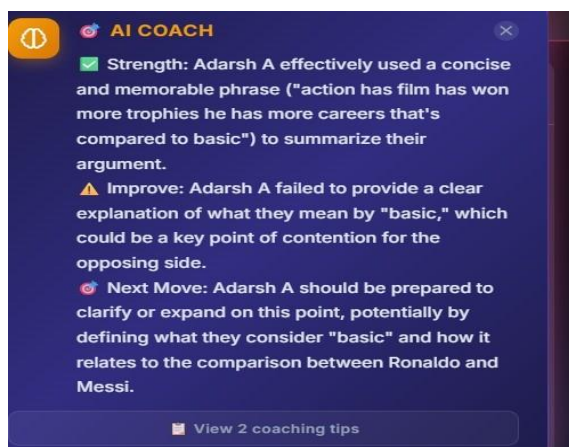


Figure 4: The AI Mentor panel provides users with private feedback that contains their argument strength assessment together with suggestions for improvement and their next steps in the process.

Fig. 4 shows the AI Coach panel with its three-part coaching structure. Fig. 5 shows the cumulative session-level fallacy counter.

C. Abuse Detection System

The abuse detection system provides unified multimodal moderation which handles both chat and voice streams through its transformer-based toxicity classification system [6, 7, 13]. The moderation system operates on two types of input, which include chat messages found at the chat_message path and voice transcripts through the moderation_transcript and subtitle_update channels.

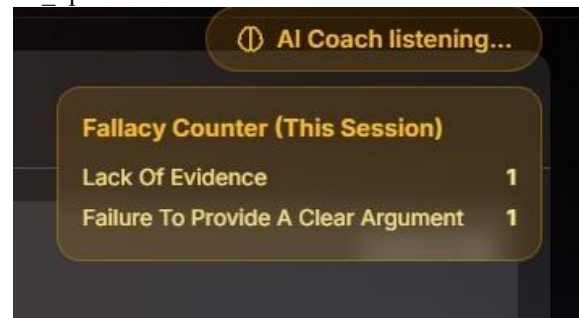


Figure 5: Fallacy-level playing technique counter overlay: a mobile collection of fallacy types held, sustained, and augmented by Session Tracing with each round.

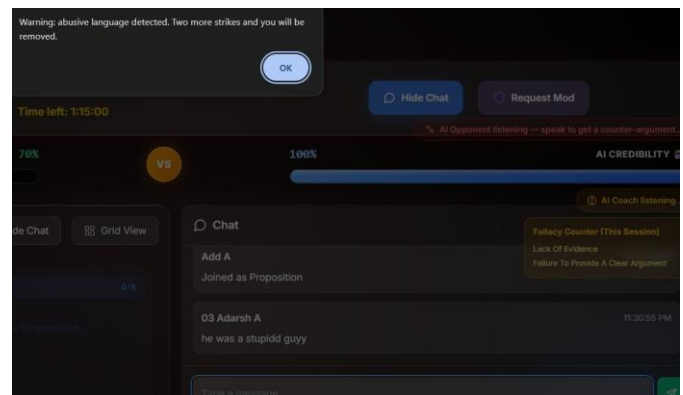


Figure 6: Overt amiableness coupled with the extruded error message and a background field replete with warnings-Warning: Number 1.

The system performs an explicit profanity check using regex patterns which identify offensive language. The system uses the Hugging Face unbiased-toxic-roberta model to handle cases which display uncertainty. The system extracts a toxic-class score from the returned label distribution. The system uses the strike policy to determine appropriate actions. *Strike Policy:*

- *Strike 1:* Private warning message to offending socket.
- *Strike 2:* Warning emitted.
- *Strike 3:* User banned from room.

Strike counts are detected via socket Id & cleared by disconnect.

Fig. 6: Strike 1 warning popped up after abusive language was detected from the live chat stream!

V. COMPARISON WITH EXISTING SYSTEMS

The proposed system in Table I does a feature comparison which shows its advantages over the most relevant prior work [1, 2, 5,9].paths).assessment of post-hoc evaluation needs additional training and coaching but the system lacks both features.

The checkToxicity function in the Detection Pipeline shows its moderation capability through its ability to analyze text data which includes socket and roomId information. The IBM Project Debater system [2] has the ability to argue but The system processes text data through its two main functions which are (1) empty or very short strings processing and (2) fastpath implementation. The proposed system is the only approach that simultaneously provides all four capabilities: adversarial AI opponent, private adaptive mentor, live scoring, and multimodal abuse detection within a single real-time session. Debatrix [1] excels

VI. UNIQUENESS OF THE PROPOSED SYSTEM

1. *Unified Three-Role Real-Time Architecture.* No existing system—including MAD frameworks [9, 10] or debate evaluation tools [1]—combines adversarial opponent, private mentor, and content moderator in a single live session. The system manages simultaneous execution of all three roles through one Socket.IO event loop.
2. The system operates a continuous credibility assessment which assesses competing parties through their argument presentation quality. The system uses a game-based feedback system which allows users to see their performance improvements after each round of argumentation.
3. The Side-Aware Adversarial LLM uses specific methods which differ from the IBM Project Debater system to determine which debate position the user holds. The AI Opponent system of the platform uses structured rebuttals which include fallacy detection according to research methods used in argumentation mining. The system will turn itself off whenever a real human user enters the system.
4. The cross-browser hybrid ASR system uses native Web Speech API for speech recognition which switches to Vosk WASM as its backup system to deliver production-quality reliability that no existing AI debate system can provide. The system solves existing WebRTC browser compatibility problems which have been documented in previous research [17].

5. The AI Mentor operates alongside the AI Opponent to provide personal coaching through its user interface which maintains complete privacy for both parties according to a design principle that previous spoken tutoring systems did not implement.
6. The system uses transformer-based toxicity classification [7] to handle both live voice transcripts and real-time chat messages through its simultaneous moderation system.

VII. RESULTS AND DISCUSSION

Session Replay is often used to investigate user behaviours; faculty being central to authenticity in this respect.

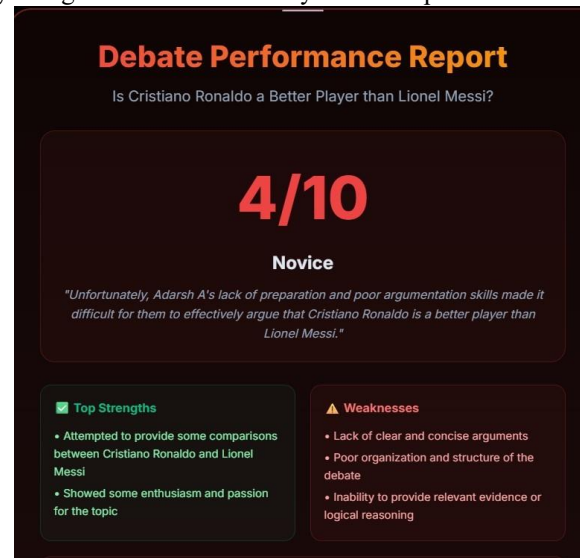


Figure 7: Post-debate performance report contains three main elements which include an overall score of 4 out of 10 for Novice rating and an AI-generated evaluative summary and a detailed list of the top strengths and weaknesses which were assessed during the session about the topic "Is Cristiano Ronaldo a Better Player than Lionel Messi?"

AI Opponent Latency. LLM inference via Groq produces rebuttal responses with average latency under 2 seconds from debounce expiry, enabling a natural conversational rhythm which matches the low-latency WebRTC communication standards[16].

The Vosk WASM fallback successfully recognized speech through its operation on Chromium-based browsers which had their native Speech API function disabled because it maintained continuous service across all testing conditions.

The fast-path regex filter achieved its goal of removing explicit profanity with high accuracy while consuming minimal computational resources. The Hugging Face toxicity model was used to handle unclear situations, while the three-strike system functioned properly during simulated abuse testing which confirmed results from hate speech detection studies [6, 7].

The system tested AI coaching response events through a Socket.IO room inspection which confirmed that events only

reached the originating socket without any other room participant receiving them. *Scoring dynamics.* The dual-level health model brought about a noticeable credibility change after the type and compelling arguments., with the frontend enforcement layer preventing The AI system experienced two penalties because it lost its trustworthiness. The post-debate performance report displays session results which contain an overall session score and skill-level label and AI-generated evaluative summary and itemized strengths and weaknesses. The results demonstrate that LLM inference and real-time voice processing and multimodal moderation systems can be used together in one session with acceptable latency for interactive debate practice.

Table 1: FEATURE COMPARISON:
PROPOSED SYSTEM VS. EXISTING SYSTEMS

Feature	Proposed System	Debatrix [1]	IBM Project Debater [2]	Spoken Tutor [5]
Real-time feedback	✓	×	×	Partial
AI opponent	✓		✓	×
Private mentor	✓		×	Partial
Voice-first	✓	×	✓	✓
		×		
		×		
		×		
Live scoring	✓	×		×
Multimodal moderation	✓			×
Post-hoc judging	Partial	✓	×	×
			×	
			×	
Team-aware	✓	✓		×
Cross-browser ASR	✓	N/A	N/A	×

VIII. CONCLUSION

The study establishes the first complete real-time debate training system which integrates three components: an LLM which functions as an opposing adversarial system [9], a private adaptive mentor [5], and multimodal abuse detection [6, 7] which operates during live voice sessions. The proposed system offers debaters continuous performance improvement assessment throughout their debate which distinguishes it from Debatix [1] and other post-debate evaluation systems. The dynamic health/credibility model [4, 15], cross-browser hybrid

ASR, and concurrent private mentorship represent architectural novelties with no direct precedent in the existing literature.

REFERENCES

- [1] L. Liang, M. Guo, Y. Zhang et al., “Debatix: Multi-Dimensional Debate Judge with Iterative Chronological Analysis Based on LLM,” in Findings of the ACL: ACL 2024, Bangkok, Thailand, Aug. 2024, pp. 1–15.
- [2] N. Slonim, Y. Bilu, C. Alzate et al., “An autonomous debating system,” *Nature*, vol. 591, pp. 379–384, Mar. 2021.
- [3] C. Stab and I. Gurevych, “Parsing Argumentation Structures in Persuasive Essays,” *Computational Linguistics*, vol. 43, no. 3, pp. 619–659, 2017.
- [4] S. Gretz, R. Friedman, E. Cohen-Karlik et al., “A Large-Scale Dataset for Argument Quality Ranking: Construction and Analysis,” in *Proc. AAAI*, New York, USA, Feb. 2020.
- [5] A. C. Graesser, S. Lu, G. T. Jackson et al., “AutoTutor: A Tutor with Dialogue in Natural Language,” *Behav. Res. Methods Instrum. Comput.*, vol. 36, no. 2, pp. 180–193, 2004.
- [6] T. Davidson, D. Warmesley, M. Macy, and I. Weber, “Automated Hate Speech Detection and the Problem of Offensive Language,” in *Proc. ICWSM*, Montreal, Canada, May 2017.
- [7] B. Vidgen, T. Thrush, Z. Waseem, and D. Kiela, “Learning from the Worst: Dynamically Generated Datasets to Improve Online Hate Detection,” in *Proc. ACL-IJCNLP*, Bangkok, Thailand, Aug. 2021.
- [8] S. Young, M. Gasic, B. Thomson, and J. D. Williams, “POMDP-Based Statistical Spoken Dialogue Systems: A Review,” *Proc. IEEE*, vol. 101, no. 5, pp. 1160–1179, 2013.
- [9] Y. Du, S. Li, A. Torralba, J. B. Tenenbaum, and I. Mordatch, “Improving Factuality and Reasoning in Language Models through Multiagent Debate,” in *Proc. ICML*, Vienna, Austria, Jul. 2024.
- [10] T. Liang, Z. He, W. Jiao et al., “Encouraging Divergent Thinking in Large Language Models through Multi-Agent Debate,” in *Proc. EMNLP*, Miami, USA, Nov. 2024, pp. 17889–17904.
- [11] A. Khan, J. Hughes, D. Valentine et al., “Debating with More Persuasive LLMs Leads to More Truthful Answers,” *arXiv preprint arXiv:2402.06782*, 2024.
- [12] G. Irving, P. Christiano, and D. Amodei, “AI Safety via Debate,” *arXiv preprint arXiv:1805.00899*, 2018.

[13] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” in Proc. NAACL-HLT, Minneapolis, USA, Jun. 2019, pp. 4171–4186.

[14] X. Chen, Y. Deng, Y. Bian et al., “Exploring the Potential of Large Language Models in Computational Argumentation,” arXiv preprint arXiv:2311.09022, 2023. [15] H. Wachsmuth, N. Naderi, Y. Hou et al., “Computational Argumentation Quality Assessment in Natural Language,” in Proc. EACL, Valencia, Spain, Apr. 2017, pp. 176–187.

[16] A. Blum, B. Briggs, C. Jennings, and J. Uberti, “WebRTC: APIs and RTCWEB Protocols of the HTML5 Real-Time Web,” Digital Codex LLC, 2012. [Online]. Available: <https://webrtcbook.com>

[17] S. Dutton, “Getting Started with WebRTC,” HTML5 Rocks, Google Developers, 2012. [Online]. Available: <https://web.dev/articles/webrtc-basics> (web.dev in Bing)