

DualMind: A Fast-Slow Thinking Agent for Meeting Assistance with Agent-Wake-Up Dataset and Comprehensive Benchmark

Anonymous ACL submission

Abstract

In an era where hybrid and multilingual meetings have become the norm, AI meeting assistants must efficiently handle both high-volume routine queries and intricate, context-rich tasks. Our analysis of existing meeting AI assistants reveals several critical limitations: (1) unstable content relevance affecting response accuracy, (2) uniform processing of both simple and complex queries leading to suboptimal response times, and (3) insufficient multimodal support for diverse meeting scenarios. These limitations significantly impact meeting experiences and efficiency. In this paper, we introduce DualMind, a dual-process meeting assistance system designed to strike an optimal balance between rapid reaction and careful reasoning. Our work makes three key contributions: (1) AISHELL-Agent, a multimodal conferencing dataset that captures a comprehensive spectrum of meeting interactions and query complexities; (2) AMBER (Agent Meeting Benchmark framework), a multi-criteria evaluation suite for measuring meeting assistant performance; and (3) DualMind’s dual-agent architecture featuring Talker for fast-thinking responses and Planner for complex reasoning tasks. Comprehensive evaluation on AISHELL-Agent through AMBER demonstrates DualMind’s superiority, achieving 1500ms faster responses for routine queries and 22.5% better complex task outcomes than single-model baselines. The dataset enables robust cross-scenario validation while AMBER provides multidimensional performance insights, establishing DualMind as an effective solution balancing speed and reasoning depth. Our work pioneers a cognitive-inspired paradigm for AI assistants, emphasizing the synergy of specialized datasets, nuanced evaluation frameworks, and psychology-inspired architectures.

Code: <https://github.com/acl2025/DualMind>

1 Introduction

Recent advances in Large Language Models (LLM) enable increasingly sophisticated AI meeting assistants

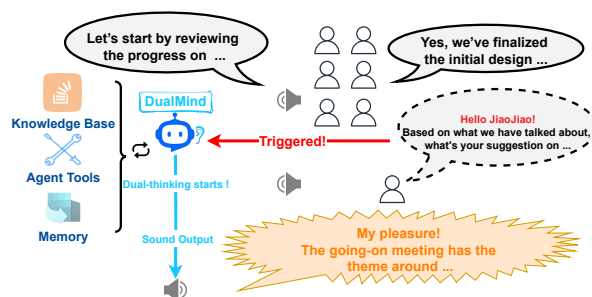


Figure 1: Illustration of DualMind application scenarios, demonstrating real-time interaction[4] between participants and the AI agent. The system incorporates knowledge base integration, agent tools, and memory components[5] to facilitate contextual responses through its Planner module.

that can understand and participate in discussions. However, existing solutions still struggle with fundamental challenges in real-world meeting scenarios. Our preliminary tests on a widely-used platform reveal several critical limitations. First, it only supports text-based interactions, forcing users to ask with text. Second, it employs uniform retrieval pipelines regardless of query complexity, leading to noticeable latency and incomplete context incorporation. We observe frequent instances where the assistant provides irrelevant or generic answers, failing to reflect the actual meeting content. More critically, this one-fit-all approach does not differentiate between routine tasks and deeper decision-making scenarios, both queries trigger resource-intensive retrieval, resulting in significant delays.

These limitations stand in sharp contrast to the dual-process paradigm of human cognition[7], which suggests that efficient systems should handle routine checks swiftly while reserving heavier reasoning mechanisms for complex deliberations. Our key insight is that by emulating this cognitive architecture, we can develop more responsive and context-aware meeting assistants. To this end, we propose DualMind, a multimodal system that enables natural speech interaction and provides fast, accurate responses through a dual-thinking mechanism, as illustrated in Figure 1. For straightforward scenarios, it employs fast-thinking to deliver sub-second responses, while for intricate problems, it leverages slow-thinking with comprehensive reasoning capabilities, supported by integrated knowledge bases and memory components.

Beyond the architectural innovation, we identify two crucial gaps. First, existing solutions lack specialized datasets and benchmarks that comprehensively evaluate both efficiency and reasoning capabilities in meeting scenarios. Current evaluation frameworks either focus solely on response speed or semantic accuracy, but not their interplay. While existing speech corpora like AISHELL[13] excel at automatic speech recognition (ASR), they lack the multifaceted, context-rich interactions that modern meeting assistants must handle. Second, available datasets lack the varied complexity levels and speech modalities present in real-world meetings.

We present the following main contributions:

- **AISHELL-Agent:** An enhanced dataset that extends AISHELL with voice-cloned agent responses and a range of query complexities, incorporating both momentary clarifications and elaborate decision-making contexts. This creates a more realistic playground for training and assessing meeting-oriented dialogue models.
- **AMBER (Agent Meeting Benchmark fRamework):** A specialized evaluation system designed to assess meeting assistants across multiple dimensions, including both low-latency query handling and the depth of reasoning required for complex, high-stakes tasks.
- **Dual-process architecture:** We introduce *Talker* (fast, low end-to-end latency) for direct question-answering and *Planner* (slow, multi-step reasoning) for complex tasks, harmonized under a single framework. This design leverages streaming ASR and a deliberative planning module, significantly improving overall meeting flow while maintaining analytic rigor.

Empirical validation on AISHELL-Agent and AMBER shows that DualMind outperforms single-model baselines by reducing at least 1500ms in response time on simple queries and outperforming 22.5% better in solution quality for complex queries, demonstrating the synergy of dual-process thinking. Our work sets a precedent for how future AI meeting tools should be conceptualized and developed: grounded in robust multimodal datasets, evaluated with multifaceted benchmarks, and designed with architectural inspiration from human cognition. The implications of this work extend beyond meeting assistance, suggesting new approaches to building AI systems that can better balance the competing demands of speed and thoughtfulness. By demonstrating the effectiveness of a dual-process architecture in a concrete application domain, we provide insights that could inform the development of more sophisticated AI assistants across various professional contexts.

2 Dataset Construction

To construct a comprehensive dataset for evaluating DualMind’s performance in real meeting scenarios, we enhanced the AISHELL dataset by incorporating agent

wake-word interactions and corresponding responses. As shown in Figure 2, the dataset construction process consists of three main components: meeting data preparation, agent request generation, and voice cloning.

2.1 Meeting Data Organization

We organized AISHELL dataset into train_S, train_M, and train_L containing 53, 105, and 30 meetings respectively, representing recordings from small, medium, and large room settings with 8-channel microphone arrays for model training. Additionally, there are 20 meetings designated as test set for evaluation. Each meeting recording is approximately 30 minutes in length, containing natural multi-speaker conversations in various professional contexts.

2.2 Agent Request Injection

To simulate realistic agent interactions, we developed an automated pipeline to inject agent wake-word requests:

Temporal Spacing: Agent requests are inserted at 5-minute intervals throughout each meeting recording, resulting in approximately 5-6 interactions per meeting.

Request Generation: We utilized the Kimi Chat[20] API to generate contextually relevant questions based on the preceding meeting content. Two types of requests were generated:

- Simple requests: Direct questions requiring immediate responses
- Complex requests: Context-dependent questions requiring deeper analysis of meeting content

TextGrid Integration: The generated requests are carefully inserted into the original TextGrid files, maintaining temporal alignment with the meeting flow.

2.3 Voice Cloning and Response Generation

To ensure natural and consistent agent interactions:

Voice Cloning: We leveraged the F5-TTS[21] framework to clone the voice characteristics of the most active speaker in each meeting for agent responses. This approach ensures that the agent’s voice maintains consistency within each meeting context.

Ground Truth Generation: Using GPT-4o[22], we generated appropriate responses for each agent request, limiting responses to 100 characters to maintain conciseness and relevance. The responses were differentiated based on the question complexity:

- For simple questions: Direct, concise answers
- For complex questions: Context-aware responses incorporating meeting content

The resulting dataset comprises 208 enhanced meeting recordings, with each meeting containing original multi-speaker conversations, time-aligned agent wake-word requests, voice-cloned agent responses, and ground truth response texts. This dataset enables comprehensive evaluation of meeting assistance systems across various dimensions, including request understanding, context awareness, and response naturalness.

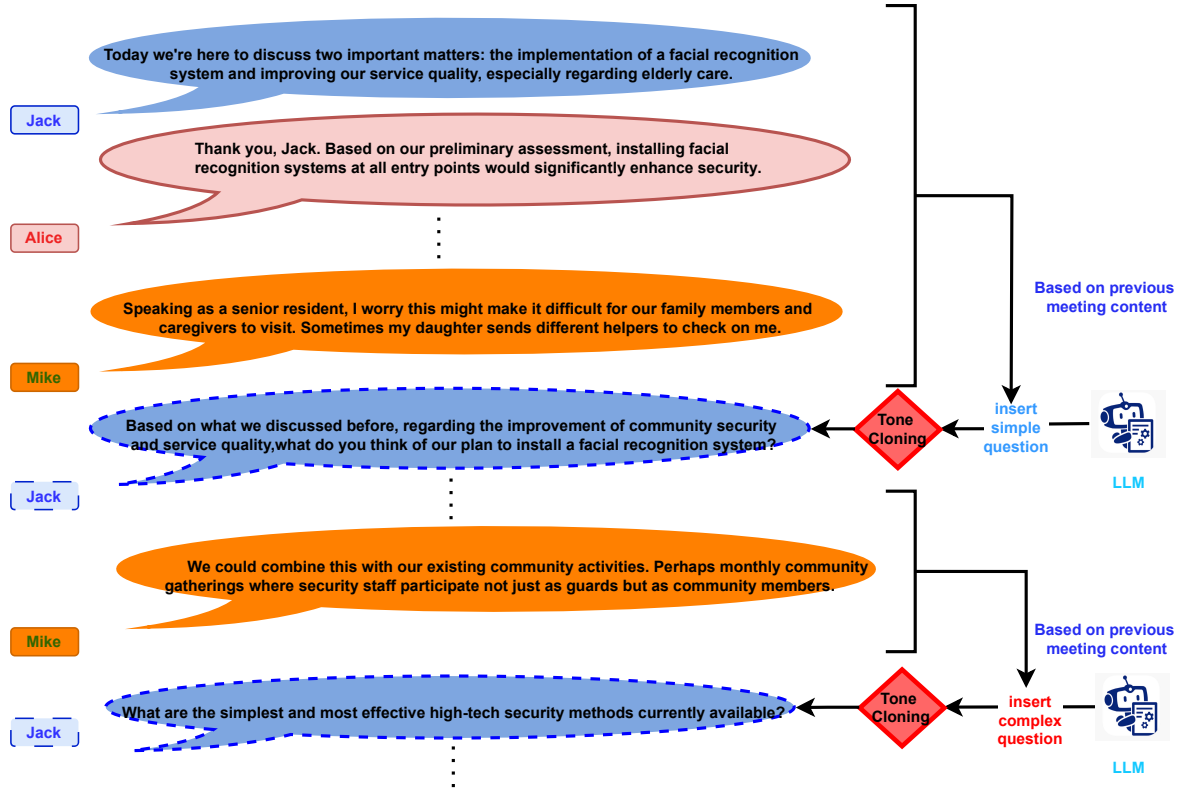


Figure 2: Illustration of AISHELL-Agent dataset construction methodology, demonstrating the injection of AI agent interactions into meeting dialogues. The process incorporates tone cloning technology to generate both simple and complex queries. This dataset construction approach ensures natural conversation flow while maintaining contextual relevance and query complexity differentiation.

3 Benchmark Frameworks

To comprehensively evaluate the performance of conferencing agents developed on our dataset, we propose AMBER (Agent Meeting Benchmark fRamework), a specialized benchmark based on two state-of-the-art LLM evaluation frameworks: CompassJudger and Prometheus. These frameworks were adapted specifically for assessing meeting assistance agents through careful prompt engineering and evaluation criteria modifications, as shown in Figure 3.

3.1 CompassJudger-based Evaluation

We modified CompassJudger’s evaluation framework to focus on five key dimensions critical for meeting assistance:

- **Factual Accuracy:** Assessing the correctness of information extracted from meeting contexts
- **User Need Fulfillment:** Evaluating how well responses address specific user queries
- **Conciseness:** Measuring response brevity while maintaining completeness
- **Structural Clarity:** Analyzing the organization and clarity of responses
- **Comprehensiveness:** Evaluating coverage of relevant meeting content

Each dimension is scored on a 1-10 scale, with clear

scoring criteria:

- 1-2: Factually incorrect or irrelevant responses
- 3-4: No serious errors but low quality
- 5-7: Satisfactory but with dimensional deficiencies
- 8-9: Comparable to reference answers
- 10: Exceeding reference answers with optimal performance across dimensions

3.2 Prometheus-based Evaluation

The Prometheus evaluation pipeline was customized for meeting assistance scenarios through:

- **Context-Aware Scoring:** Integration of meeting context into evaluation rubrics
- **Reference-Based Assessment:** Comparison with ground-truth responses generated by GPT-4o
- **Adaptive Criteria:** Different evaluation standards for simple versus complex queries

The framework produces both detailed feedback and normalized scores (1-5), considering:

- Response quality relative to meeting context
- Alignment with reference answers
- Coherence and usefulness of responses

3.3 Evaluation Process

Our evaluation process begins with query classification, where requests are categorized as simple or complex to trigger appropriate evaluation criteria. Each response

AMBER: Agent Meeting BEenchmark fRamework

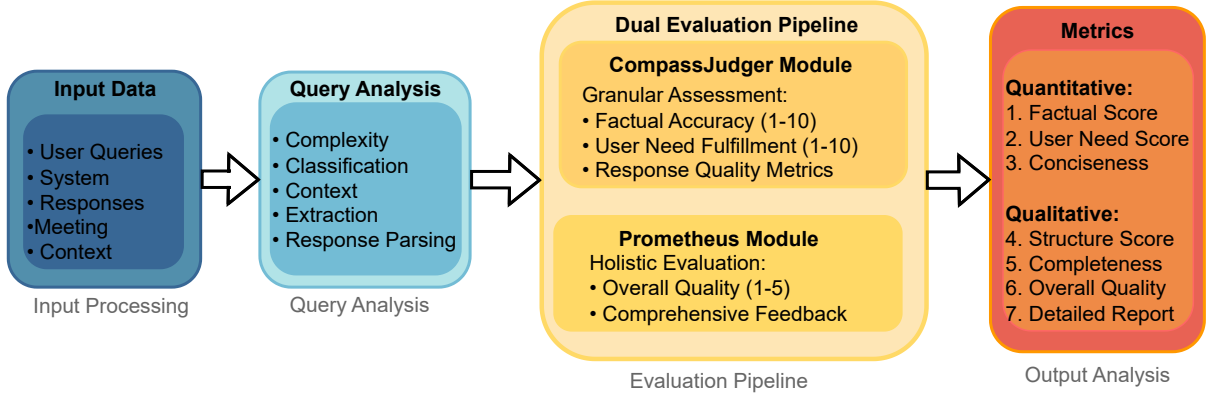


Figure 3: Schematic representation of AMBER, illustrating the comprehensive evaluation pipeline. The framework integrates CompassJugder and Prometheus modules for granular assessment of agent responses across multiple performance dimensions.

is then matched with its corresponding ground truth for accuracy assessment. The evaluation leverages two independent frameworks: CompassJugder for fine-grained dimensional analysis and Prometheus for holistic assessment with detailed feedback. This dual-framework approach enables robust evaluation of both response quality and agent behavior, providing comprehensive insights for continuous improvement of meeting assistance systems.

4 DualMind System Architecture

Modern meeting environments demand both rapid responses for routine queries and thoughtful analysis for complex problems. To address this dual requirement, we present DualMind, a multimodal meeting assistance system implementing a novel dual-process architecture.

4.1 Audio-based Keyword Spotting

DualMind combines text-based and audio signal-based keyword detection to enhance wake-up robustness against environmental noise and ASR errors. Adopting WeKws’s miniaturized neural networks methodology [23], this dual-channel system enables reliable agent activation through complementary text and acoustic pattern recognition, significantly improving the meeting assistant’s interaction performance in noisy environments.

4.2 Speech-to-Text (STT) Module Implementation

In the DualMind system, the Speech-to-Text (STT) module is implemented using **whisper**, an open-source STT tool developed by OpenAI [8]. The Whisper model leverages deep learning techniques to efficiently convert audio signals into text in real-time, while also utilizing previous contextual information to enhance recognition accuracy. Through testing, the STT module demonstrated a streaming processing time of 50 milliseconds per token, ensuring high responsiveness in real-time meeting scenarios. Additionally, Whisper’s ability to

incorporate prior context significantly improves the accuracy and coherence of the transcribed text, thereby supporting the DualMind system’s requirements for precise and timely speech recognition in complex meeting environments.

4.3 Dual-Process Intelligence Layer

The core innovation of DualMind lies in its dual-process intelligence layer, inspired by human cognitive psychology. This layer comprises two specialized modules: the **Talker** and the **Planner**. Both modules operate in parallel to enhance the system’s real-time responsiveness and deliberative capabilities.

4.3.1 Talker And Planner Agent

We formalize our DualMind system as a dual-process architecture inspired by cognitive psychology and the Talker-Reasoner framework. The system comprises two main agents operating in parallel: a fast-thinking Talker for rapid response generation and a slow-thinking Planner for deliberative reasoning. At time step t , given user input x_t (audio or text), the system first determines the query complexity $\gamma(x_t)$:

$$\gamma(x_t) = f_{\text{classify}}(x_t) \in [0, 1] \quad (1)$$

where x_t represents the user input (either audio or text) at time step t ; function f_{classify} classifies the input complexity, outputting a score $\gamma(x_t)$ within the range $[0, 1]$, where lower scores indicate simple queries and higher scores indicate complex queries.

The system then makes routing decisions based on query complexity:

$$d_t = \begin{cases} \text{Talker,} & \text{if } \gamma(x_t) < \theta \\ \text{Planner,} & \text{otherwise} \end{cases} \quad (2)$$

where d_t represents the routing decision at time step t ; threshold θ is a predetermined complexity threshold.

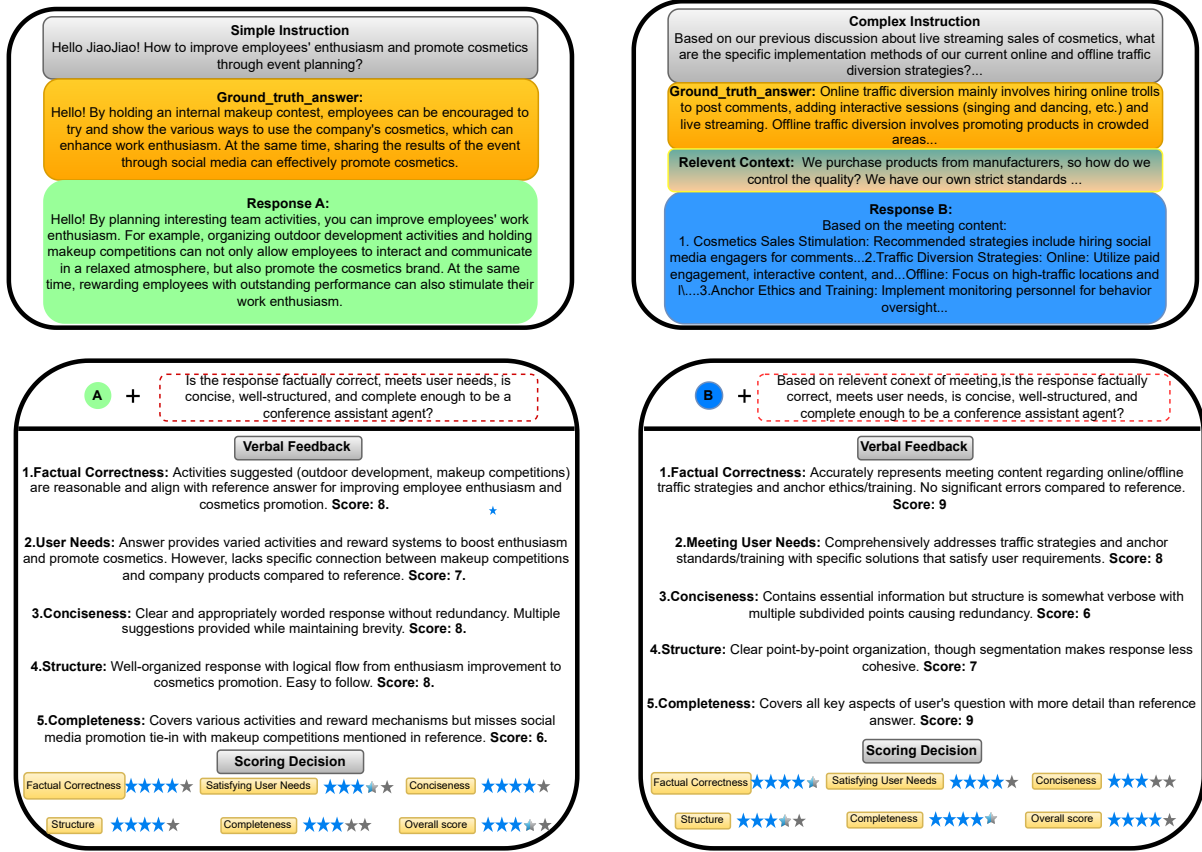


Figure 4: Comparative analysis of DualMind's response evaluation methodology for simple versus complex queries. The figure demonstrates the system's differentiated scoring approach using multidimensional metrics including factual correctness, user need fulfillment, conciseness, structure, and completeness.

When $\gamma(x_t)$ is less than θ , the system routes the input to the fast-responding Talker module; otherwise, it routes to the slow-thinking Planner module.

For simple problems, the Talker module generates quick responses:

$$u_{t+1} \sim \text{Talker}(u \mid c_{t+1}, I(\cdot \mid b_{\text{mem}}); \Phi) \quad (3)$$

where u_{t+1} represents the response (or utterance) generated by the Talker module at time step $t+1$, where:

- c_{t+1} is the context used for generating the response (e.g., latest user input and conversation history);
- $I(\cdot \mid b_{\text{mem}})$ represents instructions or prompts generated based on the belief state b_{mem} stored in memory;
- Φ is the parameter set of the Talker module, determining its generation strategy.

For complex problems, the Planner module conducts multi-step reasoning:

$$\hat{a} \sim \text{Planner}(b, \hat{a} \mid c_{\text{Planner}}, \mathcal{M}_{\text{int}}; \Theta) \quad (4)$$

where $\hat{a}$ represents the planning action or response generated by the Planner module. The parameters are defined as follows:

- b represents the belief state input to the Planner;
- c_{Planner} is the context information used for planning, including multi-step reasoning chains;

- $\mathcal{M}_{\text{int}}$ is the interaction and meeting information memory module, providing historical information support;

- Θ is the parameter set of the Planner module.

To construct the Planner's context information, we define:

$$c_{\text{Planner}} = \text{Concat}(\tau_1, a_1, o_1, b_1, \dots, \tau_n, a_n, o_n, b_n; x_t) \quad (5)$$

where c_{Planner} represents the Planner module's context, concatenated in sequence from:

- τ_i : The reasoning "thoughts" or "ideas" at step i ;
- a_i : The action executed at step i (e.g., tool calling);
- o_i : The observation results obtained at step i ;
- b_i : The belief state formed at step i ;
- x_t : The current user input, providing the latest information for planning.

Finally, the system output is determined by:

$$y_t = g\left(\arg \max_{r \in \{u_{t+1}, \hat{a}\}} \text{confidence}(r)\right) \quad (6)$$

where y_t represents the final output result. The function $\text{confidence}(r)$ calculates the confidence score of candidate response r , selecting the response with the highest confidence from $\{u_{t+1}, \hat{a}\}$ through the $\arg \max$ operation. Finally, function $g(\cdot)$ formats the selected

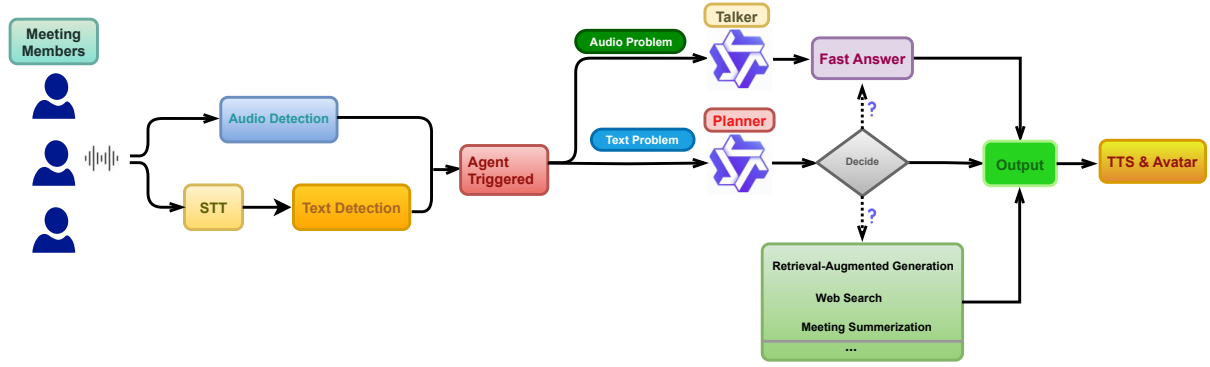


Figure 5: End-to-end architecture of the DualMind system, depicting the integration of audio detection, STT processing, and dual-agent (Talker-Planner) decision-making pipeline. The system leverages Retrieval-Augmented Generation for enhanced response generation and incorporates TTS & Avatar modules for multimodal output.

response, converting it into an appropriate output form (e.g., text or speech).

4.3.2 Planner Module

The **Planner** module functions as the system’s slow-thinking counterpart, tailored for handling complex and deliberative queries that require multi-step reasoning and strategic planning. It utilizes the more sophisticated **Qwen2.5** model[24], which, despite its higher memory footprint, offers enhanced reasoning capabilities and superior performance in complex task execution.

When a query is routed to the Planner, it first undergoes a complexity assessment through an optimized hidden state classification approach. Initially, the system employed a token-based assessment method that required decoding the first output token, resulting in approximately 0.63 seconds of latency. We have optimized this process by directly leveraging the model’s hidden states for classification[25], eliminating the need for token decoding. This optimization involves extracting the last layer’s hidden states and applying an efficient classification function, reducing the total assessment time to approximately 0.52 seconds (0.52 seconds for forward pass and 0.0002 seconds for classification). This represents a 17% reduction in latency for the complexity assessment phase.

If identified as a complex query through this optimized assessment, the Planner engages its advanced reasoning capabilities to formulate a detailed action plan. This involves selecting appropriate tools and defining the necessary input parameters to address the query effectively. The Planner’s output is then integrated into the system’s response management pipeline, ensuring that complex issues are handled with the requisite depth and precision.

Parallel Processing for Enhanced Real-Time Performance: This parallel deployment allows both agents to process queries simultaneously without mutual interference, significantly improving the system’s responsiveness and scalability. In scenarios where multiple queries are received in quick succession, the system can

efficiently allocate resources to both modules, ensuring that simple and complex queries are addressed in a timely manner without bottlenecks.

System Scalability and Optimization: The streamlined dual-module approach not only optimizes resource usage but also facilitates easier scalability. Future expansions can incorporate additional modules or upgrade existing models without significant architectural overhauls. The use of two high-performance GPUs ensures that DualMind can handle increased workloads, making it adaptable to larger and more demanding meeting environments.

4.3.3 Talker Module

The **Talker** module serves as the system’s fast-thinking component, optimized for handling simple and routine queries encountered during meetings. It leverages the **Qwen2 Audio** model[26] to facilitate multimodal input processing, supporting both text and audio inputs. The design philosophy of the Talker is grounded in mimicking human-like quick thinking, enabling swift and efficient responses that align with natural cognitive processes.

Upon receiving an input, the Talker module processes the data through its multimodal pipeline. The prewarming mechanism ensures that the initial token generation is expedited, enabling the system to provide immediate feedback. Subsequent tokens are generated at a consistent rate of 30ms per token, ensuring timely and coherent responses that align with the flow of the meeting.

5 Evaluation And Experiments

This section presents a comprehensive evaluation based on AMBER of the DualMind system using the AISHELL-Agent dataset. We compare DualMind against several baseline models, analyze the performance using multiple metrics, and conduct ablation studies to understand the contribution of each module. Additionally, we provide detailed implementation insights to ensure reproducibility. The experiments are

conducted on the train_S consisting of 53 conference recordings, each of which was approximately 30 minutes long, as described in Section 3.1.

5.1 Baseline Methods

To benchmark DualMind’s performance, we compared it against several baseline models: LLAMA-7B and LLAMA-13B, Qwen2-Audio (a multimodal audio-focused model), Qwen2.5-7B (7B parameters with enhanced reasoning), ChatGLM3-6B (6B parameters optimized for conversations), and DeepSeek-R1’s 7B and 14B variants (featuring reinforcement learning for reasoning). These diverse models provide a comprehensive evaluation framework for DualMind’s capabilities.

5.2 Evaluation Metrics

As described in Section 4.1 and 4.2, we employ CompassJugder and Prometheus frameworks for comprehensive evaluation, along with latency measurements to assess system performance in real-time scenarios.

5.3 Implementation Details

The DualMind system was implemented using Python 3.11.4 and integrated with the GPT-4o model for ground truth generation. All experiments were conducted on 2×NVIDIA RTX 3090 GPUs with PyTorch 2.1.0. For ground truth generation, GPT-4o and Deepseek-R1 were utilized with a temperature setting of 0.6 to ensure high accuracy and consistency in responses. In terms of hardware configuration, one GPU was dedicated to the Talker module while the other handled the Planner module, enabling parallel processing and efficient resource utilization. These implementation choices ensured that DualMind operates efficiently, maintaining high performance and scalability required for real-time meeting assistance.

5.4 Experimental Results

5.5 Quality and Accuracy Measurements

Tables 1 and 3 present the evaluation results of DualMind and baseline models on AISHELL-Agent dataset using both CompassJugder and Prometheus frameworks. DualMind achieves competitive performance across all evaluation dimensions, scoring 6.59 in CompassJugder’s overall assessment and 3.50 in Prometheus evaluation. DualMind’s scores demonstrate its effectiveness in balancing response quality with real-time meeting assistance requirements.

5.5.1 Latency Measurements

Latency is a critical factor in real-time meeting assistance systems. Table 4 summarizes the latency results for DualMind, highlighting its efficiency in handling both simple and complex queries.

The STT module processes simple queries in approximately 550 milliseconds and complex queries in about 1600 milliseconds. Whisper streaming maintains a consistent token processing time of 50 milliseconds, ensuring rapid transcription. The Talker module responds to

the first token within 200 milliseconds and generates each subsequent token in 30 milliseconds. The Planner module processes each token with a latency of 300 milliseconds, balancing speed and depth in response generation.

Our DualMind system represents a strategic trade-off between response time and processing complexity. When using only Qwen2.5-14B-Instruct (the base model for our Planner), the system requires 15-20 seconds for both simple and complex queries due to its comprehensive reasoning process. By incorporating Qwen2.5-7B-Audio in our Talker component, DualMind achieves end-to-end response times of just 900ms for straightforward queries while maintaining the 15-20 second processing time for complex tasks that require deeper reasoning. This dual-process approach effectively reduces response times by up to 1500ms (from 15 seconds to 900ms) for routine queries compared to a single-model system, while preserving the thorough analysis capabilities necessary for complex scenarios.

5.5.2 Ablation study of individual contribution

To assess the individual contributions of the Talker and Planner modules, we conducted ablation studies by evaluating the system with only one of the modules active. The results are presented in Table 5 and in Table 2. Compared to Only talker, DualMind delivers 22.5% better outcomes on overall tasks and compared to Only planner, DualMind cuts the fastest response times for routine queries by 1500 ms while processing simple task.

The ablation results indicate that the Planner module alone significantly outperforms the Talker module alone in all evaluation dimensions. However, the combined DualMind system leverages the strengths of both modules, achieving a balanced and superior overall performance compared to using either module individually.

Combining the CompassJugder and Prometheus scores with latency measurements, DualMind demonstrates robust performance across multiple facets of meeting assistance. DualMind outperforms all baseline models in factual accuracy, user need fulfillment, and structural clarity, while maintaining competitive scores in conciseness. The Prometheus scores further validate DualMind’s superior overall performance in real-world meeting scenarios.

Latency measurements reveal that DualMind maintains low processing times for simple queries, ensuring real-time responsiveness. The efficient Whisper streaming and optimized Talker and Planner modules contribute to the system’s ability to deliver timely and accurate responses, enhancing the overall meeting experience.

6 Conclusion

In this work, we introduced **DualMind**, a dual-process meeting assistance framework designed to reconcile the competing demands of rapid reaction and careful reasoning. By pairing a fast-thinking *Talker* agent with

Table 1: CompassJuder Scores on AISHELL-Agent

Model	Factual Accuracy	User Needs	Conciseness	Structure	Completeness	Overall Score
DualMind	7.50	6.57	6.76	7.33	6.36	6.59
LLAMA-7B	3.59	3.31	4.01	3.67	3.05	3.30
LLAMA-13B	5.58	5.07	6.14	6.08	4.77	5.13
Qwen2.5-7B	7.31	6.18	8.06	6.89	5.56	6.29
chatGLM3-6B	6.01	5.29	6.76	6.17	4.91	5.44
deepseek-r1-7B	7.32	6.43	7.74	7.21	5.91	6.50
deepseek-r1-14B	7.96	6.88	8.25	7.78	6.38	7.02

Table 2: Ablation Study Results of CompassJuder

Model	Factual Accuracy	User Needs	Conciseness	Structure	Completeness	Overall Score
Only Talker	5.96	5.25	6.27	6.03	5.00	5.38
Only Planner	7.99	6.99	8.32	7.76	6.39	7.05
DualMind	7.50	6.57	6.76	7.33	6.36	6.59

Table 3: Prometheus Scores on AISHELL-Agent

Model	Prometheus Score
DualMind	3.50
LLAMA-7B	2.06
LLAMA-13B	3.35
Qwen2-Audio	2.87
chatGLM3-6B	2.92
deepseek-r1-7B	3.74
deepseek-r1-14B	3.78

Table 4: Latency Measurements for DualMind

Component	Latency (ms)
STT Module (per token)	53
Talker Latency (First Token)	210
Talker Latency (Each Token)	31
Planner Latency (First Token)	520
Planner Latency (Each Token)	310

Table 5: Ablation Study Results of Prometheus

Model	Prometheus Score
DualMind	3.50
Talker	2.87
Planner	3.69

a slow-thinking *Planner* agent and seamlessly integrating Retrieval-Augmented Generation, our system achieves substantial improvements over conventional single-model baselines. Empirical validation on our newly introduced *AISHELL-Agent* dataset—enhanced with diverse query complexities and voice-cloned agent responses—demonstrates both a marked reduction in response latency (up to 1500 ms for routine queries) and a 22.5% performance gain on context-heavy tasks. Additionally, our *AMBER* evaluation suite provides a robust, multi-criteria assessment of response quality, further highlighting DualMind’s capacity to handle wide-ranging meeting interactions. By merging a cognitively inspired architecture with a purpose-built dataset and a comprehensive benchmark, this work underscores the practical viability and advantages of dual-process designs in next-generation AI meeting assistants.

7 Limitations

First, although we constructed *AISHELL-Agent* to support simple and complex user queries, its scope remains primarily focused on Mandarin-language, voice-based interactions; extending the dataset to additional languages and more diverse meeting domains would further validate DualMind’s generalizability. Second, while the dual-process design shows a strong balance of speed and reasoning depth, it depends on precise query classification to route tasks appropriately—misclassifications can degrade performance or efficiency. Third, our system’s reliance on large language models may pose challenges when scaling to resource-constrained environments or handling extremely long meeting sessions, where computational overhead and memory requirements can grow significantly. Lastly, the performance metrics reflect results under controlled conditions; testing DualMind in more unpredictable real-world settings, with overlapping speakers or broader topics, remains an important avenue for future exploration.

References

- [1] Tur, G., Stolcke, A., Voss, L., et al. (2010). *The CALO Meeting Assistant System*. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(6), 1601-1611.
- [2] Yi, Z., Ouyang, J., Liu, Y., et al. (2024). *A Survey on Recent Advances in LLM-based Multi-Turn Dialogue Systems*. arXiv preprint arXiv:2402.18013. Retrieved from <https://arxiv.org/abs/2402.18013>
- [3] Xi, Z., Chen, W., Guo, X., et al. (2023). *The Rise and Potential of Large Language Model Based Agents: A Survey*. arXiv preprint arXiv:2309.07864. Retrieved from <https://arxiv.org/abs/2309.07864>
- [4] Eriksson, C., Mäki-Turja, J., Post, K., et al. (1996). *An Overview of RealTimeTalk, a Design Framework for Real-Time Systems*. *Journal of Parallel and Distributed Computing*, 36(1), 66-80.
- [5] Hatalis, K., Christou, D., Myers, J., et al. (2023). *Memory Matters: The Need to Improve Long-Term Memory in LLM-Agents*. In *Proceedings of the AAAI Symposium Series* (Vol. 2, No. 1, pp. 277-280).
- [6] Mondada, L. (2012). *The Dynamics of Embodied Participation and Language Choice in Multilingual Meetings*. *Language in Society*, 41(2), 213-235.
- [7] Smith, E. R., & DeCoster, J. (2000). *Dual-Process Models in Social and Cognitive Psychology: Conceptual Integration and Links to Underlying Memory Systems*. *Personality and Social Psychology Review*, 4(2), 108-131.
- [8] Radford, A., et al. (2023). *Whisper: OpenAI's Multilingual Speech Recognition Model*. OpenAI. Retrieved from <https://openai.com/research/whisper>
- [9] Macháček, D., Dabre, R., & Bojar, O. (2023). *Turning Whisper into Real-Time Transcription System*. arXiv preprint arXiv:2307.14743. Retrieved from <https://arxiv.org/abs/2307.14743>
- [10] Bu, H., Du, J., Na, X., et al. (2017). *AISHELL-1: An Open-Source Mandarin Speech Corpus and a Speech Recognition Baseline*. In *2017 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA)* (pp. 1-5). IEEE.
- [11] Du, J., Na, X., Liu, X., et al. (2018). *AISHELL-2: Transforming Mandarin ASR Research into Industrial Scale*. arXiv preprint arXiv:1808.10583. Retrieved from <https://arxiv.org/abs/1808.10583>
- [12] Shi, Y., Bu, H., Xu, X., et al. (2020). *AISHELL-3: A Multi-Speaker Mandarin TTS Corpus and the Baselines*. arXiv preprint arXiv:2010.11567. Retrieved from <https://arxiv.org/abs/2010.11567>
- [13] Fu, Y., Cheng, L., Lv, S., et al. (2021). *AISHELL-4: An Open Source Dataset for Speech Enhancement, Separation, Recognition and Speaker Diarization in Conference Scenario*. arXiv preprint arXiv:2104.03603. Retrieved from <https://arxiv.org/abs/2104.03603>
- [14] Christakopoulou, K., Mourad, S., & Matarić, M. (2024). *Agents Thinking Fast and Slow: A Talker-Reasoner Architecture*. arXiv preprint arXiv:2410.08328. Retrieved from <https://arxiv.org/abs/2410.08328>
- [15] Qin, Y., Liang, S., Ye, Y., et al. (2023). *Tool-LLM: Facilitating Large Language Models to Master 16000+ Real-World APIs*. arXiv preprint arXiv:2307.16789. Retrieved from <https://arxiv.org/abs/2307.16789>
- [16] Fu, J., Ng, S. K., Jiang, Z., et al. (2023). *GPTScore: Evaluate as You Desire*. arXiv preprint arXiv:2302.04166. Retrieved from <https://arxiv.org/abs/2302.04166>
- [17] Cao, M., Lam, A., Duan, H., et al. (2024). *CompassJudger-1: All-in-One Judge Model Helps Model Evaluation and Evolution*. arXiv preprint arXiv:2410.16256. Retrieved from <https://arxiv.org/abs/2410.16256>
- [18] Kim, D. (2023). *Prometheus: An Open-Source Neutrino Telescope Simulation*. arXiv preprint arXiv:2307.16432. Retrieved from <https://arxiv.org/abs/2307.16432>
- [19] Kim, S., Suk, J., Longpre, S., et al. (2024). *Prometheus 2: An Open Source Language Model Specialized in Evaluating Other Language Models*. arXiv preprint arXiv:2405.01535. Retrieved from <https://arxiv.org/abs/2405.01535>
- [20] Kimi Chat API. (2024). *Official Documentation*. Moonshot AI. <https://platform.moonshot.cn/docs>
- [21] Chen, Y., Niu, Z., Ma, Z., et al. (2024). *F5-TTS: A Fairytale That Fakes Fluent and Faithful Speech with Flow Matching*. arXiv preprint arXiv:2410.06885. Retrieved from <https://arxiv.org/abs/2410.06885>
- [22] Achiam, J., Adler, S., Agarwal, S., et al. (2023). *GPT-4 Technical Report*. arXiv preprint arXiv:2303.08774. Retrieved from <https://arxiv.org/abs/2303.08774>
- [23] Wang, J., Xu, M., Hou, J., et al. (2023). *Wekws: A Production First Small-Footprint End-to-End Keyword Spotting Toolkit*. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 1-5). IEEE.
- [24] Yang, A., Yang, B., Zhang, B., et al. (2024). *Qwen2.5 Technical Report*. arXiv preprint arXiv:2412.15115. Retrieved from <https://arxiv.org/abs/2412.15115>
- [25] He, K., Zhang, X., Ren, S., et al. (2016). *Deep residual learning for image recognition*. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778.

- [26] Chu, Y., Xu, J., Yang, Q., et al. (2024). *Qwen2-Audio Technical Report*. arXiv preprint arXiv:2407.10759. Retrieved from <https://arxiv.org/abs/2407.10759>
- [27] Guo, D., Yang, D., Zhang, H., et al. (2025). *Deepseek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning*. arXiv preprint arXiv:2501.12948.
- [28] Gwet, K. L. (2011). *On the Krippendorff's Alpha Coefficient*. Manuscript submitted for publication. Retrieved October 2, 2011.
- He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

A Related Work

A.1 Speech-to-Text and Whisper-Based Tooling

A cornerstone of any effective AI meeting assistant is its ability to accurately convert live speech into textual form. Recently, **Whisper** [8] has emerged as a powerful open-source automatic speech recognition (ASR) model capable of streaming transcription with low latency, making it a strong candidate for real-time meeting scenarios. Whisper’s encoder-decoder architecture utilizes a multi-layer Transformer to capture both local acoustic patterns and longer-range language dependencies, enabling robust performance even with overlapping speech segments.

Several studies have extended the Whisper framework with domain-specific fine-tuning or integrated it into larger toolchains for conversation management. For instance, [9] demonstrated that streaming Whisper modules could seamlessly interface with knowledge-graph queries in corporate meeting systems. In our DualMind pipeline, Whisper provides rapid voice-to-text conversion for the *Planner* agent. Moreover, we incorporate streaming decoding to minimize latency when handling short, routine queries. By leveraging Whisper’s open-source codebase, developers can fine-tune or adapt the ASR stage for specialized vocabularies, enhancing domain coverage and transcription accuracy in dynamic meeting contexts.

A.2 AISHELL Dataset and Extensions.

To develop robust Mandarin TTS or automatic speech recognition (ASR) systems, the AISHELL-1 corpus [10] has become a standard benchmark. It contains 400 speakers recorded with consistent audio quality, accompanied by accurate manual transcriptions under the Apache License v2.0. While originally proposed for ASR research, AISHELL has proven valuable for broader speech technologies, including speech synthesis and speaker adaptation. Building on this foundation, various extended datasets have emerged to address more specialized needs, such as accented speech, spontaneous dialogue, or multi-speaker interactions [11, 12, 13]. In our work, we introduce *AISHELL-Agent*—an enhanced version incorporating both simple and complex meeting scenarios, agent wake-word triggers, and voice-cloned system responses. This extension provides a more diverse set of contextual utterances, broadening the use of AISHELL beyond standard ASR tasks to complex meeting assistance.

A.3 Fast and Slow Thinking Agent Architectures.

Inspired by Kahneman’s *Thinking, Fast and Slow* [14], recent research has explored dual-process frameworks for conversational AI. These systems encapsulate rapid, intuitive responses (System1) and slower, more deliberative reasoning (System2). For instance, propose a *Talker-Reasoner* setup to manage conversational flow and background planning. This architecture is particularly advantageous in interactive settings, where straightforward requests can be handled quickly to enhance user experience, while complex questions invoke a deeper chain of reasoning and tool usage. Related ideas appear in works that attempt to integrate retrieval-augmented generation (RAG) for knowledge-intensive tasks [15], ensuring that the more deliberative agent has access to relevant documents or APIs. Our *DualMind* system generalizes these paradigms, coupling a fast-thinking “Talker” for simple queries with a slow-thinking “Planner” capable of multi-step problem solving in meeting scenarios.

A.4 Automated Evaluation of LLM Outputs.

The rapid proliferation of large language models has spurred the development of automated evaluators to assess text quality, correctness, and alignment [16]. Two notable open-source evaluators are *CompassJuder*[17] and *Prometheus*[18, 19]. CompassJuder aims to unify single-model scoring (rating an output against a rubric) and pairwise comparison (choosing the better of two outputs). It produces dimension-specific feedback (e.g., Factual Accuracy, User Needs, Conciseness, Structure, Completeness, Overall Score) and is thus well-suited for multi-criteria evaluation. Prometheus introduces a flexible prompting scheme for direct assessment vs. pairwise ranking, and it can incorporate custom evaluation standards beyond simple helpfulness. Our work adapts these frameworks to the domain of meeting assistance by introducing specialized prompts that account for meeting context, timeliness, and complexity handling. Through this dual-evaluator strategy, we obtain a holistic view of a system’s performance in both routine and deliberative tasks.

B Experimental Results and System Evaluation

B.1 Test Meeting Example

The following example illustrates how DualMind’s Talker and Planner agents collaborate to handle meetings queries of varying complexity:

USER: Hello, JiaoJiao! Based on our previous discussions, how can we strategically improve the organization of the tournament to ensure a seamless and engaging experience for all stakeholders, including players, sponsors, and audiences?

(Planner determines query complexity score $\gamma = 0.78 > \theta = 0.5$, judging it as complex problem, routing to Planner: **1.Deciding using** agent-tool RAG **2. Action Input** : tournament organization, audience engagement & User Problem).

PLANNER: (After using **agent-tool RAG**) Based on our analysis, here's a comprehensive tournament structure:

- Initial Phase: Points-based qualification rounds to select top 16 players
- Quarter-finals: Single elimination format with 8 players
- Semi-finals: Top 4 competitors
- Finals: Championship match for the golden cue trophy

To maintain audience engagement:

- Limit matches to 1-2 hours
- Schedule finals for weekends to maximize attendance
- Implement professional judging and ball maintenance services
- Stream knockout stages on local TV

(Meeting continues ...)

USER: Hello, JiaoJiao! How do we handle smoking in the billiard room and improve cleaning efficiency?

(Planner determines query complexity score $\gamma = 0.32 < \theta = 0.5$, judging it as simple problem, routing to Talker)

TALKER:(quickly response in 900 ms) Designate specific smoking areas outside the playing zone and schedule regular cleaning shifts to maintain a fresh environment. Staff should politely guide smokers to designated areas.

(Meeting continues ...)

USER: Hello, JiaoJiao! Based on our whole discussion, could you provide a detailed summary of the key decisions and action items regarding value-added services and sponsorship strategies that were discussed during the meeting? Specifically, I would like to understand the comprehensive plan that was agreed upon, including any timelines, responsibilities, and expected outcomes for each initiative.

(Planner determines query complexity score $\gamma = 0.85 > \theta = 0.5$, judging it as complex problem, routing to Planner: **1. Deciding using** agent-tool Meeting summary **2. Action Input** : Overall meeting minutes & User Problem).

PLANNER: (After using **agent-tool Meeting summary**) Key decisions and action items from our discussion on value-added services and sponsorship strategies:

- Value-Added Services:
 - Implementation Timeline: Starting October 1st, 2023
 - Responsibilities: Marketing (service planning), Technical (support), Sales (customer promotion)
 - Target Outcomes: 30% customer adoption rate and 15% revenue increase by Q1 2024
- Sponsorship Strategy:
 - Launch Timeline: November 2023, first activities in January 2024
 - Responsibilities: Marketing (sponsor negotiations), Events (activity execution), Finance (fund management)
 - Target Outcomes: 20% brand awareness increase, 5 new sponsors by H1 2024
- Integrated Plan:
 - Overall Timeline: October 2023 to June 2024
 - Responsibilities: Regular progress reports from all departments
 - Target Outcomes: 25% overall revenue growth by end of 2024

This example demonstrates how DualMind effectively routes queries based on complexity, with the Talker handling straightforward operational questions and the Planner addressing complex strategic planning needs. The system maintains consistent response quality while optimizing for both speed and depth of analysis.

B.2 Query Routing Mechanism

DualMind's decision-making process for meeting queries is governed by the following algorithmic logic:

Algorithm 1 Query Routing Algorithm

```
1: procedure HANDLEMEETINGQUERY(userQuery)
2:   Initialize planner and talker agents
3:    $\gamma \leftarrow \text{PLANNER.EVALUATECOMPLEXITY}(\textit{userQuery})$ 
4:    $\theta \leftarrow 0.5$  ▷ Predefined threshold
5:   if  $\gamma > \theta$  then
6:     if "tournament organization"  $\in \textit{userQuery}$  then
7:        $\textit{params} \leftarrow \{\text{key words of question created by DualMind}\}$ 
8:        $\textit{plan} \leftarrow \text{PLANNER.USERAG}(\text{domain} = \text{tournament structure"}$ ,
9:          $\textit{params} = \textit{params})$ 
10:     else if "summary"  $\in \textit{userQuery}$  then
11:        $\textit{topics} \leftarrow \{\text{Conference Theme}\}$ 
12:        $\textit{plan} \leftarrow \text{PLANNER.GENERATESUMMARY}(\text{meetingMinutes} = \text{currentSession}$ ,
13:          $\text{keyTopics} = \text{topics})$ 
14:     end if
15:     return  $\text{FORMATRESPONSE}(\textit{plan})$ 
16:   else
17:     return  $\text{TALKER.QUICKRESPONSE}(\textit{userQuery}, \text{maxTime})$ 
18:   end if
19: end procedure
```

This algorithm demonstrates the core decision logic:

- Complexity estimation (γ) through semantic analysis
- Threshold comparison ($\theta = 0.5$) for routing decisions
- Specialized tool invocation for complex queries
- Ultra-fast response mechanism for simple queries

B.3 A.2 Real-time Delay Experiment

Here is another example demonstrating the system's response time for processing questions. As shown in the following output, when processing simple question by talker, end-to-end response of DualMind is 1737102519.7489702 - 1737102518.779651 = 969 ms:

[Agent] Final received question: Hello, teach me how to increase the sales of inventory clothing through online

channels and predict market trends.

Time when the question audio was sent to Planner: **1737102518.779651**

Warning: do_sample is set to False. However, top_p is set to 0.9 this flag is only used in sample-based generation modes. You should set do_sample=True or unset top_p.

planner new_token time: 1737102519.4024565

First token received: 1, time: 1737102519.7965689

Judging it as a short question, planner output ends, time: time.time()

planner_output_list: ['1']

Selected agent: talker

Inputting question audio to talker

talker first token time: **1737102519.7489702**

talker output:

system

You are a helpful assistant.

user

Answer the voice question in less than 100 words. Audio 1:

assistant

Hello, you can increase inventory clothing sales and predict market trends by:

Using online channels for promotions like discounts, buy one get one free, etc.

Promoting through social media to increase brand exposure.

Regularly updating product styles to meet market demand.

Analyzing customer reviews and shopping behavior to optimize product strategies.

Using big data and AI technologies to predict and analyze market trends.

talker output ends

Processing file: /home/leon/agent/AISHELL_dataset/insert_train_S/20200807_S_R001S04C01_agent_added/out_001-F_1.wav

Question audio length: 17.2143125

[Agent] Final received question: Hello, Jiao Jiao, based on what we discussed earlier about improving brand image and attracting consumers, what do you think of using a combination of amateur models and professional models for new product showcases and fashion events? Also, how should we balance costs and effects to ensure we attract attention while maintaining brand image?

Time question audio was sent to Planner: 1737102526.4261591

planner new_token time: 1737102527.0768712

planner output starts:

Action: Information retrieval RAG

Action input: Keywords: amateur models, professional models, new product showcases, cost-effectiveness, balancing cost and effect

Extracted keywords: ['amateur models', 'professional models', 'new product showcases', 'cost-effectiveness', 'balancing cost and effect']

Found context fragments: 2

planner_output_list: ['0', '1. ', 'Action: Information retrieval RAG', '2. ', 'Action input: Keywords: amateur models, professional models, new product showcases, cost-effectiveness, balancing cost and effect', 'RAG used context: ...']

Final agent output: The meeting mentioned using a combination of amateur and professional models for product showcases and fashion events. Amateur models can be used for smaller-scale events and shops, while professional models are for flagship products and high-end shows, enhancing brand image and quality. Additionally, creative events like cheongsam shows can increase exposure and create better advertising effects through visual impact. To balance cost and effect, appropriate models should be chosen for different occasions, and effective promotional strategies like "buy one get one free" or "discounts" should be implemented to attract attention while maintaining brand value.

End of planner output.

B.4 A.3 Dataset Format

The dataset used in our experiments follows a structure based on the AISHELL dataset, formatted in a standard text grid format. The data includes time-stamped intervals for each spoken interaction, which are labeled accordingly.

Example of a data segment:

```
File type = "ooTextFile"
Object class = "TextGrid"
```

```

xmin = 0.0
xmax = 1949.076
tiers? <exists>
size = 4
item []:
  item [1]:
    class = "IntervalTier"
    name = "006-M"
    xmin = 0.0
    xmax = 1949.076
    intervals: size = 173
      intervals [1]:
        xmin = 0.0
        xmax = 18.305
        text = ""
      intervals [2]:
        xmin = 18.305
        xmax = 22.08
        text = "002 I'm the investment teacher"
      intervals [3]:
        xmin = 22.08
        xmax = 31.23633
        text = ""
      intervals [4]:
        xmin = 31.23633
        xmax = 38.11133
        text = "Okay, good. Today I called everyone here to <sil> discuss together..."
      ...
      intervals [79]:
        xmin = 436.9628
        xmax = 497.63514
        text = ""
      intervals [80]:
        xmin = 497.63514
        xmax = 507.80014
        text = "Oh then you're a second-generation rich! If that's the case, <sil>
your family has sufficient financial support then you can..."
      intervals [81]:
        xmin = 507.80014
        xmax = 507.80014
text = "Hello Jiaojiao, how should college students balance funding, team and market demand when sta
...
      intervals [208]:
        xmin = 992.50424
        xmax = 1104.67465
        text = ""
      intervals [209]:
        xmin = 1104.67465
        xmax = 1104.67465
text = "Hello Jiaojiao, based on our previous discussion about challenges in breeding industry, esp
      intervals [210]:
        xmin = 1104.67465
        xmax = 1106.05717
        text = "That's a good idea,"
      ...

```

B.5 A.4 Benchmark Results

The benchmark results demonstrate the effectiveness of our approach. Below is the output of a typical evaluation:

C CompassJuder Evaluation of AI Assistant's Responses

C.1 Question 1: How can innovation improve the competitiveness of the real estate and renovation industries?

Prompt: You are an assistant skilled in evaluating the quality of responses from meeting agents. Please evaluate the quality of an AI assistant's response to the user's question as an impartial judge. Since the answer involves role-playing, you need to assess it based on the following dimensions:

1. **Accuracy of Facts:** Is the information provided in the answer accurate and based on credible facts and data?
2. **User's Needs Fulfillment:** Does the answer meet the user's purpose and needs, and does it provide a comprehensive and appropriate response to the question?
3. **Conciseness:** Is the answer concise and clear, avoiding redundancy and repetition? A brief answer is important and will score higher.
4. **Structure:** Is the answer well-organized and easy to understand quickly, with a clear focus on the main points?
5. **Completeness:** Does the answer cover most of the key points related to the question, without omitting important information?

We will provide the user's question, a high-quality reference answer, and the AI assistant's answer for evaluation. If the reference answer states that relevant content is not mentioned but the assistant's answer includes relevant information, the assistant's answer should be rated higher, possibly achieving a perfect score of 10.

When you begin your evaluation, follow this process:

1. Compare the AI assistant's answer with the reference answer, highlighting any shortcomings in the assistant's response, and explain further. Keep in mind that the reference answer may not always be better than the assistant's answer. The reference answer's quality level ranges from 3 to 10, so judge accordingly.
2. Evaluate the AI assistant's answer in different dimensions, giving a score between 1 and 10 for each.
3. Finally, provide a comprehensive score based on the evaluations in each dimension.
4. Your scoring must be as strict as possible, following these rules:
 - Significant factual errors or irrelevant/harmful content should result in a low score (1-2).
 - Answers with no serious errors but of low quality should receive a low score (3-4).
 - Answers that generally meet the requirements but are weaker in some dimensions can receive a middle score (5-7).
 - Answers that are similar to or slightly worse than the reference answer can receive a score of 8-9.
 - Answers that surpass the reference answer by being more concise and nearly perfect in all dimensions can receive a score of 10.

Example: A reference answer can score 10 points.

At the end of the evaluation, provide your scoring results in a dictionary format with the following keys:

```
{ 'Fact Accuracy': X, 'User Needs Fulfillment': X, 'Conciseness': X, 'Structure': X, 'Completeness': X, 'Overall Score': X }
```

Remember to evaluate and explain each dimension and ensure that each score is an integer between 1 and 10.

User's Question: How can innovation improve the competitiveness of the real estate and renovation industries?

Reference Answer: Innovation can enhance the competitiveness of the real estate and renovation industries by utilizing technology such as big data and AI to optimize market analysis and customer demand prediction, developing eco-friendly materials to reduce environmental impact, implementing smart home technologies to improve quality of life, and offering personalized design services to meet customer needs.

AI Assistant's Answer: Hello, Jiaojiao.

The real estate and renovation industries can innovate in several ways to improve their competitiveness:

1. Use of digital technology: Utilize technologies such as the internet, IoT, and big data to improve business efficiency, provide better customer experience, and reduce operational costs.
2. Promote green and eco-friendly concepts: With growing consumer awareness of environmental protection, the real estate and renovation industries can promote eco-friendly concepts and provide products and services that align with consumer demands.

3. Develop smart homes: Smart homes are a future trend. The real estate and renovation industries can preemptively develop related products and services to capture market opportunities.

4. Improve design quality: By collaborating with designers, the industry can enhance product design to offer more personalized and differentiated products and services.

5. Build brand reputation: Through offering high-quality products and services, businesses can build a strong brand reputation and attract more consumers.

These are some potential innovative strategies, and their implementation should be based on the specific situation of each company.

Evaluation Results:

{ 'Fact Accuracy': 9, 'User Needs Fulfillment': 8, 'Conciseness': 8, 'Structure': 9, 'Completeness': 8, 'Overall Score': 8 }

C.2 Question 2: How should real estate companies adjust their strategic layout and break the homogeneous competition to adapt to market demand and urbanization development trends, while predicting and responding to the risks of a real estate bubble?

Reference Answer: Real estate companies should optimize their commercial real estate model, innovate business strategies, and break free from homogeneous competition. Simultaneously, market research should be strengthened to precisely target consumers and offer diversified products. Regarding the risk of a real estate bubble, companies should adopt measures like reasonable pricing and accelerating capital turnover to avoid crises.

AI Assistant's Answer: Real estate companies should readjust their strategic planning, optimizing existing commercial real estate models to align with market demand and urbanization trends. To counter homogeneous competition, companies need to innovate their business models. Meanwhile, attention should be paid to the risks of a real estate bubble by taking measures such as reasonable pricing and accelerating capital turnover to mitigate potential crises. In addition, companies should strengthen market research, precisely target their customer base, and offer diversified products to meet various consumer demands.

Evaluation Results:

{ 'Fact Accuracy': 9, 'User Needs Fulfillment': 8, 'Conciseness': 8, 'Structure': 9, 'Completeness': 8, 'Overall Score': 8 }

The response provided by the assistant successfully addresses the question by offering practical steps and techniques for enhancing sales and predicting market trends.

C.3 Prometheus Evaluation of AI Assistant's Responses

C.4 Question 1: How can innovation improve the competitiveness of the real estate and renovation industries?

Instruction: How can innovation improve the competitiveness of the real estate and renovation industries?

Reference Answer: Innovation can enhance the competitiveness of the real estate and renovation industries by utilizing technology such as big data and AI to optimize market analysis and customer demand prediction, developing eco-friendly materials to reduce environmental impact, implementing smart home technologies to improve quality of life, and offering personalized design services to meet customer needs.

AI Assistant's Answer: Hello, Jiaojiao.

The real estate and renovation industries can innovate in several ways to improve their competitiveness:

1. Use of digital technology: Utilize technologies such as the internet, IoT, and big data to improve business efficiency, provide better customer experience, and reduce operational costs.

2. Promote green and eco-friendly concepts: With growing consumer awareness of environmental protection, the real estate and renovation industries can promote eco-friendly concepts and provide products and services that align with consumer demands.

3. Develop smart homes: Smart homes are a future trend. The real estate and renovation industries can preemptively develop related products and services to capture market opportunities.

4. Improve design quality: By collaborating with designers, the industry can enhance product design to offer more personalized and differentiated products and services.

5. Build brand reputation: Through offering high-quality products and services, businesses can build a strong brand reputation and attract more consumers.

These are some potential innovative strategies, and their implementation should be based on the specific situation of each company.

Feedback: The response provides a comprehensive approach to enhancing the competitiveness of the real estate and interior design industry. It effectively addresses the user's question by proposing various strategies, such as digitalization, green initiatives, smart home technology, design improvement, and brand reputation enhancement.

Each point made is relevant and aligns well with the user's inquiry about innovative methods to improve market position. Furthermore, the structure of the response is clear, making it easy to follow and comprehend. The response demonstrates a good grasp of the topic, with no significant omissions. It offers practical solutions without unnecessary elaboration, adhering closely to the user's needs. In conclusion, the response is informative, relevant, and coherent, making it a solid answer to the user's question.

Score: 5

C.5 Question 2: How should real estate companies adjust their strategic layout and break the homogeneous competition to adapt to market demand and urbanization development trends, while predicting and responding to the risks of a real estate bubble?

Instruction: Based on our previous discussions, how should real estate companies adjust their strategic layout to break homogeneous competition, adapt to market demand and urbanization trends, and anticipate and respond to the risks of a real estate bubble?

Reference Answer: Real estate companies should optimize their commercial real estate model, innovate business strategies, and break free from homogeneous competition. Simultaneously, market research should be strengthened to precisely target consumers and offer diversified products. Regarding the risk of a real estate bubble, companies should adopt measures like reasonable pricing and accelerating capital turnover to avoid crises.

AI Assistant's Answer: Real estate companies should readjust their strategic planning, optimizing existing commercial real estate models to align with market demand and urbanization trends. To counter homogeneous competition, companies need to innovate their business models. Meanwhile, attention should be paid to the risks of a real estate bubble by taking measures such as reasonable pricing and accelerating capital turnover to mitigate potential crises. In addition, companies should strengthen market research, precisely target their customer base, and offer diversified products to meet various consumer demands.

Feedback: The response effectively addresses the core issues discussed by focusing on strategic adjustments, innovation, market research, product diversity, and risk management. It offers clear, actionable steps for businesses to adapt to changing market demands and urbanization trends, while also anticipating potential risks from housing bubbles. Furthermore, the response presents these points in a structured and coherent manner, with a logical progression that enhances its clarity and usability. The information is presented in a concise and straightforward manner without repetition, which makes it easy to understand and implement. Lastly, the response covers all key aspects mentioned in the reference answer and demonstrates a comprehensive understanding of the topic at hand. Hence, the response fulfills all criteria set forth in the score rubric, as it exhibits high accuracy, relevance, simplicity, structure, and completeness in addressing the given instruction. So the overall score is 5.

Score: 5