SPEECH WORLD MODEL: CAUSAL STATE–ACTION PLANNING WITH EXPLICIT REASONING FOR SPEECH

Anonymous authors
Paper under double-blind review

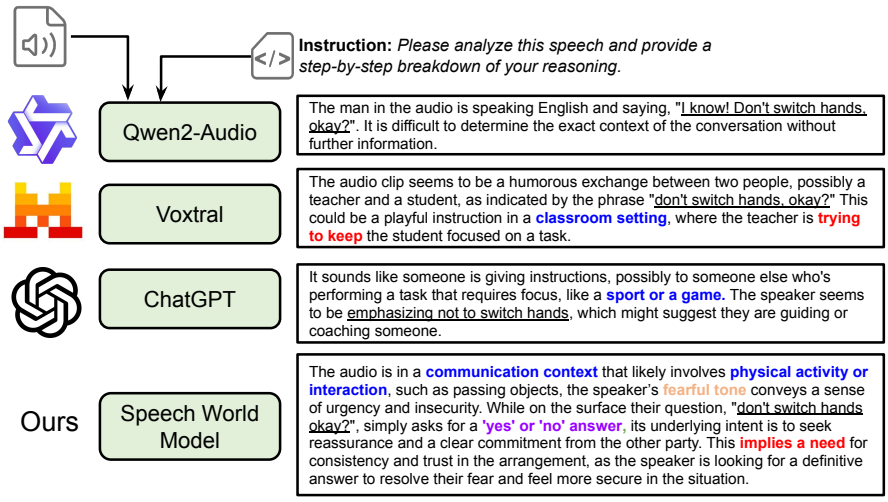


Figure 1: Speech World Model. Demo audio link: `http://bit.ly/4pBJuWP`.

ABSTRACT

Current speech-language models (SLMs) typically use a cascade of speech encoder and large language model, treating speech understanding as a single black box. They analyze the content of speech well but reason weakly about other aspects, especially under sparse supervision. Thus, we argue for explicit reasoning over speech states and actions with modular and transparent decisions. Inspired by cognitive science we adopt a modular perspective and a world model view in which the system learns forward dynamics over latent states. We factorize speech understanding into four modules that communicate through a causal graph, establishing a cognitive state search space. Guided by posterior traces from this space, an instruction-tuned language model produces a concise causal analysis and a user-facing response, enabling counterfactual interventions and interpretability under partial supervision. We present the first graph based modular speech model for explicit reasoning and we will open source the model and data to promote the development of advanced speech understanding.

1 INTRODUCTION

Speech language models (SLMs) (Cui et al., 2024) provide a unified framework for general-purpose speech understanding, enabling multitask perception, reasoning, and multi-turn interaction. At the same time, an ongoing debate persists regarding whether such models truly exhibit genuine reasoning capabilities or merely perform sophisticated pattern matching and statistical abstraction over tokens (Shojaee et al., 2025). One direct observation is that current SLMs appear to combine outputs from isolated tasks—such as automatic speech recognition, emotion recognition, communicative function detection, intent classification, and pragmatic inference—and then present this aggregation *as if it reflects reasoning*. However, this overlooks the *intrinsic dependencies among these speech components* and provides only partial supervision, which we argue is fundamental to enabling genuine

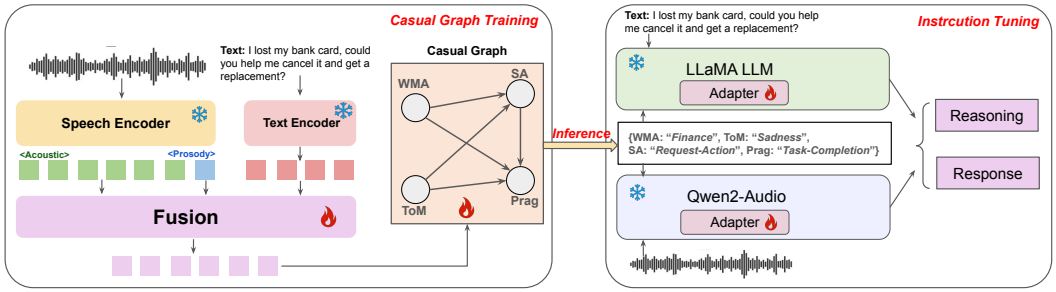


Figure 2: The pipeline of our *Speech World Model*, a two-stage training framework we term "Causal Graph-Guided Explicit Reasoning". In the first stage, a Causal Graph is trained to convert speech into latent states as structured reasoning information. In the second stage, this information is used as an explicit guide for the parameter-efficient instruction tuning of large (speech) language models to enhance their speech understanding and reasoning capabilities.

reasoning in speech understanding. From a human-centered cognitive perspective, speech perception is thought to be hierarchically organized into partially specialized modules—such as acoustic, linguistic, and paralinguistic processing—that interact through causal and reciprocal relations (Hickok & Poeppel, 2007). However, this does not entail designing strictly brain-like modular systems (e.g. verified AI (Seshia et al., 2022)), as human perception itself operates through overlapping and partially opaque processes that remain incompletely understood (Angelaki et al., 2025).

An alternative line of research seeks an intermediate approach that models latent internal states, thereby enabling more interpretable predictions of system dynamics. A representative example is *World Models* (Ha & Schmidhuber, 2018; LeCun, 2022), which posit that the subsequent state is conditionally predictable from any given current state. In the domain of speech, human perceptual systems also anticipate interdependent dynamics across states such as emotion, intent, and linguistic content, such that changes in one dimension systematically constrain others (Hickok & Poeppel, 2007). These observations point to an underlying structure of causal dependencies. It is worth noting that Chain-of-Thought (CoT) prompting (Wei et al., 2022b) represents a parallel approach, which expands the search space to improve goal-directed reasoning at the expense of increased computational cost. However, the search process is not grounded in human speech perception principles.

In this work, we propose *Speech World Models (SWM)* to advance speech understanding and reasoning by modeling the dynamics of internal speech states. SWM comprises four explicit modules—*World Model Action (WMA)* (Mesaros et al., 2018), *Theory of Mind (ToM)* (Sclar et al., 2025), *Speech Act (SA)* (Bothe et al., 2020), and *Pragmatic Intent (Prag)* (Stolcke et al., 2000)—as an approximated subspace of the cognitive perception system. A causal-graph–based perception module is introduced to bridge these components and capture the flow of causal information. In the first stage, this graph is trained in both supervised and semi-supervised settings to establish a cognitive state search space. Notably, leveraging causal dependencies among modules yields faster convergence than training the modules independently. Moreover, because the graph approximates cognitive state structure, it can reliably infer unlabeled modules from partially labeled data, suggesting that causal graphs function as natural generative structures for latent variables. Once the search space is established, in the second stage we incorporate the outputs of the causal graph—samples from the search space, state nodes, and information flow—into prompts for instruction tuning. This provides an explicit reasoning advantage: by enforcing structured reasoning chains, SLMs are guided to reduce hallucinations and achieve greater overall reasoning capacity.

Experiments demonstrate the superiority of SWM over traditional SLMs across a range of reasoning metrics. First, when comparing against a randomly initialized graph without cognitive grounding, we find that our causal graph converges faster during training and achieves substantially higher reasoning scores (e.g., ACE, ICS). Second, instruction-tuning results show that SWM outperforms open-source SLMs such as Qwen1 (Chu et al., 2023), Qwen2-Audio (Chu et al., 2024), and Voxtral (Liu et al., 2025) in reasoning ability, with particularly strong performance in emotion recognition—even surpassing commercial models—highlighting the benefits of explicit reasoning. Third, while the overall speech understanding performance of SWM (as measured by our proposed metrics) remains

slightly below that of Gemini 2.5 Pro (Gemini Team, 2025), our training cost is dramatically lower (only 20 GPU hours), validating the efficiency of the causal-graph–based explicit reasoning paradigm.

2 Related Study

2.1 Cognitive Foundations of Speech Processing

Cognitive neuroscience shows that speech perception and comprehension emerge from modular computations over latent states, rather than a single monolithic process. Core frameworks include the dual–stream model linking ventral (semantic) and dorsal (sensorimotor) pathways (Hickok & Poeppel, 2007), staged accounts of syntactic and semantic processing (Friederici, 2011), and analyses of the superior temporal gyrus that reveal intermediate phonological and prosodic representations (Leonard & Chang, 2022). Complementary perspectives highlight cortical oscillations aligned with hierarchical linguistic units (Giraud & Poeppel, 2012) and predictive coding views of the brain as a generative model updating latent states via prediction errors (Friston et al., 2021). Together, these perspectives converge on four principles: (i) modular and hierarchical decomposition of speech, (ii) forward predictive dynamics, (iii) integration of perception and action, and (iv) generative modeling of latent auditory states. Our Speech World Model operationalizes these principles computationally.

2.2 Speech Language Models and Reasoning

Early work on speech language models (Gong et al., 2024; 2023; Tang et al., 2024; Kong et al., 2024) pioneered the use of instruction tuning pipelines for speech understanding and preliminary reasoning tasks. Subsequent studies adopted Chain-of-Thought (CoT) prompting (Wei et al., 2022b) to construct reasoning-chain search spaces, which advanced reasoning capabilities in speech-related tasks (Ma et al., 2025; Xie et al., 2025a;b; Kong et al., 2025); see also (Cui et al., 2024) for a broader review. Nevertheless, these CoT-based approaches remain largely heuristic and are not grounded in principles of human speech cognition (Hickok & Poeppel, 2007).

2.3 World models

The concept of world models has long-standing roots in cognitive science and neuroscience, where internal models are thought to support prediction, planning, and reasoning in human perception and action (Craik, 1967; Friston, 2005; Hickok & Poeppel, 2007). Ha & Schmidhuber (2018) popularized the term by introducing a generative framework based on latent dynamics, demonstrating how agents could "dream" in an internal environment. This line of work inspired subsequent advances such as PlaNet (Hafner et al., 2019b), Dreamer (Hafner et al., 2019a), and MuZero (Schrittwieser et al., 2020), which emphasized model-based reinforcement learning and planning from learned dynamics. More recently, LeCun (2022) argued that world models should be central to autonomous machine intelligence, framing them as self-supervised systems that learn hierarchical predictive representations. Parallel efforts have explored domain-specific instantiations for vision (Garrido et al., 2024) , motor control (Assran et al., 2025), and language tasks (Lin et al., 2024; Feng et al., 2025).

3 Speech World Model Systems

In this section, we present the methodology behind our proposed *Speech World Model (SWM)*. At its core, the framework relies on a causal graph that factorizes speech understanding into modular components, enabling explicit reasoning over speech states and actions. These states then condition (spoken) language models, which are fine-tuned to generate both a transparent reasoning trace and a potential response to given utterance. This design allows us to validate the hypothesis that explicit state representation improves Speech Language Model's reasoning capacity and reduces hallucinations (Atwany et al., 2025).

3.1 World Model Causal Graph

We proposed a probabilistic causal model, represented by a Directed Acyclic Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a set of nodes representing different modules and $\mathcal{E}$ is a set of directed edges representing causal relationships between modules. This structure allows us to explicitly model the dependencies between different aspects of speech and perform causal reasoning.

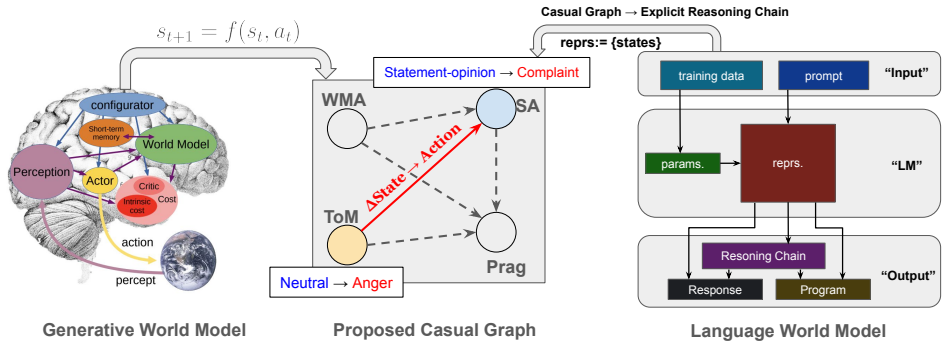


Figure 3: A unified perspective on world models. Both the Generative (left(Garrido et al., 2024)) and Language (right) world models are forward dynamic models. Our Causal Graph (center) provides a structured, explicit formulation of these dynamics. See Appendix. A.1 for details.

3.1.1 Modules and States

We define four key modules, each capturing a distinct aspect of speech understanding, forming the *cognitive perceptual subspace*. The complete label spaces are provided in the Appendix. A.4.2.

- **World Model Activation (WMA)** (Mesaros et al., 2018) captures context and situation of the speech environment.
- **Theory of Mind (ToM)** (Sclar et al., 2025) models the speaker's mental and emotional states.
- **Speech Act (SA)** (Bothe et al., 2020) identifies the communicative function of an utterance.
- **Pragmatic Intent (Prag)** Stolcke et al. (2000) represents the speaker's underlying intentions.

Graph Perspective. Each module corresponds to a node, instantiated as a neural network performing its own computation. The outputs of these modules are categorical latent states, and collectively they form the structured representation of speech, as presented in Fig. 3.

World Model Perspective. Inspired by the essence of image world models (Garrido et al., 2024), we view the entire system as a forward dynamics model: the current latent state, together with an action, leads to the next latent state. Within this view, the notion of action extends beyond external commands. Instead, an action can be understood as the causal influence exerted by one state on another. For example, when the Theory of Mind module shifts from *neutral* to *anger*, the downstream Speech Act may change from *Statement-opinion* to *Complaint/Escalate*, as shown in Fig. 3.

Thus, actions in our framework are best understood as the state transitions induced by causal relationships within the graph. Formally, for a node v with parents $\mathrm{Pa}(v)$:

$$S_v = f_v\big(\{S_u : u \in \mathrm{Pa}(v)\}, A_{u\to v}\big), \tag{1}$$

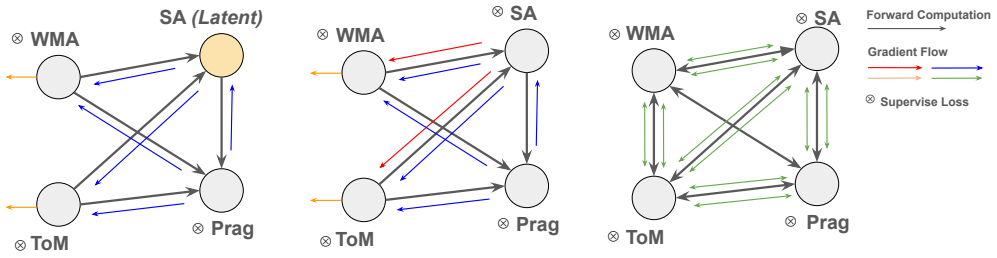
where S_v is the latent state at node v, and $A_{u\to v}$ denotes the action, interpreted as the causal influence (action) of parent node u on v.

3.1.2 Graph Calculation

Given an input speech signal X, the casual graph infers a joint posterior distribution over latent states $Z = \{z_{WMA}, z_{ToM}, z_{SA}, z_{Prag}\}$. By the DAG factorization:

$$\begin{aligned} p(Z|X) = p(z_{WMA}|X) \cdot p(z_{ToM}|X) \cdot \\ p(z_{SA}|z_{WMA}, z_{ToM}, X) \cdot p(z_{Prag}|z_{SA}, z_{ToM}, z_{WMA}, X) \end{aligned} \tag{2}$$

Each conditional $p(z_v|\mathrm{Pa}(v), X)$ is parameterized by a neural module $f_v(\cdot;\theta_v)$, modeled as a multi-class classifier. Our proposed casual graph is indicated in Fig. 4 (B). The detailed calculation and model architecture are detailed in Appendix. A.2.1.



(A) Semi-supervised Casual Graph (B) Fully-supervised Casual Graph (C) Fully-supervised Random Graph

Figure 4: Comparison of gradient flow under different training scenarios.(A) Semi-supervised causal graph: when a module lacks direct supervision, it act as latent variable generator, and gradients propagate backward through causal edges from downstream modules. (B) Fully-supervised causal graph: all modules have labels, so losses are applied locally, while causal dependencies still guide gradient flow. (C) Fully-supervised random graph: supervision is present at every node but edges are unstructured, leading to redundant and less efficient gradient propagation.

3.1.3 Multi-task training with Casual graph

Causal graph is trained in a multi-task manner to form reasoning-chain search space. The supervised loss is shown in Eq.3, where $m_{i,v}$ indicates whether the label for node v is available in sample i.

$$\mathcal{L}_{\text{sup}} = \sum_{i=1}^{N} \sum_{v \in \mathcal{V}} m_{i,v} \, \text{CE}(y_{i,v}, S_{i,v}), \tag{3}$$

Teacher forcing is applied edge-wise: each child node receives either the ground-truth parent state or the predicted distribution. For edge $u \to v$:

$$\tilde{S}_{i,u} \;=\; \tau_{i,u\to v}\,\text{onehot}(y_{i,u}) \;+\; \big(1-\tau_{i,u\to v}\big)\,\text{stopgrad}(S_{i,u}), \quad \tau_{i,u\to v} \sim \text{Bernoulli}(p_{u\to v}). \tag{4}$$

The mixed signal $\tilde{S}_{i,u}$ is used as the parent input when computing the state of child v.

3.1.4 Semi-supervised learning

Motivation. Annotations across modules are costly and often incomplete (e.g., WMA labels are missing more frequently). If parameter updates were restricted only to nodes with available labels, a large fraction of training data would be wasted. Semi-supervised training allows unlabeled parent nodes to be optimized through the supervision signals of their labeled children.

Gradient Flow. Consider a parent node j without labels in sample i ($m_{i,j} = 0$), where one of its child node k has a label ($m_{i,k} = 1$). To ensure a differentiable path to the unlabeled parent node j, teacher forcing is disabled ($p_{j\to k}$=0), forcing the child k to rely on the parent's continuous prediction $S_{i,j}$. Thus, the loss $\mathcal{L}_{i,k} = \text{CE}(y_{i,k}, S_{i,k})$ propagates to θ_j through the chain rule:

$$\frac{\partial \mathcal{L}_i}{\partial \theta_j} = \sum_{\substack{k:\, m_{i,k}=1 \\ j \in \text{Pa}(k)}} \frac{\partial \mathcal{L}_{i,k}}{\partial \eta_{i,k}} \frac{\partial \eta_{i,k}}{\partial S_{i,j}} \frac{\partial S_{i,j}}{\partial \eta_{i,j}} \frac{\partial \eta_{i,j}}{\partial \theta_j} \;\neq\; 0. \tag{5}$$

where the gradient also flows into the parent distributions $p(s_u \mid \xi_u)$. Thus unlabeled parents act as *latent variable generators* that receive updates from their supervised children as shown in Fig. 4 (A).

Multi-task supervision ensures "learn whenever a label is available" (Eq. 3), while semi-supervised training ensures "learn even when labels are missing" via chain gradients (Eq. 5). Together they allow effective learning from incompletely annotated data under the causal graph structure.

3.2 Random Graph

To validate the benefits of our proposed causal structure, grounded by cognitive speech perception, we introduce the *Random Graph* as a strong, unstructured and data-driven baseline. This model replaces

the predefined DAG with a fully connected graph where every module can, in principle, influence every other module. Consequently, the prediction for each module is conditioned on both the speech input X and the initial, teacher-forced estimates from **all** other modules in the graph, allowing the model to learn arbitrary dependencies directly from the data.

$$p(Z|X) = \prod_{v \in \mathcal{V}} p(z_v | Z \setminus \{z_v\}, X) \tag{6}$$

By comparing this model with our structured causal graph, we can quantify the contribution of imposing theoretically-grounded causal priors on speech understanding. The detailed, step-by-step computation for this iterative process is provided in Appendix. A.2.2.

3.3 Speech understanding and reasoning

To bridge the gap between the structured latent states inferred by the causal graph and human-interpretable analysis, we employ an instruction tuning (Wei et al., 2022a) phase. This final stage is designed to validate our core hypothesis: *Conditioning on these explicit graph states tends to guide the reasoning-chain search toward human-aligned spaces.*. To this end, the model is trained to produce a transparent reasoning process and a contextually appropriate response based on the graph's outputs. We explore two distinct settings for this purpose: a language-only approach and a multi-modal approach.

3.3.1 Language-only Setting

In this setting, we fine-tune a standard LLM to reason over the symbolic outputs of the pre-trained causal graph $\mathcal{G}$. The input is prompted with a general instruction (Instr) and a textual serialization of the graph's output, $I(\mathcal{G}(x))$, which encodes the inferred states and their causal relationships. The prompts we used for training are detailed in Appendix. A.5.2.

The model is trained to produce a target sequence y that contains both a step-by-step *reasoning process* and a potential *dialogue response*. The training objective is to minimize the standard cross-entropy loss over the target sequence, where θ represents the parameters of the LLM, and the target y is the ground-truth reasoning and response pair.

$$\mathcal{L}_{\text{IT}}(\theta) = -\sum_{(x,y)\in\mathcal{D}} \log p_\theta\Big(y \,\Big|\, \text{Instr},\ I(\mathcal{G}(x))\Big) \tag{7}$$

3.3.2 multi-modal setting

This approach leverages a speech language model to directly ground the symbolic states in the acoustic properties of the raw speech signal. The input to the SLM consists of the instruction (Instr), the raw speech input x, and the set of inferred states $\{S_v\}_{v\in\mathcal{V}}$. The model's task is to generate the same target sequence y containing a reasoning process and a response.

$$\mathcal{L}_{\text{IT}}(\theta) = -\sum_{(x,y)\in\mathcal{D}} \log p_\theta\Big(y \,\Big|\, \text{Instr},\ x,\ S_{\text{WMA}},\ S_{\text{ToM}},\ S_{\text{SA}},\ S_{\text{Prag}}\Big) \tag{8}$$

This objective forces the SLM to build a joint understanding of the audio signal and the structured states, effectively learning to associate "what was said" and "how it was said" with the "why" provided by our causal graph's analysis.

4 Theoretical Analysis of the Speech World Model

The use of a causal graph for speech understanding leads to more efficient training by providing a strong structural prior. This prior improves learning in two key ways: by optimizing gradient flow within the graph and by constraining the search space for downstream instruction tuning. See more detailed derivations in Appendix. A.3.

Structured Priors and Optimized Gradient Flow. The causal graph simplifies the learning problem by modeling the joint distribution over latent variables as a factorized probability, where

each module is conditioned only on its causal parents (Pearl, 2009):

$$P(Y_1, \ldots, Y_N|X) = \prod_{i=1}^{N} P(Y_i|X, Y_{\mathrm{Pa}(i)}) \tag{9}$$

This factorization directly yields a more efficient and stable gradient flow. The gradient for a module v is computed based only on its parents, precluding the need to discover relationships from scratch and focusing updates along causally relevant pathways. This structure minimizes *learning redundancy* defined as wasteful gradient updates across unrelated components, which we can quantify as:

$$\text{Redundancy} \propto \sum_{i=1}^{N} \Pi_i \left\|\nabla_{\theta_i} \mathcal{L}(\theta_i)\right\| \tag{10}$$

where Π_i is an importance indicator for module i. By pruning unnecessary dependencies, the model's transparent design directly accelerates convergence, a known benefit of causality-inspired explainability (Schölkopf et al., 2021).

Constrained Search Space for Instruction Tuning. This causal structure provides a critical advantage for downstream instruction tuning. The graph constrains the combinatorially large search space of latent cognitive states, $\mathcal{Y} = \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_N$, to a valid, low-entropy subspace $\mathcal{S}_\mathcal{G} \subset \mathcal{Y}$. This reduction in complexity is quantifiable by information entropy:

$$H(Y|X, \mathcal{G}) = \sum_{i=1}^{N} H(Y_i|Y_{\mathrm{Pa}(i)}, X, \mathcal{G}) \ll \sum_{i=1}^{N} \log |\mathcal{Y}_i| \tag{11}$$

This simplification transforms the learning task for the language model. Instead of learning a complex joint policy $\pi_\phi(Y, R|X)$ to generate both a reasoning chain Y and a response R, as in standard Chain-of-Thought (Wei et al., 2022b; Yao et al., 2023), our method allows learning a simpler conditional policy $\pi_\phi(R|X, Y_\mathcal{G})$. Providing the causally valid state $Y_\mathcal{G}$ as input acts as a *control variate*, reducing gradient variance and significantly improving the sample efficiency of instruction tuning.

5 Experiments

5.1 Data

5.1.1 Real-world dataset

- **MELD** (Poria et al., 2019) is a conversational dataset for emotion recognition. It consists of over 13,000 utterances from the TV series Friends, annotated with emotion and sentiment labels.
- **IEMOCAP** (Busso et al., 2008) is an audiovisual and multi-speaker database of dyadic interactions between actors, containing approximately 12 hours of data annotated with categorical and dimensional emotion labels.
- **SLURP** (Bastianelli et al., 2020) is a challenging spoken language understanding (SLU) dataset featuring single-turn user interactions with a voice assistant. Utterances are annotated with a hierarchical schema, providing both user intent and corresponding action.
- **VoxCeleb** (Nagrani et al., 2017) is a large-scale text-independent speaker identification dataset. It comprises hundreds of thousands of utterances from celebrities, extracted from YouTube videos.

We use the above datasets as the primary speech sources for our experiments. Detailed statistics of these datasets are provided in Appendix. A.4.1.

5.1.2 Label-generation

A central challenge is that speech data usually comes with only *partial supervision*: some modules may be annotated while others are not. To construct complete instances for fully-supervised graph training and instruction tuning, we design a two-stage label generation procedure.

Stage 1: Label completion. Given transcription and any known module labels, the goal is to predict missing labels for the remaining modules. We prompt a large language model Vicuna (Chiang et al., 2023) with the complete label space and enforce strict decoding into valid categories. This stage can be interpreted as a constrained imputation problem, where fuzzy mapping is applied to handle minor mismatches between predicted tokens and the canonical label set.

Stage 2: Reasoning and response generation. With the completed labels and transcription as input, we query the LLM again to generate two additional outputs: (i) a comprehensive reasoning analysis how the predicted states interact through the causal graph and jointly shape the interpretation of the utterance, and (ii) a potential response aligned with the inferred states.

Full details of prompts, model configuration, and other implementation specifics are provided in the Appendix. A.4.2.

5.2 METRICS

Node quality. We evaluate the performance of individual modules (nodes) on their respective sub-tasks using standard classification *Accuracy* and *weighted F1-Score*.

Edge causal effect. To validate each learned edge $X \rightarrow Y$, we measure its causal strength using the *Average Causal Effect (ACE)* (Pearl, 2010) and its directional consistency with the *Intervention Consistency Score (ICS)*. *ACE* indicates the magnitude of post-intervention effects, while *ICS* indicates if effects align with a predefined hypothesis. See Appendix. A.8.1 for computational details.

Instruction tuning. To provide a scalable and consistent evaluation of our instruction-tuned models, we employ the Model-as-Judge (M.J.) methodology (Zheng et al., 2023), where GPT-4o (OpenAI, 2024) acts as an impartial judge to score both reasoning and response quality. The overall M.J. score is a weighted sum of reasoning and response scores. *EM* and *EA* refer to emotion mention rate and emotion classification accuracy, respectively, while *R-Len* represents the length of the reasoning process. The detailed prompting strategy and scoring rubric are indicated in Appendix. A.9.

5.3 EXPERIMENTS SETUP

Our method involves a two-stage training process: causal graph training followed by instruction tuning. We train the causal graph under three settings: fully-supervised (for causal and random graphs) and semi-supervised (for the causal graph). Each module is an MLP-based classifier, with the WMA and ToM modules enhanced by a preceding temporal attention layer. We also conduct ablation studies on the causal graph, with detailed results available in Appendix. A.6. For instruction tuning, we use LoRA (Hu et al., 2022) for parameter-efficient instruction tuning of Llama3.1-8B Meta (2024) (language-only) and Qwen2-Audio (Chu et al., 2024) (multi-modal). As a baseline, we also fine-tune Qwen2-Audio using only CoT-style prompts, excluding any outputs from our causal graph. Detailed model architecture training configurations can be found in Appendix. A.5 and A.7.

5.4 GRAPH EVALUATION

The results of our graph evaluation are presented in Table. 1 and Table. 2, with a detailed breakdown of edge causal effects shown in Fig. 5. These results first demonstrate the **efficiency** of our Causal Graph, which **achieves node accuracy** comparable to a Random Graph baseline at a significantly **lower training cost**. More importantly, it provides **stable and interpretable causality** by quantifying effects with *ACE* and *ICS* metrics. In contrast, the Random Graph's internal logic is proven unstable and dependent on training hyperparameters. Furthermore, our Causal Graph proves robust under incomplete supervision. In semi-supervised settings, it **maintains strong performance**, with any reduction in causal strength logically localized to the edges connected to the unsupervised module, highlighting the reliability of its structure.

Table 1: Node and Edge performance of the Causal Graph under various supervision settings.

Method	Setting	Node Quality (Accuracy %, ↑)				Edge Casual Effect	
		WMA	ToM	SA	Prag	Ave. ACE (%, ↑)	Ave. ICS (%, ↑)
Fully-supervised	–	69.4	73.5	65.3	81.4	23.57	43.29
Semi-supervised	Latent module: WMA	34.8	75.0	70.7	83.2	21.71	26.9
	Latent module: ToM	69.1	43.3	69.6	83.5	21.98	28.9
	Latent module: SA	69.3	77.0	34.4	82.5	21.65	29.3
Random Graph	–	69.7	74.0	67.5	83.6	–	–

5.5 SPEECH UNDERSTANDING AND REASONING EVALUATION

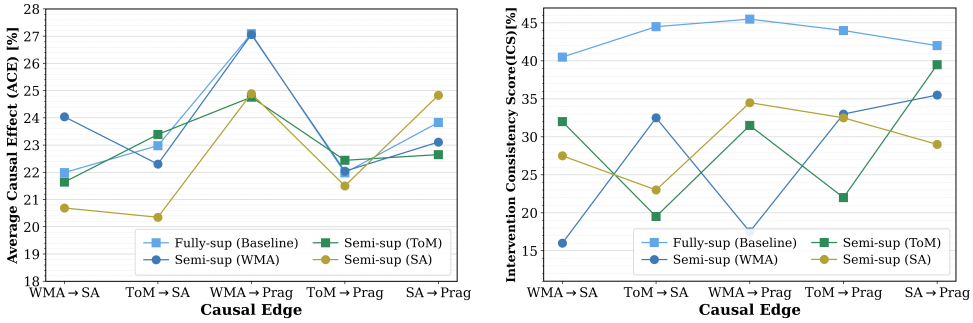


Figure 5: *ACE* and *ICS* of each casual edge under both fully-supervised and semi-supervised training.

Table 2: Node quality and information flow analysis (see Appendix.A.8.2for details) for random graph baseline.

p(teacher_force)	Node Quality (Accuracy %, ↑) WMA	ToM	SA	Prag	Information-flow Strongest	Weakest
0.0	70.1	75.2	66.1	82.8	SA → WMA	WMA → Prag
0.3	69.7	74.0	67.5	83.6	ToM → SA	ToM → Prag
0.5	69.5	73.1	65.9	82.0	WMA → SA	ToM → Prag
0.8	69.4	74.4	70.3	83.2	ToM → Prag	Prag → ToM
1.0	70.1	73.9	66.3	82.3	SA → ToM	ToM → Prag

Our speech understanding and reasoning results (Table. 3) reveal the distinct contributions of our reasoning framework. To isolate these effects, we created a tuned baseline by fine-tuning Qwen2-Audio with our CoT-style instruction data. This **baseline alone surpasses other open-source models**, confirming the high quality of our instruction tuning data, and our full SWM models, which leverage the causal graph for explicit reasoning, **significantly outperform this tuned baseline**. This demonstrates that while good data is important, the causal graph's explicit guidance is the critical factor driving superior reasoning ability. This benefit is particularly stark in **emotion recognition**, where our model's accuracy exceeds even top proprietary models. We posit that this high fidelity stems from the graph grounding the model's analysis, thus reducing hallucination and enforcing a more structured reasoning process. Finally, while a model like Gemini 2.5 Pro (Gemini Team, 2025) holds an edge in the overall score, **our approach achieves this competitive performance at a dramatically lower training cost**. This validates our causal graph-guided paradigm as a highly efficient and effective path to complex speech understanding, prioritizing resourcefulness without a significant compromise in quality.

Table 3: Performance comparison against open-source and proprietary baselines.

Method	**Prompt Style** (Inference)	**Overall M.J. Score** ↑ $0.6 \times Rs + 0.4 \times Rp$	**Reasoning Score** ↑ Rs	**Response Score** ↑ Rp	**Reasoning Breakdown (%)** ↑ *EM*	*EA*	**R-Len** *(words)*
Our Models							
SWM (Llama3.1-8b)	CoT	**7.81**	**7.84**	7.76	**97.80**	66.26	**105.70**
SWM (Qwen2-Audio)	CoT	7.59	7.26	**8.08**	91.80	**71.02**	104.64
Tuned Baseline							
Qwen2-Audio-CoT	CoT	5.18	4.76	5.82	92.11	34.72	102.44
Baselines							
Qwen-Audio (Chu et al., 2023)	Direct	2.70	2.20	3.46	14.20	8.00	28.99
Qwen2-Audio (Chu et al., 2024)	Direct	2.63	2.08	3.47	5.14	15.38	15.60
Qwen2-Audio (Chu et al., 2024)	CoT	2.39	1.96	3.04	6.11	17.50	21.19
Voxtral (Liu et al., 2025)	Direct	2.89	2.46	3.54	10.28	5.88	69.64
Voxtral (Liu et al., 2025)	CoT	2.92	2.52	3.52	10.89	5.56	71.42
Proprietary Models							
GPT-4o (OpenAI, 2024)	CoT	7.41	6.98	8.06	68.20	45.16	105.23
Gemini 2.5 Pro (Gemini Team, 2025)	CoT	**8.12**	**8.02**	**8.28**	82.47	51.29	112.62

6 Limitations and Conclusions

In this work, we proposed Speech World Model (SWM), which surpasses the current open-sourced speech language models in understanding and reasoning ability by modeling the dynamics of internal speech states. However, there are still some limitations. First, the current model includes only four modules, while more diverse states could enhance speech understanding and better capture complex speech dynamics. Second, the causal graph in SWM is predefined, limiting the model's ability to adapt to unseen dependencies. Future work could focus on learning adaptive casual structures to increase flexibility. Third, while instruction tuning effectively integrates reasoning traces and responses, it relies heavily on the label generation pipeline, where inaccuracies can propagate to both the reasoning and response stages. Therefore, achieving efficient data annotation and generation for speech remains a crucial and collaborative challenge within the speech community.

7 Ethics Statement

Our work adheres to the ICLR Code of Ethics. All experiments were conducted on publicly available datasets, ensuring data privacy and consent. Our research is intended for positive applications, and we strongly discourage any misuse of this technology for surveillance or manipulative purposes.

8 Reproducibility Statement

To ensure the reproducibility of our results, we will make our source code, including implementation, training, and evaluation scripts, publicly available at `https://github.com/eureka235/eureka235.github.io`. All datasets used are public and cited accordingly. The main experimental setup is described in Sec. 5. For a more detailed breakdown of the model architecture, hyperparameters, and implementation specifics, please refer to Appendix. A.4 - A.9.

References

Dora Angelaki, Brandon Benson, Julius Benson, Daniel Birman, Niccolò Bonacchi, Kcénia Bougrova, Sebastian A Bruijns, Matteo Carandini, Joana A Catarino, et al. A brain-wide map of neural activity during complex behaviour. *Nature*, 645(8079):177–191, 2025.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Hanin Atwany, Abdul Waheed, Rita Singh, Monojit Choudhury, and Bhiksha Raj. Lost in transcription, found in distribution shift: Demystifying hallucination in speech foundation models. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 23181–23203, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.1190. URL `https://aclanthology.org/2025.findings-acl.1190/`.

Emanuele Bastianelli, Andrea Vanzo, Pawel Swietojanski, and Verena Rieser. SLURP: A spoken language understanding resource package. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7252–7262, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.588. URL `https://aclanthology.org/2020.emnlp-main.588/`.

Chandrakant Bothe, Cornelius Weber, Sven Magg, and Stefan Wermter. EDA: Enriching emotional dialogue acts using an ensemble of neural annotators. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pp. 620–627, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL `https://aclanthology.org/2020.lrec-1.78/`.

Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N. Chang, Sungbok Lee, and Shrikanth S. Narayanan. IEMOCAP: interactive emotional dyadic motion capture database. *Language Resources and Evaluation*, 42(4):335–359, December 2008.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Jian Wu, Michael Zeng, Xiangzhan Yu, and Furu Wei. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE JSTSP*, 2022.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. URL `https://lmsys.org/blog/2023-03-30-vicuna/`.

Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models. *arXiv preprint arXiv:2311.07919*, 2023.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*, 2024.

Kenneth James Williams Craik. *The nature of explanation*, volume 445. CUP Archive, 1967.

Wenqian Cui, Dianzhi Yu, Xiaoqi Jiao, Ziqiao Meng, Guangyan Zhang, Qichao Wang, Yiwen Guo, and Irwin King. Recent advances in speech language models: A survey. *arXiv preprint arXiv:2410.03751*, 2024.

Florian Eyben, Martin Wöllmer, and Björn Schuller. opensmile - The munich versatile and fast open-source audio feature extractor. In *Proceedings of the ACM Multimedia Conference (MM)*, pp. 1459–1462, Florence, Italy, October 2010. ACM. ISBN 978-1-60558-933-6.

Jiahai Feng, Stuart Russell, and Jacob Steinhardt. Monitoring latent world states in language models with propositional probes. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=0yvZm2AjUr.

Angela D Friederici. The brain basis of language processing: from structure to function. *Physiological Reviews*, 91(4):1357–1392, 2011. doi: 10.1152/physrev.00006.2011.

Karl Friston. A theory of cortical responses. *Philosophical transactions of the Royal Society B: Biological sciences*, 360(1456):815–836, 2005.

Karl Friston, Noor Sajid, Daniel Quiroga-Martinez, Thomas Parr, Cathy J Price, and Emily Holmes. Active inference: a process theory of language. *Neuroscience & Biobehavioral Reviews*, 134: 1053–1067, 2021. doi: 10.1016/j.neubiorev.2021.01.016.

Quentin Garrido, Mahmoud Assran, Nicolas Ballas, Adrien Bardes, Laurent Najman, and Yann LeCun. Learning and leveraging world models in visual representation learning, 2024. URL https://arxiv.org/abs/2403.00504.

Google Gemini Team. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities, 2025. URL https://arxiv.org/abs/2507.06261.

Anne-Lise Giraud and David Poeppel. Cortical oscillations and speech processing: emerging computational principles and operations. *Nature Neuroscience*, 15(4):511–517, 2012. doi: 10.1038/nn.3063.

Yuan Gong, Alexander H. Liu, Hongyin Luo, Leonid Karlinsky, and James Glass. Joint audio and speech understanding. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 1–8, 2023. doi: 10.1109/ASRU57964.2023.10389742.

Yuan Gong, Hongyin Luo, Alexander H. Liu, Leonid Karlinsky, and James R. Glass. Listen, think, and understand. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=nBZBPXdJlC.

David Ha and Jürgen Schmidhuber. World models. 2018. doi: 10.5281/ZENODO.1207631. URL https://zenodo.org/record/1207631.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019a.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pp. 2555–2565. PMLR, 2019b.

Gregory Hickok and David Poeppel. The cortical organization of speech processing. *Nature Reviews Neuroscience*, 8(5):393–402, 2007. doi: 10.1038/nrn2113.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Zhifeng Kong, Arushi Goel, Rohan Badlani, Wei Ping, Rafael Valle, and Bryan Catanzaro. Audio flamingo: A novel audio language model with few-shot learning and dialogue abilities. *arXiv preprint arXiv:2402.01831*, 2024.

Zhifeng Kong, Arushi Goel, Joao Felipe Santos, Sreyan Ghosh, Rafael Valle, Wei Ping, and Bryan Catanzaro. Audio flamingo sound-cot technical report: Improving chain-of-thought reasoning in sound understanding, 2025. URL https://arxiv.org/abs/2508.11818.

Yann LeCun. A Path Towards Autonomous Machine Intelligence Version. 2022.

Matthew K. Leonard and Edward F. Chang. Speech computations of the human superior temporal gyrus. *Annual Review of Psychology*, 73(1):79–102, 2022. doi: 10.1146/annurev-psych-022321-035256. URL https://doi.org/10.1146/annurev-psych-022321-035256.

Jessy Lin, Yuqing Du, Olivia Watkins, Danijar Hafner, Pieter Abbeel, Dan Klein, and Anca Dragan. Learning to model the world with language. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=7dP6Yq9Uwv.

Alexander H. Liu, Andy Ehrenberg, Andy Lo, Clément Denoix, Corentin Barreau, Guillaume Lample, Jean-Malo Delignon, Khyathi Raghavi Chandu, Patrick von Platen, Pavankumar Reddy Muddireddy, Sanchit Gandhi, Soham Ghosh, Srijan Mishra, Thomas Foubert, Abhinav Rastogi, Adam Yang, Albert Q. Jiang, Alexandre Sablayrolles, Amélie Héliou, Amélie Martin, Anmol Agarwal, Antoine Roux, Arthur Darcet, Arthur Mensch, Baptiste Bout, Baptiste Rozière, Baudouin De Monicault, Chris Bamford, Christian Wallenwein, Christophe Renaudin, Clémence Lanfranchi, Darius Dabert, Devendra Singh Chaplot, Devon Mizelle, Diego de las Casas, Elliot Chane-Sane, Emilien Fugier, Emma Bou Hanna, Gabrielle Berrada, Gauthier Delerce, Gauthier Guinet, Georgii Novikov, Guillaume Martin, Himanshu Jaju, Jan Ludziejewski, Jason Rute, Jean-Hadrien Chabran, Jessica Chudnovsky, Joachim Studnia, Joep Barmentlo, Jonas Amar, Josselin Somerville Roberts, Julien Denize, Karan Saxena, Karmesh Yadav, Kartik Khandelwal, Kush Jain, Lélio Renard Lavaud, Léonard Blier, Lingxiao Zhao, Louis Martin, Lucile Saulnier, Luyu Gao, Marie Pellat, Mathilde Guillaumin, Mathis Felardos, Matthieu Dinot, Maxime Darrin, Maximilian Augustin, Mickaël Seznec, Neha Gupta, Nikhil Raghuraman, Olivier Duchenne, Patricia Wang, Patryk Saffer, Paul Jacob, Paul Wambergue, Paula Kurylowicz, Philomène Chagniot, Pierre Stock, Pravesh Agrawal, Rémi Delacourt, Romain Sauvestre, Roman Soletskyi, Sagar Vaze, Sandeep Subramanian, Saurabh Garg, Shashwat Dalal, Siddharth Gandhi, Sumukh Aithal, Szymon Antoniak, Teven Le Scao, Thibault Schueller, Thibaut Lavril, Thomas Robert, Thomas Wang, Timothée Lacroix, Tom Bewley, Valeriia Nemychnikova, Victor Paltz, Virgile Richard, Wen-Ding Li, William Marshall, Xuanyu Zhang, Yihan Wan, and Yunhao Tang. Voxtral, 2025. URL https://arxiv.org/abs/2507.13264.

Ziyang Ma, Zhuo Chen, Yuping Wang, Eng Siong Chng, and Xie Chen. Audio-cot: Exploring chain-of-thought reasoning in large audio language model, 2025. URL https://arxiv.org/abs/2501.07246.

Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. A multi-device dataset for urban acoustic scene classification, 2018. URL https://arxiv.org/abs/1807.09840.

Meta. meta-llama/llama-3.1-8b-instruct, 2024. URL https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct. Accessed: 2025-02-21.

Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. Voxceleb: A large-scale speaker identification dataset. In *Interspeech 2017*, pp. 2616–2620, 2017. doi: 10.21437/Interspeech.2017-950.

OpenAI. Gpt4-o. https://openai.com/index/hello-gpt-4o/, 2024.

J Pearl. *Causality: Models, Reasoning, and Inference, Second Edition*. Cambridge University Press, London, 2009.

Judea Pearl. Causal inference. *Causality: objectives and assessment*, pp. 39–58, 2010.

Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, Gautam Naik, Erik Cambria, and Rada Mihalcea. MELD: A multimodal multi-party dataset for emotion recognition in conversations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 527–536, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1050. URL https://aclanthology.org/P19-1050/.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, 2020. URL https://arxiv.org/abs/1910.01108.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021. doi: 10.1109/JPROC.2021.3058954.

Melanie Sclar, Jane Dwivedi-Yu, Maryam Fazel-Zarandi, Yulia Tsvetkov, Yonatan Bisk, Yejin Choi, and Asli Celikyilmaz. Explore theory of mind: program-guided adversarial data generation for theory of mind reasoning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=246rHKUnnf.

Sanjit A Seshia, Dorsa Sadigh, and S Shankar Sastry. Toward verified artificial intelligence. *Communications of the ACM*, 65(7):46–55, 2022.

Parshin Shojaee, Iman Mirzadeh, Keivan Alizadeh, Maxwell Horton, Samy Bengio, and Mehrdad Farajtabar. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. *arXiv preprint arXiv:2506.06941*, 2025.

Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26(3): 339–374, 2000. URL https://aclanthology.org/J00-3003/.

Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun MA, and Chao Zhang. SALMONN: Towards generic hearing abilities for large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=14rn7HpKVk.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*, 2022a. URL https://openreview.net/forum?id=gEZrGCozdqR.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022b.

Jingran Xie, Shun Lei, Yue Yu, Yang Xiang, Hui Wang, Xixin Wu, and Zhiyong Wu. Leveraging chain of thought towards empathetic spoken dialogue without corresponding question-answering data, 2025a. URL https://arxiv.org/abs/2501.10937.

Zhifei Xie, Mingbao Lin, Zihang Liu, Pengcheng Wu, Shuicheng Yan, and Chunyan Miao. Audio-reasoner: Improving reasoning capability in large audio language models, 2025b. URL https://arxiv.org/abs/2503.02318.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik R Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=5Xc1ecxO1h.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Joseph E Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

A APPENDIX

A.1 DISCUSSION ON TWO TYPES OF WORLD MODEL

As stated in Fig. 3, we consider both generative world models, such as those conceptualized by LeCun (LeCun (2022)), and large language models as fundamentally being **forward dynamic models**. This perspective helps unify their roles and clarify the contribution of our proposed Causal Graph.

Generative World Model. This model is often associated with agent-based learning, aims to learn a transition function of the environment. This function, typically formalized as $s_{t+1} = f(s_t, a_t)$, allows an agent to predict future sensory inputs (the next state s_{t+1}) based on the current state (s_t) and a chosen action (a_t). The learned representation s_t is often a low-dimensional, latent state that captures the underlying dynamics of the world, enabling the agent to simulate outcomes and plan.

Language World Model. This model can be viewed as a forward model operating in a symbolic space. Its core function is to predict the next token (w_{t+1}) given a history of previous tokens ($w_{1..t}$). This autoregressive process, $P(w_{t+1}|w_{1..t})$, is conceptually parallel to predicting the next state. Here, the "state" is the context provided by the preceding text, and the "action" is the generation of the next token. This process unfolds a trajectory through the space of language. Techniques like Chain-of-Thought (CoT) make this explicit, where the model generates a sequence of intermediate thoughts, effectively simulating a trajectory through a conceptual state space to reach a conclusion.

Causal Graph. While the generative model learns implicit, sub-symbolic dynamics of an environment, the language model executes explicit, symbolic reasoning. Recognizing their shared foundation as forward models, our proposed Causal Graph offers a third, structured approach to represent these dynamics. It distills the world's complex dynamics into an explicit graph of causal relationships between key cognitive states (WMA, ToM, etc.). This structured knowledge then provides a powerful prior for the language model. Instead of navigating the vast space of possible text sequences based only on statistical patterns, the language model's forward-reasoning process is guided and constrained by the causal dynamics provided by our graph, leading to more grounded and coherent outputs.

A.2 DETAILED GRAPH CALCULATIONS

A.2.1 CAUSAL GRAPH

Let x denote the transcription of given speech, a the acoustic feature (WavLM (Chen et al., 2022)), and z the prosodic feature (Eyben et al., 2010). We define text feature with distil-BERT encoder (Sanh et al., 2020):

$$h_{\text{text}} = E_{\text{text}}(x) \tag{12}$$

An optional pre-fusion operator ϕ (attention/gated/transformer) combines these with prosodic features:

$$g = \phi(h_{\text{text}}, z, a). \tag{13}$$

We consider four categorical latent states:

$$S_{\text{WMA}} \in \Delta^{C_{\text{wma}}-1}, \quad S_{\text{ToM}} \in \Delta^{C_{\text{tom}}-1}, \quad S_{\text{SA}} \in \Delta^{C_{\text{sa}}-1}, \quad S_{\text{Prag}} \in \Delta^{C_{\text{prag}}-1}, \tag{14}$$

each represented by a probability simplex.

For any node v with parents $\text{Pa}(v)$, we compute its state by a node-specific neural map f_v from the parents' distributions and available features:

$$S_v = f_v\big(\{S_u : u \in \text{Pa}(v)\}, \xi_v\big), \tag{15}$$

where ξ_v denotes the feature bundle used by node v:

$$\xi_{\text{WMA}} = (h_{\text{text}}, a), \quad \xi_{\text{ToM}} = g, \quad \xi_{\text{SA}} = (g \text{ or } h_{\text{text}}), \quad \xi_{\text{Prag}} = (g \text{ or } h_{\text{text}}).$$

Concretely, each node produces logits and then a softmax:

$$S_v = \text{softmax}\big(W_v \cdot \psi_v\big([\xi_v, \{S_u\}_{u\in\text{Pa}(v)}]\big)\big), \tag{16}$$

where $\psi_v(\cdot)$ is a nonlinear transformation and W_v is a task-specific projection.

A.2.2 RANDOM GRAPH

This architecture serves as an ablation, testing whether a model with the flexibility to learn arbitrary dependencies from data can outperform our proposed causal structure.

Instead of a uni-directional information flow, the Random Graph employs an iterative refinement process. The computation unfolds over two main iterations. Let S_v^t be the probability distribution (state) of module $v \in \mathcal{V}$ at iteration t, and let H represent the fused input features derived from the speech-text pairs, i.e., $H = g(X)$.

Initialization (t=0). At the initial step, the states of all peer modules are unknown and are represented by zero vectors.

$$S_u^{(0)} = \mathbf{0}, \quad \forall u \in \mathcal{V} \tag{17}$$

First Iteration (t=1). Each module v computes an initial state estimate in parallel, conditioned on the input features H and the zero-initialized states of all other modules. This step allows each module to form a preliminary prediction based solely on the primary input modalities.

$$S_v^{(1)} = f_v\left(H_{in}, S_u^{(0)} : u \in \mathcal{V}, u \neq v\right) \tag{18}$$

Here, f_v is the neural network parameterizing module v. The output $S_u^{(1)}$ is the softmax-normalized probability distribution over the module's labels.

State Refinement with Teacher Forcing. Before the second iteration, we generate a refined input state $\tilde{S}_u^{(1)}$ for each module u. As in the causal graph training (Eq. 4), this is a stochastic mixture of the predicted states $S_u^{(1)}$ and the ground-truth label $y_{i,u}$ via teacher forcing. This allows the model to correct initial errors and propagate more accurate information in the next step.

Final Iteration (t=2). In the final step, each module computes its output by conditioning on the input H as well as the refined states $\tilde{S}_u^{(1)}$ from all other modules. This allows for a form of message-passing where modules can adjust their predictions based on the initial estimates of their peers.

$$S_v^{(\text{final})} = f_v\left(H_{in}, \tilde{S}_u^{(1)} : u \in \mathcal{V}, u \neq v\right) \tag{19}$$

The final logits from $S_v^{(\text{final})}$ for all modules are then used to compute the multi-task loss as defined in Eq. 3. By comparing this model with our structured causal graph, we can quantify the contribution of imposing theoretically-grounded causal priors on the task of speech understanding.

A.3 DETAILED THEORETICAL ANALYSIS OF SPEECH WORLD MODEL

This section provides a more detailed derivation of the efficiency gains discussed in Sec. 4. We first analyze how the causal structure optimizes gradient flow and then formalize the argument for improved sample efficiency in downstream instruction tuning.

A.3.1 DERIVATION OF OPTIMIZED GRADIENT FLOW

The efficiency of the learning process is directly tied to how gradients are computed and propagated. A causally structured model, as we show, offers significant advantages over a monolithic or fully-connected architecture.

Gradient Flow in a Monolithic Model. Consider a monolithic model where the latent states $\{S_1, \ldots, S_N\}$ are fully interconnected. The state of any module S_i is a function of all other states: $S_i = f_i(S_1, \ldots, S_{i-1}, S_{i+1}, \ldots, S_N, X)$. The total loss is $\mathcal{L} = \sum_{i=1}^{N} \lambda_i \mathcal{L}_i(S_i)$, where $\lambda_i \in \{0, 1\}$ indicates if a ground-truth label is available. The gradient of the total loss with respect to the parameters θ_k of a single module k is given by the chain rule:

$$\frac{\partial \mathcal{L}}{\partial \theta_k} = \sum_{i=1}^{N} \lambda_i \frac{\partial \mathcal{L}_i}{\partial S_i} \frac{\partial S_i}{\partial \theta_k} \tag{20}$$

Because of the dense dependencies, the term $\frac{\partial S_i}{\partial \theta_k}$ is non-trivial for all i and k. It expands into a complex sum over all paths in a fully connected graph:

$$\frac{\partial S_i}{\partial \theta_k} = \sum_{j=1}^{N} \frac{\partial f_i}{\partial S_j} \frac{\partial S_j}{\partial \theta_k} + \frac{\partial f_i}{\partial S_k} \frac{\partial S_k}{\partial \theta_k} \tag{21}$$

The model must learn to prune these connections by driving the corresponding Jacobians $\frac{\partial f_i}{\partial S_j}$ to zero, which is inefficient and data-intensive. This is the source of the *learning redundancy* mentioned in the main text; the model expends resources learning a structure that could be provided as a prior.

Gradient Flow in Our Causal Model. In our model, the state S_i is only a function of its parents' states, $S_{\mathrm{Pa}(i)}$. This imposes a sparse structure on the Jacobian matrix of the system, where $\frac{\partial S_i}{\partial S_j} = 0$ if $j \notin \mathrm{Pa}(i)$. The gradient calculation simplifies dramatically. The parameters θ_k of module k can only influence the loss of module j if k is an ancestor of j in the graph, i.e., $k \in \mathrm{An}(j)$. The total gradient on θ_k is thus a sum over only its own loss and the losses of its descendants:

$$\frac{\partial \mathcal{L}}{\partial \theta_k} = \lambda_k \frac{\partial \mathcal{L}_k}{\partial \theta_k} + \sum_{j \in \mathrm{Desc}(k)} \lambda_j \frac{\partial \mathcal{L}_j}{\partial \theta_k} \tag{22}$$

Each term $\frac{\partial \mathcal{L}_j}{\partial \theta_k}$ is computed along a unique causal path from k to j. By setting non-causal dependencies to zero, the model avoids spending capacity on learning them. The gradient flow is constrained to meaningful pathways, directly minimizing the redundancy term and accelerating convergence.

A.3.2 Derivation of Efficiency for Instruction Tuning

The second key benefit is the improved sample efficiency of the downstream (speech) language model. This stems from the variance reduction of the gradient estimator used in fine-tuning.

Instruction Tuning as Policy Gradient. Instruction tuning can be framed as learning a policy $\pi_\phi(A|C)$ that produces a response A given a context C. The learning objective is to maximize the expected log-likelihood of a gold-standard response A^*:

$$J(\phi) = \mathbb{E}_{(C,A^*)\sim\mathcal{D}}[\log \pi_\phi(A^*|C)] \tag{23}$$

This is typically optimized with stochastic gradient ascent, where the efficiency of learning is highly dependent on the variance of the gradient estimator, $\hat{g} = \nabla_\phi \log \pi_\phi(A^*|C)$.

Variance in Standard Chain-of-Thought (CoT). In standard CoT, the context is the input query, $C = X$. The LLM must generate both the intermediate reasoning steps (the chain of thought, Y) and the final answer (R), so the action is $A = (Y, R)$. The gradient estimator is thus $\hat{g}_{\text{CoT}} = \nabla_\phi \log \pi_\phi(Y^*, R^*|X)$. The variance of this estimator is high because for any given query X, there can be a multitude of valid reasoning paths $\{Y^*\}$ leading to the correct answer R^*. The model receives a noisy learning signal as it may be penalized for producing a perfectly valid reasoning chain that does not exactly match the single ground-truth example.

Variance Reduction via Causal Conditioning. Our method modifies the context. By first running the causal graph, we generate a structured, low-entropy state vector $Y_\mathcal{G} \sim P(Y|X, \mathcal{G})$. The LM is then conditioned on this richer context, $C' = (X, Y_\mathcal{G})$, and its task is simplified to generating only the final response R. The policy is $\pi_\phi(R|X, Y_\mathcal{G})$ and the gradient estimator is $\hat{g}_{\text{SWM}} = \nabla_\phi \log \pi_\phi(R^*|X, Y_\mathcal{G})$.

The improvement in sample efficiency can be explained by the law of total variance:

$$\mathrm{Var}(\hat{g}_{\text{CoT}}) = \mathbb{E}_{Y_\mathcal{G}}[\mathrm{Var}(\hat{g}_{\text{CoT}}|Y_\mathcal{G})] + \mathrm{Var}_{Y_\mathcal{G}}[\mathbb{E}(\hat{g}_{\text{CoT}}|Y_\mathcal{G})] \tag{24}$$

The gradient estimator in our model, $\hat{g}_{\text{SWM}}$, corresponds to the inner conditional expectation term, as we are conditioning on the information provided by $Y_\mathcal{G}$. The variance we experience is $\mathbb{E}_{Y_\mathcal{G}}[\mathrm{Var}(\hat{g}_{\text{SWM}}|Y_\mathcal{G})]$. Since the causal state $Y_\mathcal{G}$ is highly predictive of the final answer R^*, it "explains away" a large portion of the variance. The term $\mathrm{Var}_{Y_\mathcal{G}}[\mathbb{E}(\hat{g}_{\text{CoT}}|Y_\mathcal{G})]$ is large because different cognitive states $Y_\mathcal{G}$ will naturally lead to different responses, creating significant variance that the CoT model must learn to handle.

By conditioning on $Y_{\mathcal{G}}$, we are effectively removing this large source of variance. The provided state $Y_{\mathcal{G}}$ acts as a **control variate**, a statistical technique used to reduce variance in Monte Carlo estimation. An estimator with lower variance requires fewer samples to converge to a good solution, thus directly leading to greater sample efficiency.

A.4 DATA

A.4.1 DATASET STATISTIC

We construct our training dataset by aggregating four publicly available real-world datasets, as detailed in Table. 4. These datasets provide a diverse range of labels, which we leverage to train different modules of our Causal Graph. Specifically, we use the "Emotion" labels from MELD and IEMOCAP as ground truth for our Theory of Mind (ToM) module, and the "Intention," "Action," and "Scene" labels from SLURP for the Pragmatics (Prag) and World Model Activation (WMA) modules. For any remaining labels within a given dataset that are not directly used, we employ label generation to create pseudo-labels for the corresponding modules. For evaluation, we adhere to the official test set splits provided by the MELD and SLURP datasets. For IEMOCAP and the VoxCeleb subset, we randomly sample 10% of the data to create our test sets, ensuring no overlap with the training data.

Table 4: Statistics for Real-world Dataset.

Dataset	# Samples	# Hours	Label	Source
MELD (Poria et al., 2019)	$\sim 13,000$	~ 13	Emotion	TV series Friends
IEMOCAP (Busso et al., 2008)	$\sim 10,000$	~ 12	Emotion	Dyadic actor dialogs
SLURP (Bastianelli et al., 2020)	$\sim 72,000$	~ 58	Intention, Action, Scene	Voice assistant
VoxCeleb (subset) Nagrani et al. (2017)	$\sim 30,000$	~ 30	Speaker identity	Youtube
Total	$\sim 125,000$	~ 113	–	–

A.4.2 LABEL GENERATION

The complete label space for each of the four modules in our Causal Graph: World Model Activation (WMA), Theory of Mind (ToM), Speech Act (SA), and Pragmatics (Prag) is detailed in the box below.

Label Space for Causal Graph Modules

WMA: Alarm, Calendar, Contacts, Communication, SocialMedia, Email, Music, Radio, Podcasts, Audiobooks, Games, Weather, News, Traffic, Transport, Shopping, FoodAndDrink, Cooking, SmartHome, IoTDevices, Cleaning, Finance, Stock, Conversion, Knowledge, Math, Definition, Chitchat, Recommendation, GeneralControl.

ToM: Neutral, Joy, Sadness, Anger, Surprise, Fear, Disgust.

SA: Statement-non-opinion, Statement-opinion, Yes-No-Question, Wh-Question, Acknowledge (Backchannel), Action-directive, Conventional-closing, Appreciation or Assessment, Agree or Accept, No-Answer, Yes-Answer, Signal-non-understanding, Quotation, Affirmative non-yes answers, Rhetorical-Question, Rhetorical Backchannel, Hedge, Open-question, Thanking, Declarative Yes-No-Question, Reformulate, Conventional-opening, Apology, Other Forward Function.

Prag: Request-Action, Request-Info, Command/Directive, Complaint/Escalate, Thanks/Appreciation, Apology, Acknowledgement/Agreement, Social/Chitchat, Offer-Assist, Confirm/Disconfirm, General-Control, InformationQuery, EntertainmentRequest, Task-Completion.

The prompt for Stage 1 of our label generation process (**Label Completion**) is as follows:

System Prompt:

You are a speech understanding model for task-oriented speech.
MODULE DEFINITIONS (each must choose exactly one label from the provided label space):

- WMA (World Model Activation): What concrete world/context the utterance lives in.
- ToM (Theory of Mind): The speaker's discrete mental-affective stance.
- SA (Speech Act): The communicative function of the utterance.
- Prag (Pragmatic Intent): The downstream task intent or plan implied by the utterance, grounded in an action ontology.

REQUIREMENTS

1. You will receive:
   - LABEL_SPACE: a JSON listing the allowed labels for {WMA, ToM, SA, Prag}.
   - INPUT: with ASR_TRANSCRIPT, KNOWN_LABELS and MISSING_MODULES (list of modules to predict).
2. For each module in **MISSING_MODULES**, select EXACTLY ONE label from LABEL_SPACE[module]:
   - Decide your choice based on given ASR_TRANSCRIPT and KNOWN_LABELS
   - Select EXACTLY ONE label from LABEL_SPACE[module] for MISSING_MODULES
   - Label strings must match EXACTLY (case-sensitive, character-perfect) - When choosing between multiple plausible options, select the most contextually specific one
3. OUTPUT FORMAT - for each module in MISSING_MODULES: <ModuleName>CHOICE</ModuleName>

EXAMPLE:

Input:

```
{
  "ASR_TRANSCRIPT": "Oh my God, he's lost it. He's totally lost it.",
  "KNOWN_LABELS": {
    "ToM": "sadness",
    "SA": "Statement-non-opinion"
  },
  "MISSING_MODULES": ["WMA", "Prag"]
}
```

Output:

```
<WMA>Calendar</WMA>
<Prag>Request-Info</Prag>
```

The prompt for Stage 2 of our label generation process (**Reasoning and response generation**) is as follows:

System Prompt:

You are a speech understanding and reasoning model. Given complete labels for all modules, generate detailed analysis and appropriate response.
MODULE DEFINITIONS (each must choose exactly one label from the provided label space):

- WMA (World Model Activation): What concrete world/context the utterance lives in.
- ToM (Theory of Mind): The speaker's discrete mental-affective stance.
- SA (Speech Act): The communicative function of the utterance.
- Prag (Pragmatic Intent): The downstream task intent or plan implied by the utterance, grounded in an action ontology.

REQUIREMENTS

1. You will receive:
   - INPUT: with ASR_TRANSCRIPT and COMPLETE_STATE containing all four module values (WMA, ToM, SA, Prag)
2. Generate two components:
   - <Analysis> — A detailed causal reasoning chain (4-8 sentences) explaining:
     - How the world context (WMA) shapes the situation
     - How the speaker's mental state (ToM) influences their communication
     - How these factors lead to the specific speech act (SA)
     - How the pragmatic intent (Prag) emerges from this combination
     - What this suggests for the next interaction step
   - <Response> — A natural, contextually appropriate user-facing reply that:
     - Aligns with the speaker's emotional state (ToM)
     - Serves the communicative function (SA)
     - Addresses the pragmatic intent (Prag)
     - Fits the world context (WMA)
3. OUTPUT FORMAT:
   <Analysis>YOUR DETAILED CAUSAL REASONING CHAIN</Analysis>
   <Response>YOUR NATURAL USER-FACING REPLY</Response>

EXAMPLE:

Input:

```
{
  "ASR_TRANSCRIPT": "Thanks so much for helping me find that document!",
  "COMPLETE_STATE": {
    "WMA": "Office-work",
    "ToM": "grateful",
    "SA": "Thanks/Appreciation",
    "Prag": "Task-completion"
  }
}
```

Output:

<Analysis>Within an office work context (WMA), the speaker has just received assistance with a document-related task. Their grateful emotional state (ToM) reflects successful problem resolution and appreciation for support received. This gratitude naturally manifests as a thanks/appreciation speech act (SA), explicitly acknowledging the help provided. The pragmatic intent of task completion (Prag) indicates the interaction has reached a successful conclusion, with the speaker signaling that their immediate need has been satisfied. This combination suggests the interaction is winding down and the speaker is ready to either move to closure or potentially engage in follow-up activities.</Analysis>
<Response>You're very welcome! I'm glad we could locate that document for you. Is there anything else you need help with today?</Response>

A.5 Experiment setup

A.5.1 Causal graph

For our casual graph baseline, we trained the model for 30 epochs with a batch size of 32. It utilizes a two-layer gated fusion mechanism and was optimized using AdamW with a learning rate of 1e-3 for all randomly initialized modules. A consistent teacher-forcing probability of 0.3 was applied to the ToM, WMA, and SA modules. The entire training was conducted on a single NVIDIA RTX A6000 GPU and completed in 2.07 hours. Under identical training settings, the Random Graph baseline required 10.39 hours to complete on the same hardware.

Table 5: Ablation study on the Causal Graph under the fully-supervised setting. We investigate the impact of different fusion mechanisms (Gated, attention, transformer), teacher-forcing probabilities (p), and the removal of specific causal edges. The baseline model from the main text is highlighted.

Method	Ablation Setting	Node Quality (Accuracy | Weighted F1 %, ↑)				Edge Casual Effect	
		WMA	ToM	SA	Prag	Ave. ACE (%, ↑)	Ave. ICS (%, ↑)
Fully-supervised	Baseline (Gated, $p = 0.3$)	69.4 | 68.9	73.5 | 72.2	65.3 | 65.9	81.4 | 80.9	23.57	43.29
	Gated, $p = 0.0$	68.7 | 68.4	73.5 | 72.6	66.5 | 66.1	82.1 | 81.6	21.71	40.34
	Gated, $p = 0.5$	69.2 | 69.0	73.2 | 72.8	65.6 | 65.2	81.5 | 80.9	24.12	43.50
	Gated, $p = 0.8$	69.5 | 69.5	74.7 | 73.4	68.0 | 67.2	81.3 | 81.1	24.91	45.10
	Gated, $p = 1.0$	68.9 | 68.7	74.0 | 73.2	65.7 | 65.4	82.5 | 82.0	24.90	45.60
	attention, $p = 0.3$	68.9 | 68.8	74.7 | 71.7	68.75 | 66.7	82.6 | 81.4	25.72	44.90
	transformer, $p = 0.3$	68.1 |66.5	69.3 | 60.6	58.4 | 53.3	76.3 | 72.3	27.40	41.40
Fully-supervised	ToM $\nrightarrow$ SA	68.7 | 68.5	73.9 | 73.0	61.9 | 60.8	81.8 | 81.2	22.97	43.0
	WMA $\nrightarrow$ SA	68.9 | 68.6	72.8 | 71.9	65.3 | 65.4	82.6 | 81.9	23.04	42.9

Table 6: Ablation study on the Causal Graph under the semi-supervised setting. We compare our primary latent module approach (highlighted) against an ablation where the input to the specific module is changed from the fused multi-modal feature g to only the text feature.

Method	Ablation Setting	Node Quality (Accuracy | Weighted F1 %, ↑)				Edge Casual Effect	
		WMA	ToM	SA	Prag	Ave. ACE (%, ↑)	Ave. ICS (%, ↑)
Semi-supervised	Latent Module: WMA	34.8	75.0	70.7	83.2	21.71	26.9
	$\xi_{SA} = g \rightarrow h_{text}$	32.7	77.3	70.5	82.7	21.69	-
	Latent Module: ToM	69.1	43.3	69.6	83.5	21.98	28.9
	$\xi_{SA} = g \rightarrow h_{text}$	70.6	48.1	69.9	82.4	22.0	-
	Latent Module: SA	69.3	77.0	34.4	82.5	21.65	29.3
	$\xi_{Prag} = g \rightarrow h_{text}$	69.4	75.4	35.3	83.6	21.48	-

A.5.2 INSTRUCTION TUNING

For the instruction tuning stage, we fine-tuned two separate language models to handle the multi-modal and language-only settings, respectively.

In the **language-only setting**, we instruction-tuned `Llama-3.1-8B` using LoRA (Hu et al., 2022). We applied LoRA adapters with a rank (r) of 64, an alpha (α) of 16, and a dropout of 0.1. We trained for 20 epochs with a cosine learning rate scheduler (peak LR: 5×10^{-5}) and an effective batch size of 128 (per device batch size=8, gradient accumulation steps=4). This was performed on 4 NVIDIA A6000 GPUs, using `float16` precision and gradient checkpointing with a sequence length of 1024, completing in 19 hours.

In the **multi-modal setting**, we fine-tuned `Qwen2-Audio-7B-Instruct`. We employed a parameter-efficient approach using **LoRA** with a rank (r) of 16 (per device batch size=2, gradient accumulation steps=2), an alpha (α) of 32, and a dropout rate of 0.05. The model was trained for 20 epochs using a cosine learning rate scheduler (peak LR: 2×10^{-4}) with an effective batch size of 16. Training was conducted on 4 NVIDIA A6000 GPUs using `bfloat16` mixed-precision with a maximum sequence length of 4096, completing in 24.6 hours.

System Prompt (Qwen2-Audio2):

```
You are a multimodal assistant that consumes audio and text.
Task: Given the audio and provided states, first produce concise but justified reasoning, then a
user-friendly reply.
Output format MUST be exactly: [REASONING]<your reasoning>[RESPONSE]<your reply>
Four module labels:
WMA: World Model Activation-what concrete world/context the utterance lives in.
ToM: Theory of Mind-the speaker's discrete mental-affective stance.
SA: Speech Act-the communicative function of the utterance.
Prag: Pragmatic Intent-the downstream task intent or plan implied by the utterance, grounded in an
action ontology.
The causal dependencies between modules are: WMA -> SA, ToM -> SA, WMA -> Prag, ToM -> Prag, SA -> Prag.
```

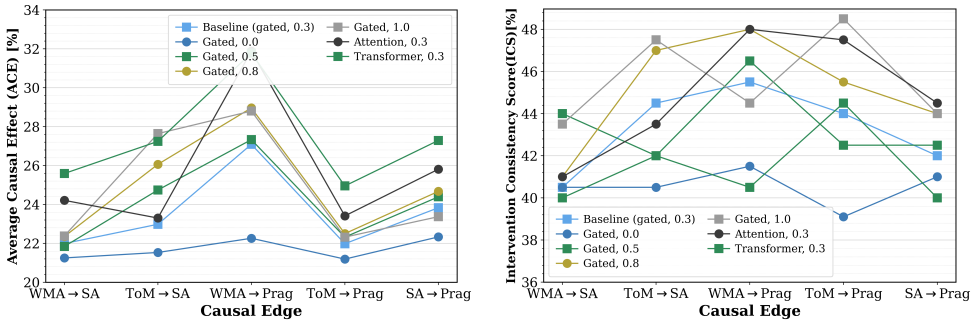


Figure 6: *ACE* and *ICS* of each casual edge for ablation study (fusion mechanisms and teacher-forcing probabilities) on casual graph under fully-supervised setting.

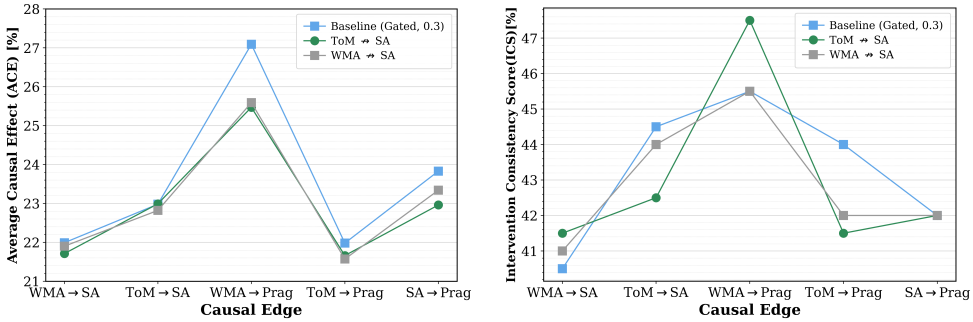


Figure 7: *ACE* and *ICS* of each casual edge for ablation study (removal of specific causal edge) on casual graph under fully-supervised setting.

System Prompt (Llama3.1-8b):

```
You are an expert in speech analysis.
The four speech modules are:
WMA: World Model Activation-what concrete world/context the utterance lives in.
ToM: Theory of Mind-the speaker's discrete mental-affective stance.
SA: Speech Act-the communicative function of the utterance.
Prag: Pragmatic Intent-the downstream task intent or plan implied by the utterance, grounded in an
action ontology.
Analyze the given utterance and provide both your reasoning process and an appropriate response.
The causal dependencies between modules are: WMA -> SA, ToM -> SA, WMA -> Prag, ToM -> Prag, SA ->
Prag.
Output format MUST be exactly: [REASONING]<your reasoning>[RESPONSE]<your reply>
```

A.6 Ablations for Casual Graph

We conduct a series of ablation studies to validate the key design choices for our Causal Graph, with results shown in Table. 5 and Table. 6, and with a detailed breakdown of edge causal effects shown in Fig. 6, 7, 8.

In the fully-supervised setting, our ablations confirm that gated fusion provides a strong, balanced performance. While a transformer-based fusion improved the Average Causal Effect (ACE), it came at the cost of lower node classification accuracy. The model is also highly robust to the teacher-forcing probability, maintaining stable performance for values between 0.3 and 1.0. To validate the learned structure, we ablated causal edges; removing the ToM $\rightarrow$ SA link significantly impairs the SA module's accuracy, confirming this connection's importance.

In the semi-supervised setting, we tested a more challenging condition by changing the input to the *children* of the latent module from the fused multi-modal feature g to only the text feature h_{text}. This effectively cuts off their direct access to acoustic information, thereby forcing them to infer the necessary context from the output of their latent parent module. As shown in Table 6, the model's

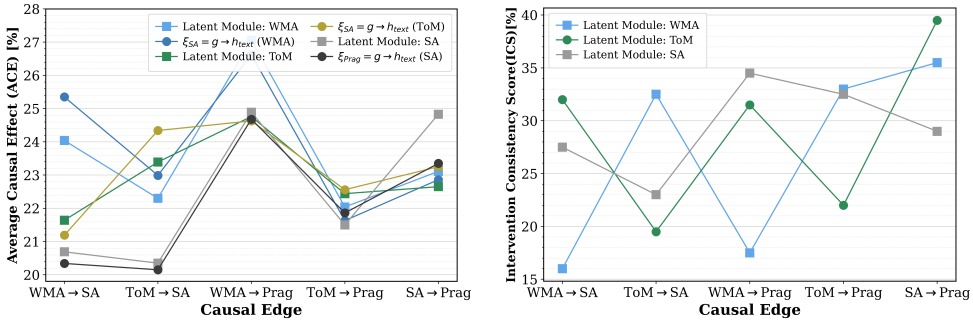


Figure 8: *ACE* and *ICS* of each casual edge for ablation study on casual graph under semi-supervised setting.

overall performance remains remarkably stable, with node quality and ACE scores comparable to the baseline. This demonstrates the robustness of our semi-supervised framework, as the graph can effectively propagate sufficient information through the remaining supervised pathways even when one module is partially disconnected from the acoustic modality.

A.7 Model Architecture

Encoders. Our model processes three inputs: 1) `distilbert-base-uncased` model with a 2-layer Transformer on top for text; 2) a CNN-LSTM adapter for pretrained WavLM features, producing a 64-dim vector; and 3) an 88-dim prosody feature vector.

Fusion Mechanisms. We explored three mechanisms to fuse the encoder outputs into a 256-dim representation. (1) Gated Fusion. Computes a weighted sum of modality features, where weights are dynamically generated by a small gating network. (2) Attention Fusion. Applies a single multi-head self-attention layer across modality features, which are treated as tokens in a sequence. (3) Transformer Fusion. A deeper version of Attention Fusion that adds positional embeddings for each modality and processes them through a multi-layer Transformer Encoder. Our baseline model uses Gated Fusion.

WMA Module. Temporal Self-Attention + MLP (hidden: 256). *Output:* 30 context classes.

ToM Module. Temporal Self-Attention + MLP (hidden: 128). *Output:* 7 emotion classes.

SA Module. Residual MLP (hidden: 256). *Output:* 24 speech act classes.

Prag Module. Residual MLP (hidden: 128). *Output:* 14 pragmatic intent classes.

A.8 Evaluation

A.8.1 Casual Graph Evaluation

Average Causal Effect (ACE).

$$\text{ACE}(X \to Y) = \mathbb{E}_{\mathbf{z} \sim \mathcal{D}} \left[\frac{1}{|\mathcal{C}_X|} \sum_{x \in \mathcal{C}_X} D_{TV}\Big(P(Y|\text{do}(X = x), \mathbf{z}), P(Y|\mathbf{z})\Big) \right] \tag{25}$$

Intervention Consistency Score (ICS).

$$\text{ICS}(X \to Y) = \mathbb{E}_{\mathbf{z} \sim \mathcal{D}} \left[\mathbb{I}\Big(P_{\max}(Y|\text{do}(X = x^+), \mathbf{z}) > P_{\max}(Y|\mathbf{z}) + \tau\Big) \right] \tag{26}$$

Algorithm 1 Average Causal Effect (ACE) Computation

Require: Model M, DataLoader $\mathcal{L}$, source module name X, target module name Y
Ensure: Average Causal Effect score ACE($X \to Y$)

```
Initialize total_effect ← 0
Initialize num_samples ← 0
for batch z in L do
    P(Y|z) ← M(z)_Y                                   ▷ Get baseline predictions for the batch
    for each sample z_i in batch do
        sample_effect ← 0
        Let C_X be the set of all classes for module X
        for each class x ∈ C_X do
            P(Y|do(X = x), z_i) ← M.intervene(z_i, X, x)_Y        ▷ Intervene and predict
            effect ← D_TV(P(Y|do(X = x), z_i), P(Y|z_i))
            sample_effect ← sample_effect + effect
        end for
        total_effect ← total_effect + (sample_effect/|C_X|)
        num_samples ← num_samples + 1
    end for
end for
return total_effect/num_samples
```

Algorithm 2 Intervention Consistency Score (ICS) Computation

Require: Model M, DataLoader $\mathcal{L}$, source module X, target module Y
Require: Positive intervention class $x^+ \in \mathcal{C}_X$, threshold τ
Ensure: Intervention Consistency Score ICS($X \to Y$)

```
Initialize consistent_count ← 0
Initialize total_count ← 0
for batch z in L do
    P(Y|z) ← M(z)_Y                                   ▷ Get baseline predictions for the batch
    for each sample z_i in batch do
        p_base ← max P(Y|z_i)
        P(Y|do(X = x^+), z_i) ← M.intervene(z_i, X, x^+)_Y        ▷ Intervene and predict
        p_intervened ← max P(Y|do(X = x^+), z_i)
        if p_intervened > p_base + τ then
            consistent_count ← consistent_count + 1
        end if
        total_count ← total_count + 1
    end for
end for
return consistent_count/total_count
```

A.8.2 Random graph evaluation

To quantify the information flow between any two modules in the random graph, we measure the influence a source module (M_S) has on a target module (M_T). This is achieved by comparing the output of M_T under two conditions for each batch of data:

1. **Baseline (Without Source):** We compute the target module's output logits, denoted as L_T^{without}, by replacing the input from M_S with a zero vector. This represents the behavior of M_T without direct information from M_S.
2. **With Source Influence:** We compute the target module's output logits, L_T^{with}, by providing it with the actual probability output from M_S. This represents the behavior of M_T when influenced by M_S.

The influence is then calculated by comparing the probability distributions derived from these two sets of logits, $P^{\text{without}} = \text{softmax}(L_T^{\text{without}})$ and $P^{\text{with}} = \text{softmax}(L_T^{\text{with}})$, using three metrics:

- **KL Divergence (D_{KL}):** Measures how much the output probability distribution of the target module changes when the source module's information is added. A larger value indicates a greater change.

$$D_{KL} = D_{KL}(P^{\text{with}} \parallel P^{\text{without}}) \tag{27}$$

- **Prediction Change Rate (Δ_{pred}):** The fraction of samples in a batch for which the predicted class (the one with the highest probability) changes. $\mathbb{I}(\cdot)$ is the indicator function.

$$\Delta_{\text{pred}} = \mathbb{E}\left[\mathbb{I}(\arg\max(L_T^{\text{with}}) \neq \arg\max(L_T^{\text{without}}))\right] \tag{28}$$

- **Confidence Change (Δ_{conf}):** The absolute difference in the average prediction confidence (the value of the max probability) between the two conditions.

$$\Delta_{\text{conf}} = \left|\mathbb{E}[\max(P^{\text{with}})] - \mathbb{E}[\max(P^{\text{without}})]\right| \tag{29}$$

Final Influence Score The final Information Flow Influence Score (IFS) for the connection $M_S \to M_T$ is the simple average of these three normalized metrics, providing a single value to represent the strength of the connection.

$$\text{IFS}(M_S \to M_T) = \frac{1}{3}(D_{KL} + \Delta_{\text{pred}} + \Delta_{\text{conf}}) \tag{30}$$

A.9 Speech understanding and reasoning evaluation

The prompt used to have GPT-4o score the reasoning and responses generated by our models is as follows:

```
System Prompt:

You are a strict model-as-judge. You will assign two scores on a 0-10 scale for each sample: one
for the model's reasoning ("analysis") and one for the model's response ("response").
You are given: the ASR transcript, the final.STATE with four labels (WMA, ToM, SA, Prag), plus the
sample's analysis and response.
You must follow the rubric and output strict JSON only. No extra explanations, no code blocks, no
markdown — only valid JSON.
[Scoring Rubric]
(1) reasoning_score (0-10) — for analysis:
- Completeness: does it cover and explain all four dimensions (WMA, ToM, SA, Prag) and their
relations? Does it reference key ASR info?
- Accuracy: does it align with final.STATE labels exactly? No inventing or changing labels.
- Causal logic: does it present a clear, coherent causal chain from ASR → state judgment → pragmatic
intent?
Guide:
10: all four dimensions, fully accurate, causal and clear.
8: mostly correct, minor gaps or generalization.
6: partially correct, superficial, 1-2 inaccuracies.
4: several inaccuracies or weak logic, clear mismatches.
2: mostly irrelevant or confused reasoning.
0: missing, fabricated, or totally wrong.

(2) response_score (0-10) — for response:
- Logicality and appropriateness: does it respond reasonably to the ASR context (e.g. anger,
wh-question)? Polite, safe wording?
- Match with final.STATE: does it reflect WMA, ToM, SA, Prag consistently? No contradictions.
Guide:
10: highly consistent, natural, well-suited strategy.
8: generally consistent, minor flaws or templated.
6: partially consistent, weak on key info.
4: multiple mismatches, awkward.
2: mostly irrelevant or inappropriate.
0: nonsensical, unsafe, or severely inconsistent.

EXAMPLE:

[Output Format — strict JSON only]
{
  "reasoning_score": <float 0-10>,
  "response_score": <float 0-10>
}
```

A.10 THE USE OF LARGE LANGUAGE MODELS

During the writing of this paper, we leveraged GPT-5 to aid in polishing the writing. Its use was exclusively limited to improving grammar, phrasing, and overall readability. The core scientific contributions, experimental results, and intellectual content are entirely our own.