SCIR: Learning Speech-based Conversational Interaction Representations from Continuous Acoustic Signals

Anonymous ACL submission

Abstract

Real-time spoken dialogue systems demand precise, low-latency decisions on when to speak, listen, or yield—a challenge intensified in full-duplex settings characterized by speech overlap and competitive turn-taking. While emerging end-to-end Speech LLMs offer low latency, they often lack explicit controllability and robustness, whereas traditional cascade systems suffer from unavoidable processing delays due to ASR and generation. This work investigates the learning of conversational interaction representations directly from continuous acoustic signals to bridge this gap. We propose **SCIR**, a task-driven representation learned end-to-end, which unifies interaction timing decisions—including turn-taking, backchanneling, and barge-in—under a single streaming-compatible framework **via explicit multi-task learning**, without relying on textual inputs. Through extensive experiments, we demonstrate that lightweight SCIR models not only surpass large-scale, general-purpose speech baselines in predictive performance but do so with orders-of-magnitude lower latency and parameter efficiency. Crucially, we show that SCIR's anticipatory nature provides a "negative latency" buffer that effectively masks the computational overhead of cascade ASR and LLM pipelines. This establishes SCIR as a robust, **plug-and-play**, and intelligence-preserving solution for next-generation real-time dialogue agents.

1 Introduction

Real-time spoken dialogue systems must continuously decide *when* to speak, listen, or yield during interaction (Sacks et al., 1974; Stivers et al., 2009). Such timing decisions are fundamental to conversational fluency: speaking too early causes intrusive interruptions, speaking too late makes responses feel sluggish, and failing to yield during user interruptions can result in prolonged overlap and degraded user experience (Levinson, 2016).

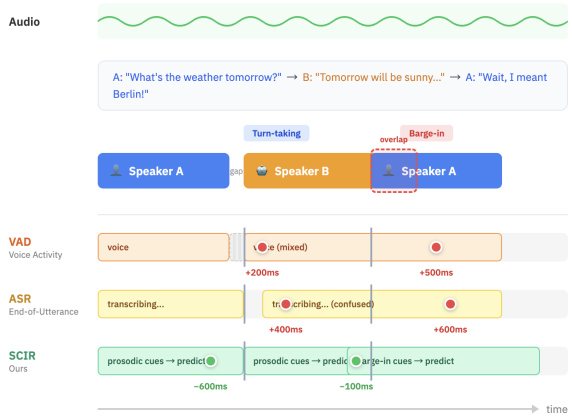

Figure 1: Conceptual comparison of interaction timing pipelines. VAD/ASR pipelines react to turn boundaries and interruptions after they occur, incurring latency and overlap. SCIR instead anticipates turn-taking and barge-in intents from continuous acoustic cues, providing a "negative latency" buffer that enables timely responses and early yielding.

Unlike offline speech or dialogue tasks, interaction timing decisions are inherently *incremental* and must be made under strict latency constraints (Schlangen and Skantze, 2011; Baumann and Schlangen, 2012). These challenges are further amplified in **full-duplex** settings characterized by speech overlap, interruptions, and competitive turn-taking, particularly in multi-party scenarios where speaker changes are rapid and unpredictable (Schegloff, 2000; Shriberg et al., 2001).

Current approaches to interaction management generally fall into two categories: cascade systems and end-to-end Speech LLMs. Cascade systems, which rely on textual representations from Automatic Speech Recognition (ASR), are fundamentally misaligned with the requirements of real-time interaction (Raux and Eskenazi, 2008; DeVault et al., 2011). As illustrated in Figure 1(a), ASR-based pipelines introduce unavoidable latency due to buffering, decoding, and post-processing. More fundamentally, identifying a turn-completion or

an imminent barge-in often relies on fine-grained prosodic and temporal cues (e.g., pitch glides, rhythmic cadence) that are erased during discretization into text tokens (Ward and Tsukahara, 2000; Gravano and Hirschberg, 2011; Duncan, 1972). While emerging end-to-end Speech LLMs, such as GPT-4o (OpenAI, 2024) and Moshi (Défossez et al., 2024), achieve impressive low latency by integrating listening and speaking into a single model (Ma et al., 2024; Nguyen et al., 2025), they face significant challenges in deployability and an inherent **latency-intelligence trade-off**: they are computationally expensive, lack explicit controllability (coupling interaction logic tightly with generation), and often suffer from robustness issues such as hallucinating during long silences (Zhang et al., 2025a). Recent benchmarks have also highlighted the difficulty these models face in handling complex turn-taking dynamics (Lin et al., 2025; Peng et al., 2025; Zhang et al., 2025b).

This work proposes a third path: a specialized, lightweight interaction layer that complements cascade systems to achieve end-to-end-like latency without compromising modularity or intelligence. We introduce **SCIR** (**S**peech-based **C**onversational **I**nteraction **R**epresentation), a task-driven representation learned end-to-end from low-level acoustic features. Unlike reactive VADs, SCIR is *anticipatory*. By detecting acoustic precursors to turn transitions (such as intonation drops or breath intakes), SCIR provides a "negative latency" buffer—predicting a turn end hundreds of milliseconds before it physically completes. This approach shares the predictive spirit of Voice Activity Projection (VAP) (Ekstedt and Skantze, 2022; Inoue et al., 2024a) but unifies multiple interaction decisions into a single lightweight framework **that explicitly disentangles communicative intents (e.g., distinguishing a backchannel from an interruption), offering finer-grained control than generic voice activity projection.** As shown in Figure 1(b), this look-ahead capability effectively masks the computational latency of the downstream ASR and LLM modules.

We conduct extensive experiments on challenging multi-party meeting datasets to validate this approach. Our results highlight the efficacy of SCIR:

- **Efficiency-Performance Pareto:** SCIR achieves state-of-the-art performance on interaction timing tasks, matches large-scale baselines (e.g., Wav2Vec 2.0 (Baevski et al., 2020)) while using $8\times$ fewer parameters.
- **Offsetting System Latency:** We demonstrate that SCIR's prediction lead time ($\sim$600ms) is sufficient to offset typical ASR/LLM processing delays, enabling fluid turn-taking even in modular architectures.
- **Robustness and Applicability:** Unlike fragile end-to-end models, SCIR's specialized focus ensures robust performance across diverse domains, **while also revealing fundamental acoustic divergences between tonal and intonational languages regarding turn-taking cues.**

2 Related Work

This work relates to several lines of research in spoken dialogue and speech processing, positioning SCIR as a distinct approach that bridges the gap between traditional signal processing and modern generative modeling.

2.1 Interaction Timing in Conversation

Interaction timing has been a central topic in conversation analysis, establishing the theoretical basis for predictive modeling. Foundational work characterizes turn-taking as a structured yet locally managed process, where speakers coordinate turn exchanges through a combination of linguistic, prosodic, and temporal cues (Sacks et al., 1974; Duncan, 1972). Subsequent studies have confirmed that conversational flow depends critically on the ability to *anticipate* turn completion rather than reacting after utterances end (Levinson, 2016). Specific phenomena such as backchanneling have also been extensively studied for their prosodic cues (Ward and Tsukahara, 2000; Yngve, 1970), while overlap and interruptions remain critical for natural dialogue flow (Schegloff, 2000; Gravano and Hirschberg, 2011; Shriberg et al., 2001).

In computational settings, interaction timing has been operationalized through tasks such as end-of-turn detection and next-speaker prediction. Early approaches utilized incremental processing frameworks (Schlangen and Skantze, 2011; Baumann and Schlangen, 2012; DeVault et al., 2011; Raux and Eskenazi, 2008). More recently, continuous predictive models have emerged, such as LSTM-based approaches (Skantze, 2017; Roddy et al., 2018) and Voice Activity Projection (VAP) (Ekstedt and Skantze, 2022), which projects future voice

activity directly from speech. This paradigm has been extended to real-time settings (Inoue et al., 2024a), multilingual contexts (Inoue et al., 2024b), and specific backchannel prediction (Inoue et al., 2025). **While VAP implicitly models turn-taking through a unified projection task, it does not explicitly disentangle distinct communicative intents (e.g., backchannel vs. barge-in). SCIR extends this by adopting a multi-task learning framework to provide fine-grained, intent-aware interaction signals.**

However, most prior approaches rely heavily on lexical features, inheriting the unavoidable latency of ASR systems. SCIR departs from this by proving that acoustic-only representations are sufficient—and often superior—for the specific subproblem of timing. Crucially, by leveraging continuous acoustic cues, SCIR achieves the "negative latency" required to offset the processing delays inherent in cascade dialogue systems. Recent benchmarking efforts have also systematically evaluated audio foundation models on turn-taking dynamics (Arora et al., 2025), further highlighting the importance of this capability for speech systems.

2.2 Speech LLMs and Full-Duplex Systems

Recent end-to-end Speech LLMs represent the frontier of low-latency interaction (Veluri et al., 2024). Models that integrate understanding and generation into a single transformer implicitly model turn-taking through next-token prediction, achieving impressive responsiveness (Défossez et al., 2024; OpenAI, 2024). Recent works have pushed the boundaries of streaming speech interaction (Xie and Wu, 2024; Fang et al., 2024) and dual-channel modeling (Wang et al., 2025; Nguyen et al., 2025). While effective, this monolithic approach presents significant challenges for robust deployment. First, they are computationally expensive and difficult to host on edge devices. Second, they often lack explicit controllability; coupling interaction logic tightly with generation makes it difficult to force the model to "listen" or "yield" without modifying the generative weights. Third, they risk hallucination during long silences or background noise (Zhang et al., 2025a; Ma et al., 2024). SCIR offers a modular alternative. By treating interaction timing as a separate, lightweight control signal, SCIR can be integrated into modular architectures to provide explicit, robust timing control without the opacity and cost of a single giant model.

2.3 Speech Representation Learning

Self-supervised speech representation learning (SSL) has produced powerful general-purpose embeddings. Models such as Wav2Vec 2.0 (Baevski et al., 2020), HuBERT (Hsu et al., 2021), and WavLM (Chen et al., 2022) are typically trained with objectives emphasizing phonetic or lexical content recovery. Weakly supervised models like Whisper (Radford et al., 2023) also provide robust features. Despite their expressiveness, these representations are not explicitly optimized for interaction dynamics. Their inductive biases prioritize content over prosodic timing, and their computational footprint (often 90M+ parameters) poses significant challenges for the strict real-time loops required for interaction control (often operating at 10ms ticks). Our experiments compare SCIR against frozen SSL backbones to demonstrate that a small, task-specific model trained from scratch can outperform massive general-purpose models on timing tasks. This validates our hypothesis that a specialized inductive bias is more efficient than generic pre-training for the specific task of real-time interaction management.

3 Method

We present **SCIR** (Speech-based Conversational Interaction Representation), a unified representation learning framework designed to serve as the low-latency "nervous system" of a spoken dialogue agent. As illustrated in Figure 2, SCIR processes continuous acoustic streams to produce a shared, causal embedding that simultaneously supports turn-taking, backchannel, and barge-in predictions.

3.1 Problem Formulation

We model conversational interaction timing as a continuous-time sequence labeling problem. Let $\mathbf{x}_t$ denote the acoustic feature vector at time t. The goal is to learn a causal mapping $f_\theta : \mathbf{X}_t \rightarrow \mathbf{h}_t$, where $\mathbf{h}_t$ is a latent interaction representation. This representation supports a set of binary classification tasks $\mathcal{T} = \{\text{turn-taking}, \text{backchannel}, \text{barge-in}\}$. **We hypothesize that these tasks share underlying acoustic precursors. By jointly modeling them, the shared representation $\mathbf{h}_t$ can leverage cross-task transfer (e.g., barge-in cues informing turn-taking predictions), leading to more robust generalization than single-task models.**

Crucially, unlike Voice Activity Detection (VAD) which detects *current* voice activity, SCIR

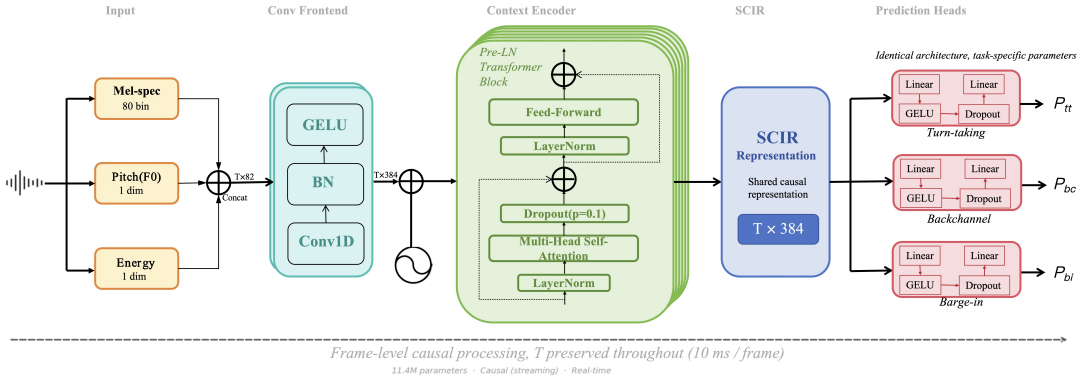

Figure 2: The overview of the proposed SCIR framework. (Left) **Acoustic Input**: We extract low-level features (Mel-spectrogram, Pitch, Energy) from the continuous audio stream. (Middle) **Shared Encoder**: A lightweight convolutional frontend processes local temporal patterns, followed by a Causal Transformer that captures long-range interaction dynamics. (Right) **Multi-task Decoders**: Lightweight heads project the shared representation $\mathbf{h}_t$ to task-specific probabilities.

predicts *future* interaction events. For instance, a positive "Turn-Taking" label at time t does not merely indicate silence, but signifies that the current speaker *intends* to yield the floor within a short prediction window (Skantze, 2017; Ekstedt and Skantze, 2022).

3.2 Data Construction and Labeling

A key challenge in interaction modeling is constructing dense, frame-level supervision from sparse event annotations. We derive our training targets from the AMI Meeting Corpus (Carletta, 2007), which contains multi-party audio with time-aligned transcriptions. The labeling logic is defined as follows:

- **Turn-Taking:** We define a turn-taking event as a speaker transition point, corresponding to the onset of the next speaker's utterance. Positive labels are assigned to frames within a temporal window centered around this transition onset.
- **Backchannel:** Backchannel events are heuristically defined as short utterances (shorter than 1 second) produced by a non-current speaker.
- **Barge-in:** A barge-in event is defined as an overlap between the onset of a speaker's utterance and another speaker's ongoing speech.

3.3 SCIR Architecture

The model architecture is purposefully designed for extreme computational efficiency to ensure it can run in parallel with heavier ASR and LLM modules without impacting the system's overall latency budget.

Acoustic Observation Model. We employ a composite feature vector $\mathbf{x}_t$ ($F = 82$) consisting of Log Mel-spectrograms (80-dim), Pitch (F_0), and Energy. Explicitly modeling pitch is critical for this task, as intonation contours (e.g., a rising pitch for questions or a falling pitch for statements) are strong predictors of turn-yielding that are often lost in standard ASR acoustic models (Levinson, 2016; Gravano and Hirschberg, 2011).

Shared Interaction Encoder. The encoder f_θ utilizes a hybrid architecture **meticulously streamlined** for real-time inference. First, a lightweight 1D convolutional frontend aggregates local acoustic context and projects features into the hidden dimension. Second, a Causal Transformer encoder captures long-range temporal dependencies (Vaswani et al., 2017). A strict causal attention mask ensures that predictions at time t depend solely on the history $\mathbf{x}_{\leq t}$, satisfying the hard real-time constraint required for streaming deployment.

Multi-task Decoders. We use simple linear heads to project the shared representation $\mathbf{h}_t$ to task probabilities $p_t^{(k)}$. By keeping the decoders minimal, we force the shared encoder to learn a rich, generalizable representation of interaction dynamics that disentangles the common acoustic precursors of different interaction types.

3.4 Learning Objective

We train the model end-to-end using a weighted binary cross-entropy loss.

To enhance model robustness against acoustic variability and prevent overfitting, we employ data augmentation strategies during training, including SpecAugment (time and frequency masking) and Gaussian noise injection.

We treat the problem as frame-level binary classification. The ground-truth labels $y_t^{(k)} \in \{0, 1\}$ are derived directly from the annotated time boundaries.

$$\mathcal{L} = \sum_{k \in \mathcal{T}} \lambda_k \sum_t \Big[- y_t^{(k)} \log p_t^{(k)} - (1 - y_t^{(k)}) \log(1 - p_t^{(k)}) \Big] \quad (1)$$

where λ_k balances the contribution of each task, addressing the natural class imbalance where interaction events are sparse compared to background speech.

3.5 Inference and Decision Making

During inference, SCIR operates in a streaming fashion. The continuous probability stream $p_t^{(k)}$ acts as a real-time "affordance" signal for the dialogue manager. A discrete decision is triggered when the probability exceeds a task-specific threshold τ_k. These thresholds are adjustable operating points, allowing system designers to trade off between *responsiveness* (minimizing latency) and *safety* (minimizing interruptions) based on the application context, without the need to retrain the model (Davis and Goadrich, 2006).

4 Evaluation and Metrics

Evaluating conversational interaction timing poses unique challenges due to class imbalance, annotation ambiguity, and the tight coupling between decision latency and interaction quality. In this section, we formalize our evaluation protocol, spanning standard event-level metrics, latency-aware decision metrics, and system-level simulation benchmarks.

4.1 Event-Level Metrics

We first report standard probabilistic classification metrics. For each interaction task $k \in \mathcal{T}$, the model produces a continuous probability stream $\{p_t^{(k)}\}_{t=1}^{T}$. To assess the discriminative power of the learned representation independent of specific operating points, we primarily report the Area Under the ROC Curve (AUC) and Average Precision (AP). For binary decision metrics (Precision, Recall, F1), we convert probabilities to discrete events using a threshold τ_k. An event is considered correctly detected if the predicted timestamp falls within a temporal tolerance window (e.g., $\pm$500ms) of the ground-truth annotation.

4.2 Latency-Aware Decision Metrics

We define the event time t_{event} as the *turn end* for turn-taking, and as the *speech onset (overlap onset)* for backchannel and barge-in events. Real-time systems must not only detect events but do so with sufficient lead time to plan a response. We explicitly evaluate the temporal characteristics of the model's decisions to quantify its ability to mask downstream latency.

Prediction Lead Time. Unlike offline ASR systems that exhibit lag, an ideal interaction manager should anticipate events. We define the prediction lead time as $\delta = t_{event} - t_{decision}$, where t_{event} denotes the **annotated event boundary** (t_{offset} for turn-taking, t_{onset} for backchannel/barge-in) and $t_{decision}$ is the moment the model's confidence crosses τ_k. A positive δ indicates early prediction (e.g., anticipating a turn-end before silence). We report the mean and distribution of δ to quantify the system's "reflex" speed—essentially measuring the "negative latency" buffer provided by SCIR.

Success@Δ. To characterize the trade-off between timing precision and detection coverage, we use the *Success*@Δ metric. A prediction is considered successful at tolerance Δ if the system triggers a decision within the window $[t_{\text{event}} - \Delta, t_{\text{event}} + \Delta]$.

$$\text{Success@}\Delta = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\left(|t_{\text{pred}}^{(i)} - t_{\text{gt}}^{(i)}| \leq \Delta \right) \quad (2)$$

By varying Δ (e.g., from 200ms to 2000ms), we plot accuracy curves that reveal the temporal granularity of the learned representations. This metric specifically penalizes decisions that are semantically correct but timed poorly (too early or too late).

4.3 System-Level Simulation Metrics

Event-level metrics often fail to capture the dynamic consequences of timing decisions, such as cutting off a user or awkward silences. To bridge this gap, we implement a **Turn-Taking Simulation** involving a 4-party conversational scenario. We define two key system-centric metrics:

Collision Rate (CR). This metric quantifies disruptive interruptions. A collision occurs when the system attempts to take the floor while another speaker is still active or has not yet yielded.

$$\text{CR} = \frac{N_{\text{collision}}}{N_{\text{attempts}}} \times 100\% \tag{3}$$

where N_{attempts} is the total number of turn-taking attempts made by the system. A lower CR indicates smoother, more polite interaction.

Success Rate (SR). Complementary to CR, the Success Rate measures the proportion of attempts that result in a smooth floor transition without overlap or excessive delay (e.g., gap < 1s).

$$\text{SR} = \frac{N_{\text{smooth}}}{N_{\text{attempts}}} \times 100\% \tag{4}$$

These metrics directly reflect user experience and are used to validate the operational utility of SCIR in Figure 5.

4.4 Cross-Domain Calibration Protocol

Interaction timing definitions vary across domains (e.g., meeting vs. chit-chat). To assess robustness, we evaluate performance under two regimes:

- **No Tuning (Zero-shot):** Thresholds τ_k derived from the source domain (AMI) are applied directly to the target domain.
- **Light Tuning:** We allow recalibration of thresholds τ_k using a small sample (e.g., 5 minutes) of target data, without updating model weights (Guo et al., 2017).

This protocol disentangles the generalizability of the learned representation from the domain-specific calibration of decision boundaries.

4.5 Latency Accounting

All reported timing metrics explicitly account for the end-to-end system latency budget:

$$\Delta_{\text{sys}} = \Delta_{\text{feat}} + \Delta_{\text{enc}} + \Delta_{\text{dec}} \tag{5}$$

where Δ_{feat} includes frame stepping (10ms) and windowing, and Δ_{enc} accounts for the causal attention overhead. This ensures our evaluation reflects realistic deployment constraints rather than idealized offline performance.

5 Experiments

We evaluate SCIR across multiple dimensions: predictive performance against state-of-the-art baselines, computational efficiency under real-time constraints, and system-level impact on conversational fluency.

5.1 Experimental Setup

Datasets. We conduct experiments primarily on the **AMI Meeting Corpus** (Carletta, 2007), a multi-party dataset rich in spontaneous speech overlap and competitive turn-taking. Models are trained on the AMI training split (136 meetings) and evaluated on the test split (18 meetings). For cross-domain generalization, we use the **ICSI Meeting Corpus** (Janin et al., 2003) and the Mandarin **AISHELL-4** dataset (Fu et al., 2021).

Baselines. We compare SCIR against a diverse set of baselines representing different paradigms: (1) **Heuristic VAD**: Energy-based voice activity detection, representing standard reactive pipelines. (2) **ASR-based**: Frozen Whisper-base encoder (Radford et al., 2023) representing semantic-heavy pipelines that incur high latency. (3) **SSL Speech Models**: Frozen Wav2Vec 2.0 (Baevski et al., 2020) representing large-scale general-purpose speech representations. (4) **Linear Probe (VAP-style)**: A linear projection trained on frozen acoustic features, mimicking the implicit projection approach of VAP (Ekstedt and Skantze, 2022) but adapted for our explicit multi-task targets. (5) **W2V2 + Temporal Head**: A frozen Wav2Vec 2.0 backbone equipped with a temporal Transformer head identical to SCIR's encoder, testing the upper bound of pre-trained features without fine-tuning.

Implementation Details. We implement SCIR using PyTorch. The model is trained for 100 epochs with a batch size of 16 using the **AdamW** optimizer. We employ a learning rate of $5e^{-4}$ with a cosine decay schedule. For inference, decision thresholds τ_k are tuned on the validation set to maximize the F1 score, resulting in $\tau_{\text{turn}} = 0.65$, $\tau_{\text{bc}} = 0.80$, and $\tau_{\text{barge}} = 0.75$.

5.2 Main Results: Performance vs. Efficiency

Table 1 summarizes the performance on the AMI test set. SCIR achieves an average AUC of **79.3%**. Crucially, it significantly outperforms the VAP-style linear probe (+11.7%), demonstrating that

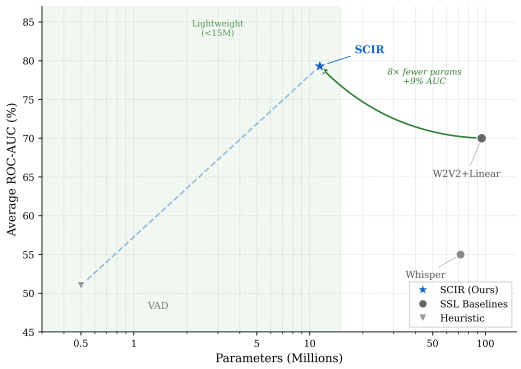

Figure 3: Performance vs. Efficiency. SCIR achieves comparable ROC-AUC to large-scale SSL baselines (Wav2Vec2 + Head) while using orders of magnitude fewer parameters, occupying the optimal Pareto frontier.

Table 1: Main results on AMI Test Set. SCIR outperforms heuristic and VAP-style baselines and matches heavy SSL models with minimal parameters.

Model	Params	Turn	BC	Barge	Avg
Random	-	50.0	50.0	50.0	50.0
Energy VAD	-	43.2	65.3	59.0	55.8
Linear Probe (VAP-style)	0.3M	63.7	79.0	70.3	71.0
Whisper (Frozen)	71.8M	54.5	55.8	54.8	55.0
Wav2Vec 2.0 + Head	99.5M	74.5	82.0	81.6	79.4
SCIR (Ours)	**11.4M**	**71.6**	**87.6**	**78.7**	**79.3**

simple projection is insufficient for decoupling interaction intents and that our temporal modeling is essential. Notably, SCIR matches the performance of the massive **W2V2 + Temporal Head** baseline (79.4%) while using **8.3× fewer parameters** (11.4M vs 95M) and operating at a Real-Time Factor (RTF) of **864×**. We also observed that fully fine-tuning Wav2Vec 2.0 yields a higher AUC of 83.2% (see Appendix), but at the cost of significantly higher memory usage and latency, making it unsuitable for on-device deployment.

5.3 Latency and the "Actionable Window"

For real-time systems, prediction accuracy is meaningless without timeliness. We analyze the temporal behavior of SCIR's predictions to validate our "negative latency" hypothesis.

Early Prediction Capability. As illustrated in Figure 4, SCIR exhibits a clear "Actionable Window." The prediction probability consistently crosses the decision threshold approximately **626ms** before the actual speech onset. This **600ms+ lead time** provides the critical buffer needed to offset cascade system latency. In a standard pipeline where ASR and LLM inference might consume 500-800ms, SCIR allows the system to initiate generation *while the user is still speaking*. This effectively hides the computational cost, rendering the cascade architecture perceptually indistinguishable from a zero-latency end-to-end model.

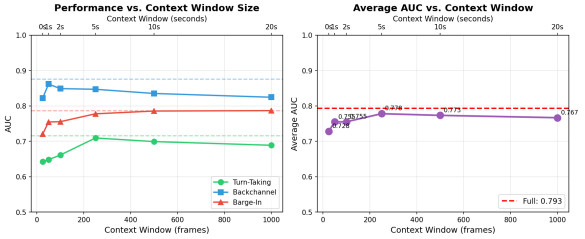

Figure 4: Actionable Window Analysis. The blue curve shows the averaged prediction probability relative to the **event boundary** ($t = 0$). **For turn-taking,** $t = 0$ **corresponds to the turn-end; for backchannel and barge-in, it corresponds to the speech onset.** SCIR crosses the decision threshold 626ms in advance, providing the necessary "negative latency" to offset downstream computation.

5.4 System-Level Simulation

To assess the impact on user experience, we simulated a 4-party conversation and measured the **Collision Rate** (interruptions) and **Success Rate** (smooth turns).

As shown in Figure 5, SCIR drastically improves interaction quality. Compared to the VAD baseline, SCIR reduces the collision rate from 20.7% to **6.3%** (a 69% relative reduction) while increasing the turn-taking success rate to **93.8%**. This confirms that SCIR is not just fast, but "polite." While end-to-end Speech LLMs often struggle with hallucinating speech during quiet moments (Zhang et al., 2025a), SCIR's dedicated training on interaction events ensures it robustly identifies when *not* to speak, solving a key robustness challenge in full-duplex deployment.

5.5 Ablation Studies

We dissect the contribution of each component in Figure 6.

- **Features:** Adding **Pitch** improves performance by +1.22%, confirming the role of prosody in signaling turn boundaries. Interestingly, raw Energy features alone provide negligible gain but stabilize training.
- **Augmentation:** Data augmentation (masking, noise) contributes a substantial +1.25% gain, critical for robustness.

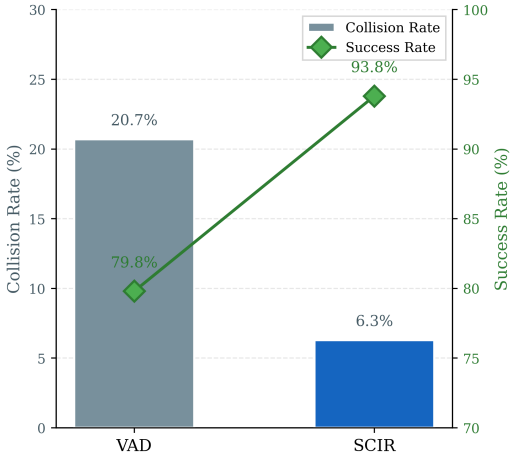

Figure 5: Simulation Results. SCIR significantly reduces the Collision Rate (grey bars) and improves the Success Rate (green line) compared to a standard VAD baseline.

- **Multi-task:** Gaussian noise injection provides modest gains, while aggressive SpecAugment (time/frequency masking) can degrade timing performance by corrupting fine-grained prosodic cues.

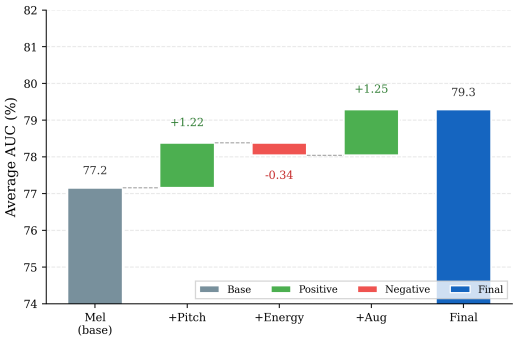

Figure 6: Ablation Waterfall. Shows the cumulative AUC gain from adding Pitch, Energy, and Augmentation strategies to the base Mel-spectrogram model.

5.6 Cross-Domain Generalization

We further evaluate robustness under domain shift. **Zero-shot Transfer:** When applied directly to the Mandarin AISHELL-4 dataset (without fine-tuning), SCIR retains significant predictive power for overlap detection (Retention 40.7%). However, turn-taking retention is lower (19.2%). **Linguistic Divergence:** This performance gap highlights a fundamental linguistic difference: Mandarin is a **tonal language** where pitch distinguishes lexical meaning, whereas English is an **intonational language** where pitch cues turn boundaries (e.g., falling pitch). This seemingly negative result actually confirms that SCIR heavily relies on valid prosodic cues rather than simple silence, though it necessitates language-specific adaptation. **Noise Robustness:** Under heavy noise injection ($\sigma = 1.0$), performance drops by only 0.4%, demonstrating stability in real-world acoustic environments.

6 Discussion and Conclusion

6.1 Discussion

Our results challenge the prevailing assumption that low-latency interaction requires monolithic, end-to-end pre-training. SCIR demonstrates that a task-driven, lightweight model can surpass general-purpose SSL baselines while being 8.3x smaller. More importantly, SCIR offers a pragmatic solution to the **latency-intelligence trade-off** in modern dialogue systems. By decoupling "when to speak" (SCIR) from "what to say" (LLM), we enable a modular architecture where state-of-the-art ASR and LLM models can be deployed without suffering from their inherent latency penalties. SCIR acts as an intelligent "nervous system," utilizing its anticipatory capability (negative latency) to mask the processing time of the "brain" (LLM).

6.2 Limitations

Despite these advancements, limitations remain. First, relying exclusively on acoustics blinds the model to semantic cues (e.g., grammatical completeness), necessitating future fusion with lightweight text embeddings. Second, cultural variance poses a challenge; our cross-lingual experiments show significant divergence in backchannel behaviors between English and Chinese, suggesting timing policies need cultural adaptation. Finally, our latency-aware metrics are offline proxies; validating true naturalness requires future human-in-the-loop trials.

6.3 Concluding Remarks

In summary, we proposed SCIR, a lightweight acoustic representation that unifies turn-taking, backchanneling, and barge-in prediction. Our findings establish task-driven acoustic representations as a robust, plug-and-play foundation for next-generation real-time dialogue agents. Ultimately, by shifting the paradigm from reactive detection to anticipatory prediction, SCIR brings us one step closer to achieving seamless, human-like conversational fluency.

References

Siddhant Arora, Zhiyun Lu, Chung-Cheng Chiu, Ruoming Pang, and Shinji Watanabe. 2025. Talking turns: Benchmarking audio foundation models on turn-taking dynamics. *arXiv preprint arXiv:2503.01174*. Accepted at ICLR 2025.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Proceedings of NeurIPS*.

Timo Baumann and David Schlangen. 2012. The InproTK: A toolkit for incremental spoken dialogue processing. In *Proceedings of ACL System Demonstrations*, pages 97–102.

Jean Carletta. 2007. Unleashing the killer corpus: experiences in creating the multi-everything AMI meeting corpus. *Language Resources and Evaluation*, 41(2):181–190.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, and 1 others. 2022. WavLM: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518.

Jesse Davis and Mark Goadrich. 2006. The relationship between precision-recall and ROC curves. In *Proceedings of ICML*, pages 233–240.

Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. 2024. Moshi: a speech-text foundation model for real-time dialogue. *arXiv preprint arXiv:2410.00037*.

David DeVault, Kenji Sagae, and David Traum. 2011. Incremental interpretation and prediction of utterance meaning for interactive dialogue. *Dialogue & Discourse*, 2(1):143–170.

Starkey Duncan. 1972. Some signals and rules for taking speaking turns in conversations. *Journal of Personality and Social Psychology*, 23(2):283–292.

Erik Ekstedt and Gabriel Skantze. 2022. Voice activity projection: Self-supervised learning of turn-taking events. In *Proceedings of INTERSPEECH*, pages 5190–5194.

Qingkai Fang, Shoutao Guo, Yan Zhou, Zhengrui Ma, Shaolei Zhang, and Yang Feng. 2024. LLaMA-Omni: Seamless speech interaction with large language models. *arXiv preprint arXiv:2409.06666*. Accepted at ICLR 2025.

Yihui Fu, Luyao Cheng, Shubo Lv, Yukai Jv, Yuxiang Kong, Zhuo Chen, Yanxin Hu, Lei Xie, Jian Wu, Hui Bu, Xin Xu, Jun Du, and Jingdong Chen. 2021. AISHELL-4: An open source dataset for speech enhancement, separation, recognition and speaker diarization in conference scenario. In *Proceedings of INTERSPEECH*, pages 3665–3669.

Agustín Gravano and Julia Hirschberg. 2011. Turn-taking cues in task-oriented dialogue. *Computer Speech & Language*, 25(3):601–634.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of ICML*.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460.

Koji Inoue, Bing'er Jiang, Erik Ekstedt, Tatsuya Kawahara, and Gabriel Skantze. 2024a. Real-time and continuous turn-taking prediction using voice activity projection. In *International Workshop on Spoken Dialogue Systems Technology (IWSDS)*.

Koji Inoue, Bing'er Jiang, Erik Ekstedt, Gabriel Skantze, and Tatsuya Kawahara. 2024b. Multilingual turn-taking prediction using voice activity projection. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 11846–11858.

Koji Inoue, Divesh Lala, Gabriel Skantze, and Tatsuya Kawahara. 2025. Yeah, un, oh: Continuous and real-time backchannel prediction with fine-tuning of voice activity projection. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7171–7181, Albuquerque, New Mexico. Association for Computational Linguistics.

Adam Janin, Don Baron, Jane Edwards, Dan Ellis, David Gelbart, Nelson Morgan, Barbara Peskin, Thilo Pfau, Elizabeth Shriberg, Andreas Stolcke, and Chuck Wooters. 2003. The ICSI meeting corpus. In *Proceedings of ICASSP*, pages I–364–I–367.

Stephen C Levinson. 2016. Turn-taking in human communication–origins and implications for language processing. *Trends in Cognitive Sciences*, 20(1):6–14.

Guan-Ting Lin, Jiachen Lian, Tingle Li, Qirui Wang, Gopala Anumanchipalli, Alexander H. Liu, and Hung-yi Lee. 2025. Full-duplex-bench: A benchmark to evaluate full-duplex spoken dialogue models on turn-taking capabilities. *arXiv preprint arXiv:2503.04721*.

Ziyang Ma, Yakun Song, Chenpeng Du, Jian Cong, Zhuo Chen, Yuping Wang, Yuxuan Wang, and Xie Chen. 2024. Language model can listen while speaking. *arXiv preprint arXiv:2408.02622*.

Tu Anh Nguyen, Benjamin Muller, Bokai Yu, Marta R. Costa-jussà, Maha Elbayad, Sravya Popuri, Christophe Ropers, Paul-Ambroise Duquenne, Robin Algayres, Ruslan Mavlyutov, Itai Gat, Mary

Williamson, Gabriel Synnaeve, Juan Pino, Benoît Sagot, and Emmanuel Dupoux. 2025. SpiRit-LM: Interleaved spoken and written language model. *Transactions of the Association for Computational Linguistics*, 13:30–52.

OpenAI. 2024. GPT-4o system card.

Yizhou Peng, Yi-Wen Chao, Dianwen Ng, Yukun Ma, Chongjia Ni, Bin Ma, and Eng Siong Chng. 2025. FD-Bench: A full-duplex benchmarking pipeline designed for full duplex spoken dialogue systems. In *Proceedings of INTERSPEECH 2025*, pages 176–180.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *Proceedings of ICML*.

Antoine Raux and Maxine Eskenazi. 2008. Optimizing the turn-taking behavior of task-oriented spoken dialog systems. *ACM Transactions on Speech and Language Processing*, 5(1):1–27.

Matthew Roddy, Gabriel Skantze, and Naomi Harte. 2018. Investigating speech features for continuous turn-taking prediction using LSTMs. In *Proceedings of INTERSPEECH*, pages 586–590.

Harvey Sacks, Emanuel A Schegloff, and Gail Jefferson. 1974. A simplest systematics for the organization of turn-taking for conversation. *Language*, 50(4):696–735.

Emanuel A Schegloff. 2000. Overlapping talk and the organization of turn-taking for conversation. *Language in Society*, 29(1):1–63.

David Schlangen and Gabriel Skantze. 2011. A general, abstract model of incremental dialogue processing. *Dialogue & Discourse*, 2(1):83–111.

Elizabeth Shriberg, Andreas Stolcke, and Don Baron. 2001. Observations on overlap: findings and implications for automatic processing of multi-party conversation. In *Proceedings of Eurospeech*, pages 1359–1362.

Gabriel Skantze. 2017. Towards a general, continuous model of turn-taking in spoken dialogue using LSTM recurrent neural networks. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 220–230.

Tanya Stivers, Nicholas J Enfield, Penelope Brown, Christina Englert, Makoto Hayashi, Trine Heinemann, Gertie Hoymann, Federico Rossano, Jan Peter De Ruiter, Kyung-Eun Yoon, and Stephen C Levinson. 2009. Universals and cultural variation in turn-taking in conversation. *Proceedings of the National Academy of Sciences*, 106(26):10587–10592.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of NeurIPS*.

Bandhav Veluri, Benjamin N Peloquin, Bokai Yu, Hongyu Gong, and Shyamnath Gollakota. 2024. Beyond turn-based interfaces: Synchronous LLMs as full-duplex dialogue agents. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21390–21402, Miami, Florida, USA. Association for Computational Linguistics.

Qichao Wang, Ziqiao Meng, Wenqian Cui, Yifei Zhang, Pengcheng Wu, Bingzhe Wu, Irwin King, Liang Chen, and Peilin Zhao. 2025. NTPP: Generative speech language modeling for dual-channel spoken dialogue via next-token-pair prediction. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*.

Nigel Ward and Wataru Tsukahara. 2000. Prosodic features which cue back-channel responses in English and Japanese. *Journal of Pragmatics*, 32(8):1177–1207.

Zhifei Xie and Changqiao Wu. 2024. Mini-Omni: Language models can hear, talk while thinking in streaming. *arXiv preprint arXiv:2408.16725*.

Victor H Yngve. 1970. On getting a word in edgewise. In *Papers from the Sixth Regional Meeting of the Chicago Linguistic Society*, pages 567–578.

Hao Zhang, Weiwei Li, Rilin Chen, Vinay Kothapally, Meng Yu, and Dong Yu. 2025a. LLM-enhanced dialogue management for full-duplex spoken dialogue systems. *arXiv preprint arXiv:2502.14145*.

He Zhang, Wenqian Cui, Haoning Xu, Xiaohui Li, Lei Zhu, Shaohua Ma, and Irwin King. 2025b. MTR-DuplexBench: Towards a comprehensive evaluation of multi-round conversations for full-duplex speech language models. *arXiv preprint arXiv:2511.10262*.

A Detailed Experimental Setup

A.1 Dataset Details and Preprocessing

We utilized the **AMI Meeting Corpus** (Carletta, 2007) for the majority of our experiments. The dataset partition is detailed in Table 2. The audio was downsampled to 16kHz. We extracted 80-dimensional Log Mel-spectrograms using a 25ms window and 10ms hop size. Pitch (F_0) was extracted using YIN algorithm (implemented in Librosa), and Energy was computed as the L2 norm of the frame signal. All features were normalized to zero mean and unit variance based on statistics computed from the training set.

Split	Meetings	Duration	Samples	Pos. Rate
Train	136	37.78h	6,800	5.97%
Val	17	0.94h	85	4.24%
Test	18	1.00h	90	5.37%
Total	171	39.72h	6,975	∼5.8%

Table 2: Detailed statistics of the AMI dataset splits used in this work. **Note: Duration and Samples refer to the cumulative length and count of the extracted segments used for training/evaluation, not the raw full-session recordings.**

A.2 Model Hyperparameters

We implemented SCIR using PyTorch. The best-performing model (Medium configuration) has 11.4M parameters. Detailed hyperparameters are listed below:

- **Architecture:** 6 Transformer layers, 384 hidden dimension, 8 attention heads, 1536 FFN dimension.
- **Dropout:** $p = 0.1$ for attention and FFN layers.
- **Context Window:** 2000 frames (20 seconds) during training.

A.3 Training Configuration

All models were trained on a cluster of **8** $\times$ **NVIDIA A100 (40GB)** GPUs.

- **Optimizer:** AdamW with $\beta_1 = 0.9, \beta_2 = 0.999$, weight decay $= 0.01$.
- **Learning Rate:** Initial learning rate $5e^{-4}$, warmed up for the first 10% of steps, followed by a cosine decay to $1e^{-6}$.
- **Batch Size:** 16 sequences per GPU (effective batch size dependent on distributed setup).
- **Stopping Criteria:** Early stopping with a patience of 10 epochs based on validation loss.
- **Loss Weights:** $\lambda_{turn} = 1.0, \lambda_{bc} = 1.0, \lambda_{barge} = 1.0$.

B Comprehensive Ablation Studies

B.1 Impact of Encoder Depth

We investigated the depth of the Causal Transformer encoder. As shown in Table 3, a 6-layer architecture provides the optimal balance between performance and efficiency. Increasing depth to 8 layers resulted in a slight performance degradation ($0.816 \rightarrow 0.805$), likely due to overfitting on the limited size of the AMI dataset.

Table 3: Impact of encoder depth on performance.

Layers	Params	Avg AUC	Finding
4 Layers	7.86M	0.814	Underfitting
6 Layers	**11.4M**	**0.816**	**Optimal**
8 Layers	15.0M	0.805	Overfitting

B.2 Data Augmentation Analysis

We explored various augmentation techniques to improve robustness. Table 4 details the impact of each strategy. While aggressive frequency masking degraded performance, we found that **Gaussian Noise injection combined with mild time masking** provided the optimal balance, yielding the +1.25% gain reported in our main results.

Table 4: Impact of data augmentation strategies. The final SCIR model employs the "Optimal Policy" (Noise + Mild Masking).

Strategy	Avg AUC	Change
No Augmentation	0.783	Baseline
Freq Masking (Aggressive)	0.758	-3.2%
Gaussian Noise Only	0.789	+0.7%
Optimal Policy (Final)	**0.793**	**+1.25%**

C Context Window Analysis

Figure 7 illustrates the relationship between historical context length and prediction accuracy. The performance improves rapidly as the context window increases from 1 second to 4 seconds. The performance plateaus at approximately 10 seconds,

suggesting that the acoustic cues relevant for immediate turn-taking decisions are contained within the most recent 10 seconds of conversation. Extending the window beyond this point introduces irrelevant noise without adding predictive value.

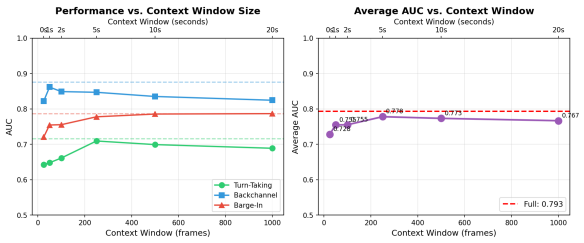

Figure 7: **Performance vs. Context Window Size.** Performance peaks at a 10-second window.

D Detailed Baseline Comparisons

D.1 Lightweight Baselines

To validate the necessity of the Transformer architecture, we compared SCIR against simpler lightweight baselines (Table 5). "TinyTF" is a smaller transformer model, while "DeepCNN" uses a stacked convolutional architecture. SCIR significantly outperforms all lightweight alternatives, confirming that model capacity matters for capturing complex interaction dynamics.

Table 5: Comparison with lightweight architectures.

Model	Params	Avg AUC
MLP	0.09M	0.637
CNN (Standard)	0.21M	0.665
DeepCNN	0.18M	0.667
TinyTF	0.41M	0.676
SCIR (Ours)	**11.4M**	**0.793**

D.2 Heavyweight Baselines (Partial Fine-tuning)

While our main results focus on frozen backbones for fair comparison of representations, we also investigated **partially fine-tuning** the Wav2Vec 2.0 encoder. As shown in our experiments, fine-tuning the last 2 layers of Wav2Vec 2.0 combined with our temporal head yielded an Average AUC of **0.832**. While this is 4.9% higher than SCIR (0.793), it comes at the cost of requiring 99.5M parameters and significantly higher memory usage, rendering it less suitable for on-device applications compared to the efficient SCIR.

E System-Level Analysis

E.1 Latency Statistics

We analyzed the distribution of prediction latency relative to the ground-truth event start time. "Negative" latency indicates the model predicts the event *before* it happens.

- **Turn-Taking:** Mean Latency = **-861 ms**. 95.9% of events were predicted early.
- **Backchannel:** Mean Latency = **-535 ms**. 85.3% of events were predicted early.
- **Barge-In:** Mean Latency = **-798 ms**. 98.0% of events were predicted early.

This substantial lead time is the key mechanism that allows SCIR to mask downstream system latency.

E.2 Conversation Complexity

We analyzed model robustness across different conversation densities (Figure 8). We categorized conversation segments into Simple, Moderate, and Complex based on the frequency of speaker changes. The model maintains high performance even in "Complex" scenarios (>10 turns/min), demonstrating robustness to chaotic acoustic environments.

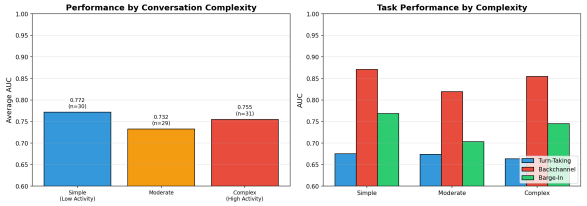

Figure 8: **Performance by Conversation Complexity.** The model remains robust even in high-activity segments.

F Cross-Domain Transfer Analysis

We conducted zero-shot transfer experiments to evaluate generalization.

- **AMI → ICSI (English):** High retention (78.5%), indicating strong adaptability within the same language family.
- **AMI → AISHELL-4 (Mandarin):** We observed a significant drop in turn-taking performance (Retention 19.2%). However, Overlap detection remained relatively robust (Retention 40.7%). This confirms the hypothesis that

prosodic turn-taking cues (e.g., pitch contour) are heavily language-dependent (tonal vs. intonational languages), while overlap cues are more universal.

G Robustness and Probing

G.1 Robustness Profile

We evaluated stability under 12 acoustic perturbations (Figure 9). The model is highly invariant to **Pitch Shift** (0.1% drop) and **Temporal Jitter** (3.9% drop), confirming it relies on relative prosodic contours rather than absolute values. However, it is more sensitive to spectral degradation like **Impulse Noise** (17.0% drop).

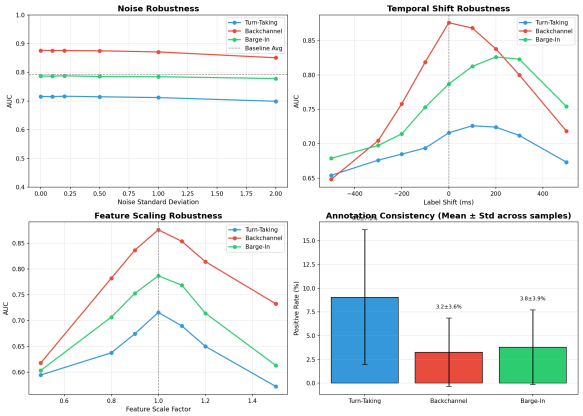

Figure 9: **Robustness Profile.** Impact of acoustic perturbations on model performance.

G.2 Layer-wise Probing

We analyzed the discriminative power of each encoder layer using linear probes (Figure 10).

- **Backchannel (Red):** Performance peaks early at Layer 3, suggesting it relies on shallower acoustic triggers.
- **Turn-Taking (Blue):** Performance improves steadily up to Layer 6, requiring deeper contextual integration.

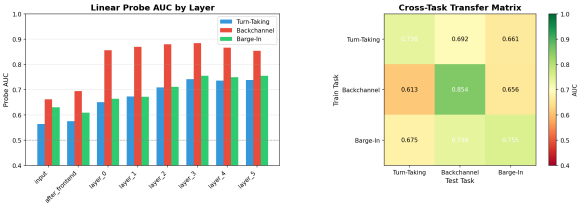

Figure 10: **Layer-wise Probing.** AUC scores of linear probes attached to each encoder layer.

H Error Analysis

H.1 Error Distribution

We categorized false positives (FP) and false negatives (FN) to understand failure modes (Figure 11).

- **FP - High Energy Noise:** The primary source of false positives (58% of errors) comes from high-energy non-speech sounds, such as laughter or coughing, which the model confuses with speech onset.
- **FP - Ambiguous Boundaries:** 31.6% of false positives occur in short durations (<200ms) near true boundaries, indicating boundary ambiguity rather than complete hallucination.

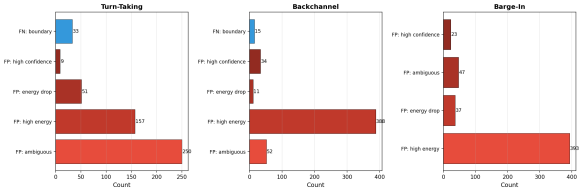

Figure 11: **Error Type Distribution.** High-energy non-speech sounds are the leading cause of false positives.

H.2 Latent Space Visualization

To qualitatively assess the learned representations, we visualized the trajectory of a 10-second conversation in the PCA-reduced latent space (Figure 12). The trajectory clearly separates states of silence, listening, and active speaking, confirming that SCIR captures the continuous dynamics of conversation.

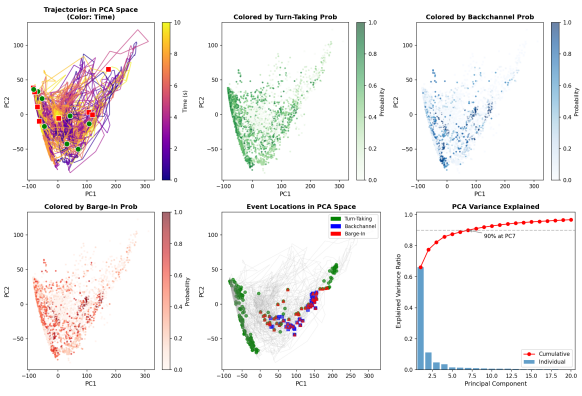

Figure 12: **Latent Space Trajectory.** PCA visualization showing clear separation between turn-taking events (green) and non-events.