

Conversational yielding beyond streaming speech generation

Banuru Mohammed Fawaz | Speech AI Research Internship | Vaani Research Lab |
23 October 2026

Streaming speech generation is not sufficient for natural real-time conversation, because conversational control – deciding when to continue, hold, or yield while still listening – is not fully specified as a separate policy in these reports and is not measured by first-audio latency.

This thesis concerns control over the conversational floor, rather than incremental audio alone. Moshi explicitly supports simultaneous listening and speaking; its turn-taking is learned within joint stream modeling. Qwen2.5-Omni describes streaming perception and generation. These are different evidential claims. The research opportunity is to test whether a separately supervised controller can distinguish a listener response from an attempt to take the floor, using information already available inside a frozen speech model.

Evidence convention. Paper-stated claims carry M or Q citations; architectural consequences are interpretation. The controller and experiments are proposals. External related work is marked [external]. No experimental results are claimed. M and Q identify the attached Moshi and Qwen2.5-Omni reports [1,2]; page numbers are printed page numbers.

1 Seven architectural decisions

Point	Moshi	Qwen2.5-Omni	Consequence for the thesis
Encoder	Causal SeaNet-based Mimi; WavLM distillation (M §§3.3.1-3.3.2, pp.10-12).	Mel input; Whisper-large-v3 initialization (Q §2.2, p.4; §3, p.6).	Receiving speech does not itself specify a floor-control action.
Representation	Eight codebooks at 12.5 Hz; split semantic/acoustic quantization (M §§3.3.1-3.3.2, pp.11-12).	Input frames cover about 40 ms; qwen-tts-tokenizer produces output codes (Q §§2.2-2.3, pp.4-5).	Representation quality and conversational decisions need separate assessment.
Alignment	Summed token embeddings; aligned self-text supplies Inner Monologue (M §3.4.1, p.14; §3.4.4, p.16).	Adapter alignment; Talker receives hidden states and sampled text embeddings (Q §3, p.6; §2.3, p.5).	Generation state may supply context beyond speech activity.
Streaming	Mimi frames: 80 ms; reported latency: 160 ms theoretical, 200 ms practical (M §3.3.1, p.11; abstract, p.1).	Input blocks: 2 seconds; DiT context: four blocks, two past and one future (Q §2.4, p.5).	Processing delay and appropriate yielding are different quantities.
Generation	Temporal/depth autoregression; Mimi waveform decoding (M §3.4.1, pp.12-14; §3.3.1, p.10).	Dual-track autoregressive Talker; flow-matching DiT and BigVGAN decoding (Q §§2.1-2.4, pp.3-5).	Intervene on playback control without assuming a new generator.
Completion	Continuous speech/silence; PAD/EPAD organize self-text timing (M §3.4.4, pp.16-17).	Text EOS in Figure 2; ChatML message boundaries (Q Fig.2, p.3; §4.1, p.6).	Silence, text completion, and conversational yielding differ.

Point	Moshi	Qwen2.5-Omni	Consequence for the thesis
Training	Weighted joint-token cross-entropy; conversation-specific fine-tuning (M §4.4, pp.20-21).	Staged multimodal training; Talker continuation, DPO, speaker tuning (Q §§3-4.3, pp.6-7).	Capability and learned interaction behavior can have different causes.

Disclosure. Qwen does not report output-codec rate/codebooks/bitrate, adapter dimensions/layers, output-block duration, numerical end-to-end latency, audio hours/named audio training corpora, or complete pretraining/tokenizer objectives. Neither report fully specifies endpoint, playback-completion/flush, or session-stop rules; Qwen also leaves interruption handling unspecified. Moshi’s practical-latency figure lacks a complete measurement protocol. Qwen’s causal-decoder wording coexists with DiT lookahead (Q §§2.3-2.4, p.5). These omissions prevent matched codec, latency, data-efficiency, and stopping-protocol comparisons; they do not demonstrate failures.

2 Three limitations, ranked

2.1 Continuous generation leaves conversational decisions implicit

Addressed problem: turn-segmented systems cannot naturally represent overlapping speech and interruptions. **Mechanism:** Moshi jointly models separate user and system streams, allowing simultaneous listening and speaking and natural silence (M §1, p.2; §3.4.4, p.17).

Residual question: when does this learned behavior correctly preserve or relinquish the floor? Moshi evaluates gaps, overlaps, and dialogue quality (M §5.6, p.31), but aggregate distributions cannot determine whether a particular backchannel caused an inappropriate stop. Qwen’s streaming architecture likewise does not establish this event-level behavior. The evidence favors Moshi as the experimental base because simultaneous interaction is explicit. The question is whether its internal state supports better action selection than an audio-trigger rule. Continuous modeling permits this experiment; it does not prove adequate control decisions.

2.2 Low transport delay does not establish timely social action

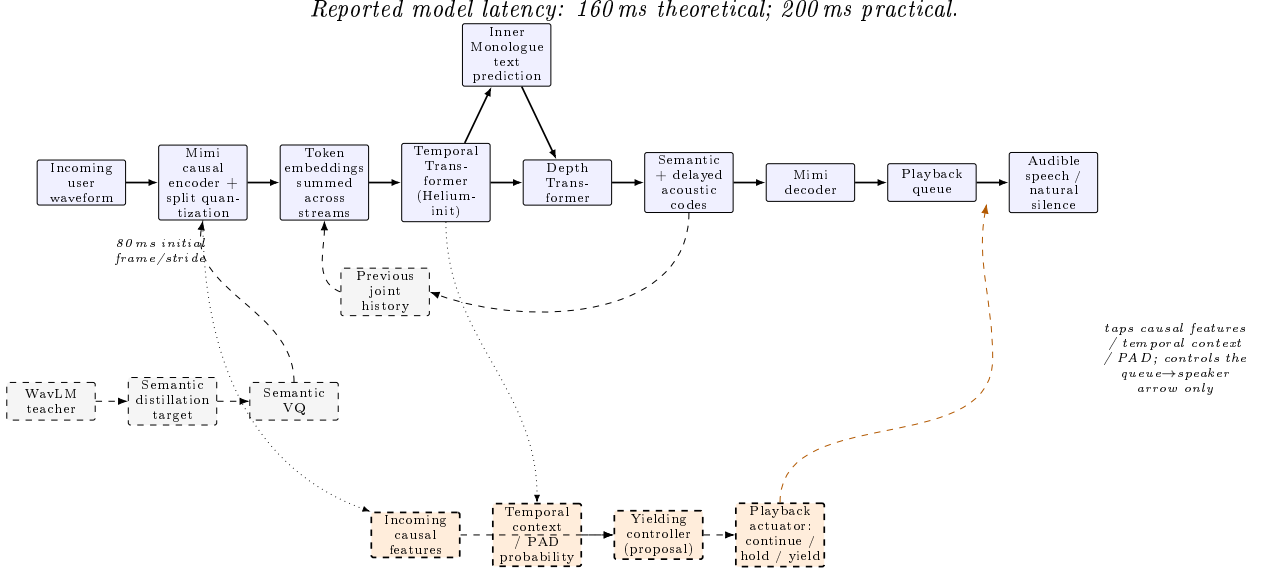
Addressed problem: serial processing and delayed audio generation obstruct rapid interaction. **Mechanisms:** Moshi uses causal codec frames and hierarchical prediction; Qwen uses block-wise processing and bounded-context waveform decoding (M §§3.3-3.4, pp.9-17; Q §2.4, p.5).

Residual question: does the system stop appropriately after an interruption while continuing through a backchannel? First-audio latency measures onset, not that distinction. Faster yielding is trivially possible by suppressing output whenever speech appears; fewer false yields are possible by refusing to stop. Neither establishes improved conversation. Compare false yields under common interruption-latency and missed-interruption constraints. Input buffering, controller computation, and playback queues all belong in that timing account.

2.3 Available cues and training supervision may be insufficient

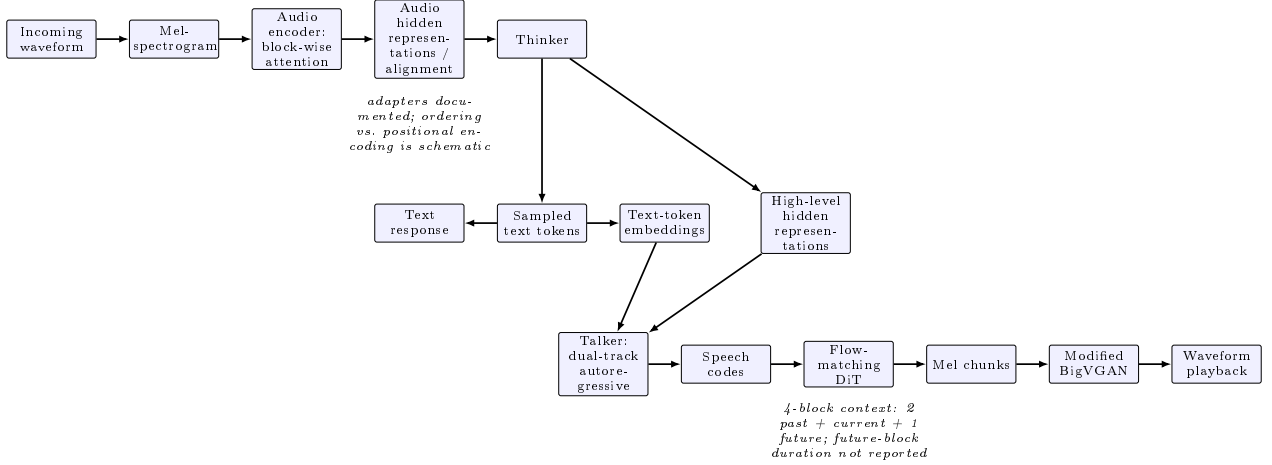
Addressed problem: speech models must combine linguistic content and acoustic detail. **Mechanisms:** Mimi distills semantic information while retaining acoustic codes; Qwen supplies hidden representations and discrete text to reduce phonetic ambiguity (M §3.3.2, p.12; Q §2.3, p.5).

Residual question: which cues distinguish acknowledgment, hesitation, and interruption in context? Generation state might indicate whether the assistant is completing a phrase, waiting, or continuing an explanation, but its utility must be tested. Moshi identifies shortcomings in simulated dialogue without overlap; Qwen acknowledges noisy supervision (M §4.4, pp.20-21; Q §4.3, p.7). A gain must be attributed to a controlled state intervention rather than differences between training corpora.



Per-stage compute duration (encoder, Transformer, decoder, queue) is not individually reported; only the end-to-end figures above are stated. Dashed gray boxes are training-only; dashed orange boxes are the proposed controller, not an evaluated component.

Input: 2-second attention blocks (not an end-to-end latency measurement). DiT: 4-block context, 2 past + current + 1 future; future-block duration not reported.



Per-stage compute duration and total onset delay are not reported in the source for either path.

Figure 1: Architecture and timing dependencies. Top: Moshi (M §§3.3–3.4, pp.9–17; abstract, p.1). Bottom: Qwen2.5-Omni (Q §§2.1–2.4, pp.3–5; §3, p.6). Arrows show dependencies, not proportional execution durations. Solid = established pipeline; dashed gray = training-only path; dashed orange = the proposed yielding controller (a proposal, not a reported or evaluated component).

3 End-of-speech is four decisions

Decision	Operational meaning	Observable contrast
Input endpoint	User has finished the current contribution.	Hesitation pause versus completed question.
Conversational yield	Assistant gives the user the floor.	Backchannel versus genuine interruption.
Output completion	Intended response has finished playing.	Intended completion versus playback/queue state.
Session termination	Conversation processing stops running.	Explicit session-stop action versus ongoing listening.

Moshi establishes continuous listening and generation, including silence; PAD/EPAD concern text timing rather than a standalone utterance-stop command (M §3.4.4, pp.16-17). Qwen establishes text/message boundaries and streaming waveform generation (Q Fig.2, p.3; §2.3-2.4, p.5; §4.1, p.6). The disclosure identifies the remaining specification boundary.

The observable test should preserve the floor through a user’s hesitation, permit an answer after a completed question, continue through a non-interrupting backchannel, and yield after a genuine interruption. These are annotation targets, not assumed behaviors. A user-owned hesitation tests premature response onset; it cannot count as a false yield when the assistant has no active or pending output. Report that case separately. For hesitation episodes eligible for false-yield analysis, annotators must independently establish that the assistant should retain the floor.

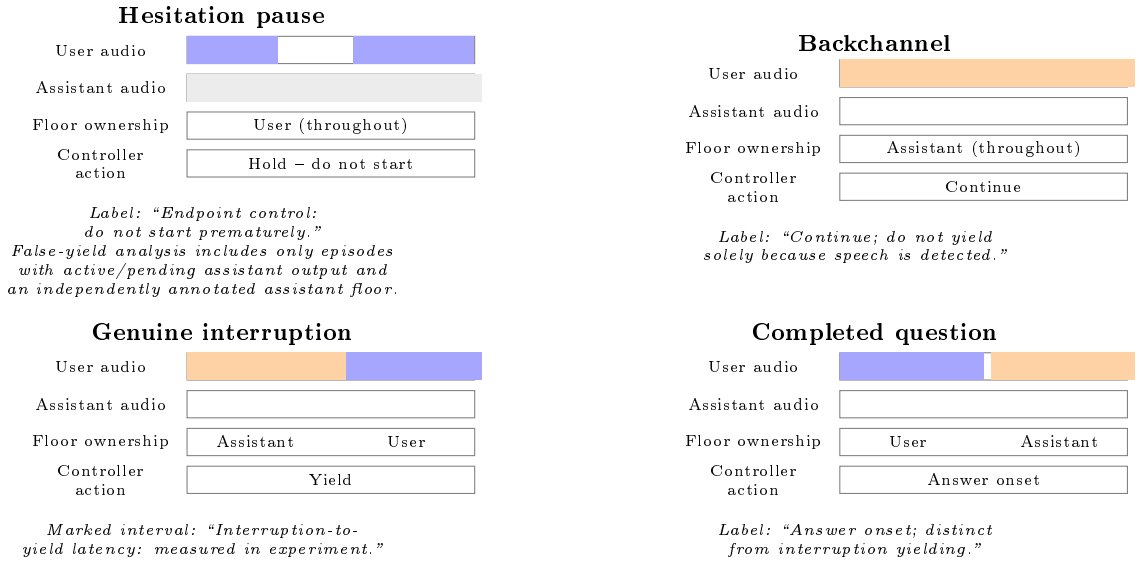


Figure 2: Conversational events and desired actions. Four lanes per panel: user audio, assistant audible audio, desired floor ownership, desired controller action. Time is symbolic (no numerical durations). These are desired behaviors for the proposed test, not measured traces.

4 Selected hypothesis

We hypothesize that a frozen speech LLM’s own generation state improves continue/hold/yield decisions beyond incoming audio and observable playback history, which predicts fewer false yields on eligible backchannel and hesitation events than speech-activity rules and an audio-only learned controller, under matched interruption-to-yield latency and missed-interruption constraints.

The hypothesis is refuted if the state-aware controller cannot improve that constrained comparison

on held-out speakers and conversations, or if removing generation state preserves the benefit.

Four alternatives remain secondary: whether buffering dominates onset latency; whether codec phonetic quality predicts interaction accuracy; whether limited lookahead causes prosodic commitment errors; and whether real overlapping dialogue contributes beyond synthetic conversation training.

5 Proposed improvement: state-aware yielding

5.1 Failure mechanism and proposed change

A speech-activity rule can interpret acknowledgment as a takeover attempt. This is a hypothesized baseline failure, not an established failure of either paper’s model.

Attach a small causal controller to **frozen Moshi**. Inputs are current Mimi representations and causal speech-activity features, observable assistant playback activity, elapsed activity/silence durations, and queue occupancy. Add Moshi’s Temporal Transformer context vector and self-text PAD probability as generation-state features. Temporal context and text prediction are defined in the architecture (M §3.4.1, pp.13-14; §3.4.4, pp.16-17); implementation access requires checking. Only values available at decision time enter the controller.

A learned projection and recurrent classifier produce **continue, hold, or yield**. Continue releases available audio. Hold briefly suspends release while listening continues. Yield suppresses output and clears pending playback while retaining listening; playback resumes only through a later controller decision. All variants share the same bounded actuator. Hold duration is capped on development data; inappropriate holding counts as disruption, not as a successful non-yield.

The contribution under test is the speech LLM’s generation state, not turn-taking prediction itself. Base-model parameters remain frozen; controller size and runtime are measured rather than assumed negligible.

5.2 Training and labeled data

Train only the projection and classifier with supervised action cross-entropy. Labels identify event type, floor ownership, event onset, acceptable action intervals, and interruption-response deadlines. Annotators hear surrounding dialogue and mark ambiguity independently of controller outputs. Future context can inform annotation but cannot become an input feature.

Collect consented human-Moshi conversations and controlled reenactments, recording separate user and actually played assistant channels. Replay recordings through frozen Moshi to obtain causal feature traces. Fisher’s separate channels provide a potential auxiliary source for human conversational cues (M §4.2, p.18); new action annotations and human-system validation remain necessary. Split by speaker, conversation, and scenario. Retain disagreement scores for sensitivity analysis rather than silently excluding ambiguous cases.

Select class weights, thresholds, and hold limits on development data, leaving the test set untouched. Include logged policy rollouts as well as reenactments to expose training to action histories. A pilot should establish label reliability and event coverage before fixing the evaluation population.

5.3 Predicted effect and baselines

The prediction is fewer inappropriate floor relinquishments without slower or less reliable interruption handling. The **speech-activity-triggered baseline** suppresses output after incoming activity satisfies an activity threshold and persistence condition. Include both a **prespecified fixed-threshold version** and a **development-tuned version**. Give tuned rules and learned controllers the same validation protocol and playback actuator. Report unmodified Moshi as context.

Winning only against the fixed rule does not establish a generation-state benefit. Winning through additional waiting does not satisfy the hypothesis. The audio-only learned condition is necessary to distinguish useful internal state from the general benefit of supervised event classification.

6 Experimental design and causal controls

6.1 Primary experiment and metric

Use matched **backchannel**, **hesitation-pause**, and **genuine-interruption** scenarios, balancing speakers and recording conditions across systems. Infer floor intent from context, not event duration alone. Completed questions provide an endpoint sanity check outside the primary false-yield comparison.

The primary metric is the **unweighted mean of false-yield rates for eligible backchannels and hesitation episodes**, also reported separately. A false yield is suppression when floor retention is annotated. A hold exceeding the annotated tolerance counts as equivalent disruption. Publish eligibility denominators, premature starts during user-owned hesitation, and unnecessary hold time.

Compare operating points under common **interruption-to-yield latency** and **missed-interruption** limits established from development-set baseline behavior before testing. Latency runs from annotated interruption onset to actual playback cessation. Failure to yield within the annotated deadline is a miss. Report latency distributions alongside misses, never successful responses alone. If no operating point meets both constraints, the proposed benefit is unsupported. Estimate paired uncertainty by conversation, preserving within-dialogue dependence.

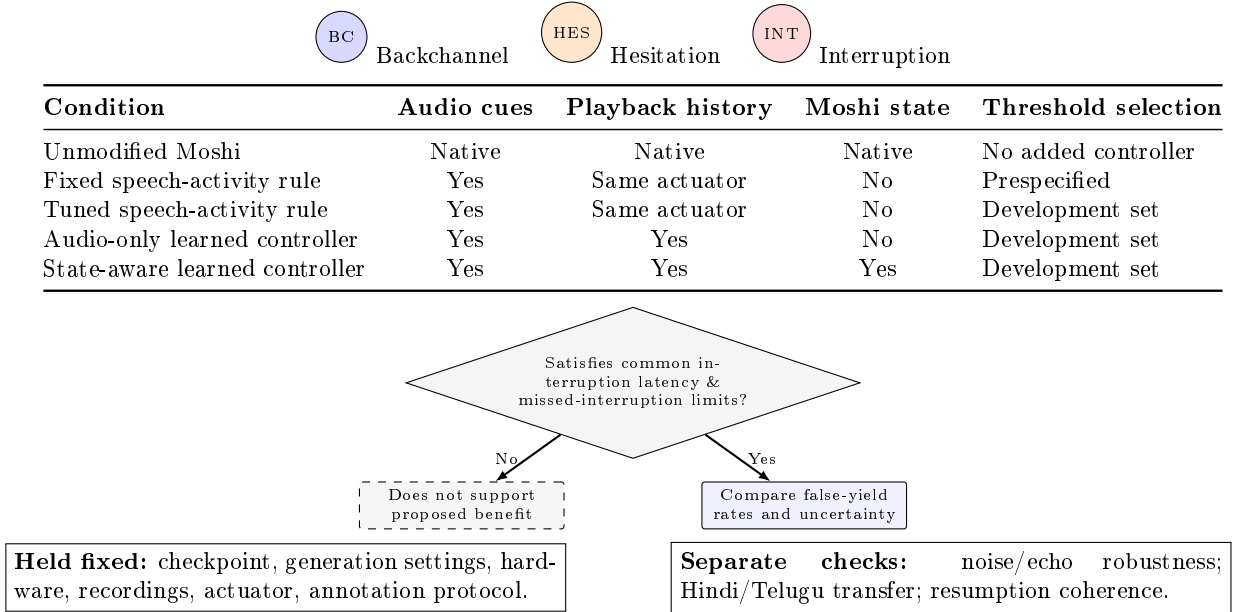


Figure 3: Experimental design matrix. No empirical curve is implied; all rows are proposed conditions, not measured results.

6.2 Diagnostic ablation

Remove generation-state features while retaining acoustic features, playback history, queue state, classifier capacity, and training procedure. This tests whether a learned audio policy or knowledge that the assistant is speaking explains the gain. Both learned variants receive observable timing and playback cues; internal state must contribute beyond them. Keep the classifier architecture and training budget matched, and report any parameter-count difference introduced by the input projection.

6.3 Robustness and Indic transfer question

Repeat held-out events with controlled noise and echo perturbations, identical across variants. Report activity-detection and controller errors separately. A **Hindi/Telugu subset** asks whether learned cues transfer, using fluent annotators and language-specific agreement reporting. Separate adaptation from zero-shot transfer; assume no known Indic failure of either base model. Report basic

instruction comprehension so lexical misunderstanding is not automatically attributed to yielding.

6.4 What is controlled

Fix the Moshi checkpoint, generation settings, hardware, user recordings, playback mechanism, and annotation protocol. Immediate replay comparisons start from a common assistant prefix. Counterbalance live trials where different assistant actions change user responses. The design does not control Moshi’s original training history or establish causality between Moshi and Qwen architectures.

7 Failure modes and alternative explanations

Timing shortcuts. Generation state may reveal only elapsed time or current speaking status. Supplying playback history, queue state, and durations to both learned variants tests that explanation. A benefit surviving state removal would reject the proposed contribution even if the controller still beats the rule.

Threshold selection and extra waiting. The development-tuned rule tests whether calibration explains an apparent gain. Common interruption constraints and explicit hold accounting test whether conservatism merely hides errors. Report the full operating-point comparison rather than selecting favorable test thresholds.

Speaker and script shortcuts. Speaker-disjoint splits test identity dependence. Natural held-out conversations test whether reenactment-specific lexical or prosodic cues explain performance. Language-specific disagreement and comprehension checks are needed before interpreting the Hindi/Telugu subset.

Internal versus audible history. Moshi can internally generate speech that suppression prevents the user from hearing. Freezing the model does not solve this mismatch. Track unheard output, resumption coherence, truncation, and controller overhead in follow-up turns. This proposal controls audible yielding; it does not repair the generator’s internal dialogue state. Failure on these checks would limit or invalidate deployment despite a favorable primary metric.

Metric sufficiency. Fewer false yields are useful only when genuine interruptions remain responsive and subsequent conversation remains coherent. Audio quality, excessive pauses, and task completion should accompany the primary constrained metric as diagnostic outcomes, without replacing it after results are seen.

8 Related work and contribution boundary

dGSLM provides a duplex-dialogue comparison in Moshi’s evaluation (M §5.6, p.31); **Mini-Omni** motivates Qwen’s dual-track decoder (Q §2.1, p.3). **TurnGPT [external]** predicts conversational turn-taking [3], and **Voice Activity Projection [external]** models future conversational voice activity [4]. Turn prediction therefore predates this proposal. The narrower question is whether Moshi’s own generation state adds control information beyond matched audio and playback cues. Priority for that combination requires a wider literature check; no first-use claim is made.

9 Position

Moshi is the justified experimental base because simultaneous listening and speaking are explicit in its formulation. Qwen demonstrates why streaming input and output alone do not establish event-level conversational decisions. The controller tests whether generation context reduces false yields without delaying genuine interruptions. Success depends on the constrained comparison, the state-removal ablation, and coherent resumption, not a lower first-audio number.

References

- [1] Defossez et al. **Moshi: a speech-text foundation model for real-time dialogue**. Attached arXiv:2410.00037v2, 2 October 2024.
- [2] Qwen Team. **Qwen2.5-Omni Technical Report**. Attached arXiv:2503.20215v1; title-page date 27 March 2025.
- [3] Ekstedt and Skantze. **TurnGPT: a Transformer-based Language Model for Predicting Turn-taking in Spoken Dialog**. Findings of EMNLP, 2020. <https://aclanthology.org/>

2020.findings-emnlp.268/

- [4] Ekstedt and Skantze. **Voice Activity Projection: Self-supervised Learning of Turn-taking Events**. 2022. <https://arxiv.org/abs/2205.09812>

Appendix A. Detailed Moshi evidence

Encoder and representation learning

Mimi uses a SeaNet-based causal autoencoder. Four convolutional blocks with strides (4, 5, 6, 8) and a final convolution with stride 2 map **24 kHz** mono audio to **512-dimensional** representations at **12.5 frames/s**. Transformer modules before and after quantization each have **8 layers and 8 heads**, with causal masking (M §3.3.1, p.10).

Mimi uses **8 codebooks**, each with **2,048 entries**, giving the reported **1.1 kbps** at **12.5 Hz** (M §3.3.1, p.11). Split quantization combines one semantic VQ with a **7-level** acoustic RVQ, summing their outputs for reconstruction. WavLM teacher features are **1,024-dimensional at 50 Hz**, pooled to **12.5 Hz**. Cosine-distance distillation uses non-causal teacher information during training while Mimi remains causal at inference (M §3.3.2, p.12).

The paper evaluates an adversarial-only codec variant retaining feature and discriminator losses alongside semantic distillation. This is a description of the evaluated variant rather than an unsupported final-configuration designation (M §3.3.1, p.11; §5.2, Table 4, p.24).

Alignment, generation, and streaming

The Temporal Transformer receives summed learned embeddings across parallel sequences. The Depth Transformer conditions on temporal context and preceding within-timestep tokens (M §3.4.1, pp.13-14). Inner Monologue aligns self-text using Whisper timestamps and PAD/EPAD. User input is not mediated by a user transcript in this formulation (M §3.4.4, p.16).

The joint sequence includes both participants’ audio and Moshi’s text: $K = 2Q + 1$, with $Q = 8$. Actual user audio replaces predicted user-stream tokens during interaction (M §3.4.4, p.17). Mimi’s initial frame and stride are **80 ms** (M §3.3.1, p.11). The reported model latency is **160 ms theoretical and 200 ms practical** (M abstract, p.1). Acoustic delay is **2** during pretraining and **1** during fine-tuning (M §5.3, p.27).

Separate ASR/TTS configurations use **2 seconds** of text/audio delay; these are distinct from the conversational configuration (M §5.7, pp.31-32; Appendix C, p.56). EPAD ends padding, not an utterance; Appendix C’s text <bos>/<eos> separates speakers in the multi-stream TTS setup (M §3.4.4, p.16; Appendix C, p.56).

Training objectives and data

Stages are Helium pretraining, single-stream audio pretraining, simulated multi-stream post-training, Fisher fine-tuning, and synthetic instruction fine-tuning. Joint-token cross-entropy gives semantic tokens weight **100** and acoustic tokens weight **1** (M §4.4, pp.20-21, Eq.7).

Helium uses **2.1 trillion text tokens** (M §1, p.3), drawn from CommonCrawl, Wikipedia-family sources, StackExchange, and peS2o (M §4.1, p.17). Audio pretraining uses **7 million hours**, predominantly English, transcribed with Whisper-large-v3. Fisher supplies **2,000 hours** with separate conversation channels (M §4.2, p.18).

Synthetic instruction data contains **more than 20,000 hours**. A separate **170-hour** conversation collection supports TTS training and transcript preparation; Moshi is not directly trained on that audio collection (M §§4.2-4.3, p.18; §4.4, p.21). These figures do not establish a unique-data total.

Appendix B. Qwen evidence and proposal checks

Input representation and alignment

Audio is resampled to **16 kHz**, using a **128-channel** mel-spectrogram, **25 ms** window, and **10 ms** hop. Representations correspond approximately to **40 ms** each (Q §2.2, p.4). The Qwen2-Audio encoder is initialized from Whisper-large-v3; adapters are trained before encoders against the initially frozen LLM (Q §2.2, p.4; §3, p.6).

Thinker is a Transformer decoder. Talker receives high-level representations and sampled text-token embeddings, combining semantic context with phonetic specificity (Q §2.1, p.3; §2.3, p.5). Exact adapter ordering relative to positional encoding remains schematic in Figure 1.

Generation and streaming

Talker autoregressively generates text and audio without requiring word-level/timestamp-level text-speech alignment (Q §2.3, p.5). Input attention uses **2-second blocks**; audio-video time interleaving separately uses **2-second groups** (Q §2.2, p.4; §2.4, p.5).

Speech codes pass through flow-matching DiT, mel-spectrogram generation, and modified BigVGAN. DiT uses **4 blocks**, including **2 lookback blocks** and **1 lookahead block** (Q §2.4, p.5). Distinguish this waveform path from Talker’s autoregressive code generation.

Training requirements

Pretraining freezes the LLM initially, unfreezes all parameters next, and extends context last. Second-phase additions are **800 billion image/video-related tokens**, **300 billion audio-related tokens**, and **100 billion video-with-audio-related tokens**. Context increases from **8,192 to 32,768 tokens** (Q §3, p.6).

Thinker uses text, visual, audio, and mixed-modality instruction data. Talker uses continuation next-token prediction, DPO, and multi-speaker instruction/speaker tuning. Preference ranking uses WER and punctuation-pause error (Q §§4.2-4.3, pp.6-7).

Annotation and causal interpretation

Annotate event class and desired floor action separately. Establish eligibility before observing controller outcomes. Record onset, floor ownership, active/queued output, action tolerance, and interruption deadline. Keep ambiguity scores. The primary false-yield denominator excludes user-owned pauses without assistant output; those belong to premature-start analysis.

Replay tests immediate actions from a common prefix. Interactive follow-up tests whether different policies change human behavior and resumption coherence. These experiments support different conclusions and should not be merged into a single causal claim.

VERIFY before a novelty or implementation claim

[external] Check TurnGPT’s turn-prediction role against [3], and Voice Activity Projection’s future-activity task against [4]. Their existence does not establish novelty of the proposed state-aware wrapper. Search for previous generation-state-based turn controllers before making a priority claim.

Check causal feature extraction, PAD-probability access, controller overhead, audible discontinuities, and mismatch between internal and played history. Confirm Fisher access and permitted use before selecting it as an annotation source. Controller architecture, supervision, metrics, thresholds, Hindi/Telugu transfer, and all expected effects are proposals, not findings from the papers.