FULL-DUPLEX SPOKEN LANGUAGE MODELING WITH DISCRETE DIFFUSION

Anonymous authors
Paper under double-blind review

ABSTRACT

We introduce **Diffuplex**, a full-duplex spoken language model based on discrete diffusion that achieves an unprecedented balance between linguistic intelligence and response speed. With thinking enabled, Diffuplex finalizes an average of 10.15 tokens per forward pass, achieving substantially higher output speed than comparably intelligent turn-based autoregressive models. Without thinking, it matches the first-audio latency of existing full-duplex models while delivering markedly stronger spoken intelligence. This capability is enabled by separating *when to say* from *what to say*, rather than compressing both into a single autoregressive timeline. During inference, Diffuplex continuously tracks the conversation through an autoregressive interaction policy, while independently generating and refining response content through parallel diffusion. Consequently, semantic computation is no longer bounded by the perception audio frame rate, allowing intelligence and responsiveness to improve simultaneously. Taken together, these results establish a new paradigm for spoken agents that can sustain real-time dialogue while devoting parallel computation to intelligent response generation.

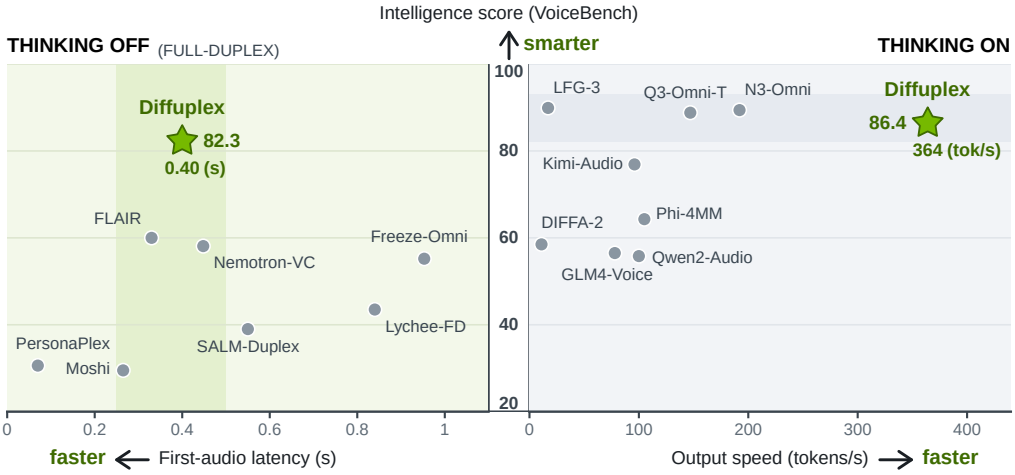


Figure 1: **Spoken intelligence and response efficiency.** The shared vertical axis is the VoiceBench overall score across 9 subsets. *Left:* Diffuplex without thinking versus full-duplex spoken dialogue models, plotted against first-audio latency (from the end of user speech to the start of model speech). *Right:* Diffuplex with thinking versus advanced turn-based AR models, plotted against output speed. All output speeds were measured at batch 1 on the same NVIDIA RTX 6000 GPU.

1 INTRODUCTION

An ideal spoken language model should exhibit both fluid conversational behavior (*when to say*) and intelligent response generation (*what to say*). From a human cognitive perspective, when to

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

say requires continuously listening to the interlocutor to pinpoint appropriate moments to speak or stop in real time. Conversely, what to say entails articulating internal thoughts at opportune moments, which are progressively refined during speaking. Neural recordings during conversational turn-taking identify distinct activity associated with speech perception, planning, and production, even when these processes overlap in time (Castellucci et al., 2022). This suggests a useful design principle: conversational timing and response formulation should be coordinated without being forced into a single sequential process.

Turning this separation into a spoken language model, however, remains a formidable challenge. Currently, highly intelligent spoken language models predominantly learn turn-based interaction, where response strategy usually relies on an external Voice Activity Detection (VAD) model. While recent full-duplex models that unify when to say and what to say deliver fluid interactions, they exhibit severe linguistic and cognitive degradation. We argue that this degradation arises from conflating these two learnable dimensions into a single one-dimensional autoregressive (AR) timeline: (1) Conflicting Objectives: Optimizing both tasks via a single Cross-Entropy loss results in gradient conflict, where the dense supervision of low-level temporal alignment heavily dilutes the delicate representations required for high-level semantic generation. (2) Temporal Frame-Rate Binding: By tying AR generation directly to a rigid physical frame rate, e.g., 80 ms in Moshi (Défossez et al., 2024), the model is compelled to process semantics in lockstep with real-time speech, stripping its capacity to allocate elastic test-time compute (i.e., Chain-of-Thought) for deep inference.

To break this coupling, we exploit a distinctive property of discrete diffusion language models (DLM): response tokens can be generated and refined in parallel. This introduces a second computational axis that is independent of physical time. As illustrated in Figure 2, an AR policy advances along the horizontal timeline t, continuously processing incoming speech and determining when a response should begin. Once triggered at time θ, a diffusion process expands orthogonally from the timeline and iteratively refines an entire response canvas (i.e., a generation block). To support streaming interaction, we further design an anchored first-n decoding strategy: the first n response tokens are finalized early and immediately synthesized, while audio playback provides a time buffer for the remaining token refinement.

We instantiate this two-axis formulation as **Diffuplex**, a spoken language model that separates semantic generation from conversational control within a unified backbone. We efficiently connect a pretrained speech encoder (NVIDIA, 2025) to a text DLM (DiffusionGemma Team et al., 2026), allowing Diffuplex to take spoken input as diffusion condition, while inheriting the linguistic capabilities and knowledge encoded in the text model. Since the same backbone can also operate in AR mode, we introduce only a lightweight state embedding and prediction head to learn semantically informed speaking decisions from conversational data. At inference time, the two modes assume complementary roles: the AR mode continuously processes the incoming speech stream and determines *when to say*, while the diffusion mode is triggered at appropriate moments to generate and iteratively refine *what to say* in parallel.

Experiments on public benchmarks evaluate Diffuplex along two complementary dimensions: response quality and conversational behavior. With thinking enabled, Diffuplex achieves an overall score of 86.4 across the nine VoiceBench subsets, approaching the strongest leaderboard systems while finalizing an average of 10.15 tokens per forward pass on spoken AlpacaEval, compared with one token for the evaluated autoregressive baselines. Without thinking, Diffuplex exhibits responsive full-duplex behavior, with turn-taking and barge-in latency comparable to strong full-duplex models, while exceeding existing open-source systems by more than 20 points in spoken intelligence.

Our contributions are threefold. First, we present an efficient approach for adapting a pretrained text-only DLM to spoken input, preserving its linguistic capabilities while benefiting from parallel decoding; to the best of our knowledge, Diffuplex achieves the highest reported spoken question-answering score among diffusion-based spoken language models. Second, we decouple conversational control from response generation within a shared backbone: a lightweight autoregressive policy tracks streaming speech, while diffusion handles response content. This separation supports semantically informed turn-taking and interruption handling while largely preserving the linguistic capabilities learned before full-duplex training. Third, we introduce an anchored first-n decoding strategy tailored to spoken agents: it commits a short prefix early and sends it to the speech synthesizer while the rest of the response continues denoising. Speech can thus begin before the full response has converged, making diffusion generation compatible with streaming speech output.

2 Related Work

2.1 Spoken Language Models

Turn-based spoken language models extend text LLMs with speech encoders or discrete acoustic tokens, enabling unified speech understanding and generation (Wang et al., 2026a). AudioPaLM transfers linguistic knowledge from a text LLM to speech tasks, while SpeechGPT develops cross-modal instruction-following through modality adaptation and speech instruction tuning (Rubenstein et al., 2023; Zhang et al., 2023). More recent systems, such as Qwen2.5-Omni, jointly generate text and speech through specialized reasoning and synthesis modules (Xu et al., 2025a). Despite their strong linguistic capabilities, these models largely retain a turn-oriented interaction protocol and generate response content autoregressively.

Full-duplex models instead process user and agent speech as concurrent streams, allowing interruptions, overlaps, and backchannels without explicit turn segmentation (Wang et al., 2026b). Moshi models the two speakers with parallel audio-token streams, while OmniFlatten serializes multimodal dialogue events into a unified sequence (Défossez et al., 2024; Zhang et al., 2024). Freeze-Omni further connects streaming speech interfaces to a frozen text LLM to preserve its linguistic capabilities (Wang et al., 2024). PersonaPlex adds role and voice control to a full-duplex speech model, Lychee-FD addresses acoustic–semantic interference, and Listen-Write-Speak provides a visible text channel during real-time conversation (Roy et al., 2026; Liu et al., 2026; Zhang et al., 2026). These systems achieve responsive interaction, but typically bind semantic generation and conversational control to the same frame-synchronous AR process. Diffuplex instead separates these functions: an AR policy tracks conversational timing, whereas a DLM generates response content along an independent computational axis.

2.2 Diffusion Language Models with Spoken Input

Diffusion language models generate text through iterative refinement rather than strictly left-to-right decoding. Early work explored continuous diffusion for controllable text generation (Li et al., 2022), while masked discrete diffusion substantially narrowed the likelihood gap with AR models (Sahoo et al., 2024). LLaDA demonstrated that diffusion language models can scale to general-purpose pretraining and instruction following (Nie et al., 2025). Most relevant to our work, DiffusionGemma converts a pretrained AR mixture-of-experts model into a discrete DLM that refines 256-token canvases in parallel while retaining AR capabilities (DiffusionGemma Team et al., 2026).

The extension of DLMs to spoken input remains comparatively underexplored. Recent works have applied diffusion decoders to automatic speech recognition (ASR), either training a speech-conditioned diffusion decoder from scratch or attaching a frozen encoder or DLM for parallel transcript decoding (Baas et al., 2022; Kwon et al., 2025; Yen et al., 2026; Khurdula et al., 2026). DIFFA moves beyond transcription by first aligning speech through ASR and subsequently training speech-conditioned instruction following; DIFFA-2 further scales this framework to general audio understanding (Zhou et al., 2026a;b). These studies establish that DLMs can understand spoken input, but do not address full-duplex interaction or explicitly separate conversational timing from response generation. Diffuplex closes this gap by combining a streaming AR interaction policy with parallel diffusion-based response generation. Cross-modal distillation has also been explored to align the capabilities of speech models with text teachers (Cao et al., 2026).

3 Method

Overview. Diffuplex follows a three-stage training curriculum. In Stage 1, we perform speech–text modality alignment using ASR data, enabling a pretrained text diffusion language model to condition on continuous speech representations. In Stage 2, we train the model on spoken-instruction–text-response pairs to acquire instruction-following capabilities, thereby learning *what to say*. In Stage 3, we preserve the capabilities acquired in Stage 2 while introducing an AR interaction mode that learns a conversational policy, i.e., *when to speak*.

At inference time, the model operates along two decoupled computational axes. A causal AR branch continuously processes incoming speech to model conversational behavior, while a diffusion branch

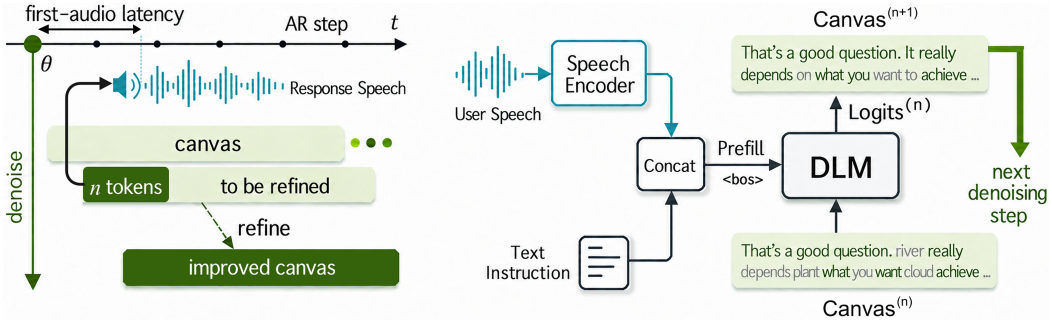


Figure 2: Overview of Diffuplex. **Left:** The horizontal axis represents real-time AR interaction, while a predicted <bos> triggers diffusion generation along the vertical axis at θ. An early refined prefix of n tokens is streamed to the speech synthesizer as the remaining canvas continues denoising. **Right:** Speech features and the text instruction are jointly prefilled into the DLM as condition, which iteratively refines each response canvas through bidirectional denoising.

generates responses conditioned on the accumulated context. In Section 3.4, we introduce denoising strategies designed to enable low-latency and fluid speech-to-speech interaction.

3.1 Stage 1: Modality Alignment

We construct the speech frontend using a Parakeet encoder (NVIDIA, 2025) followed by a randomly initialized projection layer that maps acoustic representations into the embedding space of DiffusionGemma (DiffusionGemma Team et al., 2026). Given these speech embeddings and textual instruction as conditioning context, the model reconstructs the complete ASR transcript through iterative denoising. We first warm up the projection layer on LibriSpeech (Panayotov et al., 2015), then jointly optimize the projection layer and speech encoder on an English ASR mixture drawn primarily from Granary (Rao Koluguri et al., 2025). The diffusion language model remains frozen throughout this stage, requiring the speech frontend to produce representations compatible with its pretrained textual latent space.

Auxiliary loss. Recent work suggests that an auxiliary CTC objective can improve diffusion-based ASR performance (Khurdula et al., 2026). We did not observe consistent gains in our setting and instead employ an auxiliary AR next-token loss alongside the diffusion objective. The resulting Stage 1 model achieves a word error rate (WER) of 2.3 on LibriSpeech (Panayotov et al., 2015) test-clean and 4.2 on test-other.

3.2 Stage 2: Spoken Instruction-Following

We sample 600K speech-instruction examples from the Nemotron-Omni SFT corpus (NVIDIA, 2026) and mix them with 5% ASR replay, drawn from 37.5K transcription examples. To strengthen the capability without thinking, 70% of responses directly supervise final answers without explicit <think> content. The speech encoder, projection layer, and DLM are jointly optimized using the same diffusion and auxiliary AR objectives as in Stage 1, while the MoE router remains frozen.

Joint adaptation exposes an empirical trade-off. With a small learning rate, long-form generation of DLM remains coherent, but the model only weakly grounds its responses in the spoken input. Increasing the learning rate improves speech adaptation but causes the pretrained language distribution to drift, resulting in repetition and incorrect termination. This tension cannot be addressed by setting different learning rates for three modules, or replaying text-only data.

Text-Anchored distillation. We therefore retain a frozen copy of the pretrained DLM as a textual anchor. For each paired training example, the trainable student receives the spoken request, whereas the frozen teacher receives its written counterpart. Then we minimize forward KL divergence between their output distributions for both the diffusion canvas and the causal AR readout.

This cross-modal supervision effectively transfers the anchor teacher's textual capabilities to the speech-conditioned student.

3.3 Stage 3: Full-duplex Interaction Modeling

Natural conversation does not arrive as a sequence of pre-segmented turns with explicit boundaries. We therefore learn a streaming interaction policy directly from conversational speech data. The user audio is treated as a continuous stream and divided into 80-ms chunks. At each step, the model autoregressively predicts one interaction state conditioned only on the observed speech prefix and previous states.

In Diffuplex, we represent this policy using four states: <listen>, <speak>, <bos>, and <eos>. The first two indicate whether the model should listen or retain the conversational floor, whereas <bos> and <eos> denote taking and yielding the floor, respectively. These transition events are sparse and therefore receive larger loss weights than the persistent states:

$$\mathcal{L}_{\text{policy}} = -\sum_{t} w_{s_t} \log p(s_t \mid a_{\leq t}, s_{<t}), \tag{1}$$

where $a_{\leq t}$ denotes the speech observed up to time t. As the model cannot anticipate user's behavior, the <bos> and <eos> targets include a short reaction delay that allows sufficient acoustic evidence to accumulate. The interaction state tokens are independent of the DLM vocabulary, with a trainable word embedding matrix. At each AR step, the previous state embedding is combined with the chunk-level speech representation for the next state token prediction.

Causality of encoder. Stage 3 reuses speech encoder from Stage 2 but operates under strict causal constraints: a 10-second trailing window without lookahead aligned with streaming inference. To bridge this representation shift, a lightweight LoRA adapter (Hu et al., 2021) is introduced, but is not activated when computing diffusion context. This design preserves encoding quality while meeting online constraints, without requiring a separate causal encoder or suffering train-test mismatch.

3.4 Inference and Denoise

When the AR branch emits <bos>, the speech-encoder features accumulated so far are prefilled, together with the text prompt, through the DLM's causal pass to build its key–value cache, and the response is generated as a sequence of 256-token canvases that attend to this cache. Each denoising pass is one bidirectional forward over the canvas. We largely follow the inference recipe of DiffusionGemma (DiffusionGemma Team et al., 2026): at every pass its entropy-bounded sampler selects the lowest-entropy positions whose mutual-information bound stays below an entropy bound β=0.1, keeps their sampled tokens for the next pass, and fully re-noises all non-selected positions; the sampling temperature decays linearly across the budget of at most N=48 passes. Denoising terminates early once the highest-probability predictions are identical across two consecutive passes and the average entropy over the canvas falls below a confidence threshold γ. The converged canvas is committed and appended to the cache before the next one is opened. Because the original recipe was designed for offline text generation, we depart from it in two respects that are essential for spoken interaction. First, the opening of the response is committed long before the canvas converges, so that speech synthesis can start within a few passes (*anchored prefix*). Second, the denoising schedule is shortened so that fewer passes separate consecutive speech packets (*denoising schedule*).

Anchored prefix. Under the original decoding scenario, the first word of the answer is final only when the whole canvas has converged, typically after tens of passes. The leading positions, however, settle much earlier. We therefore commit the first n=8 positions of the first canvas after at most K passes and stream them to the synthesizer while the rest keeps denoising. Position 0 is seeded with the argmax of the causal AR readout at the prefill, which is trained with the auxiliary AR loss and costs nothing extra. During the first $K-1$ passes, non-accepted front positions are refilled with the denoiser's argmax rather than noise, and a front position $i < n$ is frozen at its argmax at the first pass j satisfying

$$H\big(p_i^{(j)}\big) < \tau \quad \text{or} \quad j = K, \tag{2}$$

where $p_i^{(j)}$ is the temperature-scaled distribution at pass j and τ=0.05. Frozen positions are never re-sampled, so the first n tokens can be handed to the TTS system after one prefill and at most K

passes regardless of response length, and speech synthesis begins while the remainder of the canvas is still denoising; K trades latency against the quality of the opening (Appendix D).

Denoising schedule. The pretrained schedule (temperature $0.8 \rightarrow 0.4$, γ=0.005) spends its last passes confirming an already converged canvas. We anneal from 0.7 to 0.2 and relax γ to 0.025, and cut any response whose canvas ends in a trailing run of ≥ 32 tokens drawn from at most two distinct tokens, a degenerate loop that a sharper schedule occasionally produces. Committed text is streamed to an interleaved text–speech TTS model (Du et al., 2025), the anchored prefix forming the first packet, so synthesis of the opening overlaps with the denoising of the remainder.

The AR interaction policy and diffusion response generator share the same backbone parameters but maintain separate KV caches. The AR cache advances causally with incoming audio to support turn-taking decisions; once the policy emits <bos>, the observed speech context is separately prefilled to initialize the diffusion cache. This keeps the policy's streaming history distinct from the context used to denoise the response.

3.5 Speech Synthesizer

As the first n tokens are finalized early in the denoising process, Diffuplex is agnostic to the choice of speech synthesizer. These tokens can be immediately passed to any high-quality TTS model while the remaining canvas continues to be refined; consequently, the TTS model itself need not provide native streaming inference. In this work, we adopt CosyVoice 3 (Du et al., 2025) for its synthesis quality and voice-cloning capability. When voice cloning is unnecessary, it can be replaced with a smaller single-speaker TTS model to further reduce first-audio latency. This modular interface allows synthesis quality, speaker flexibility, and latency to be traded off without retraining Diffuplex.

User interruption. When the AR interaction policy emits <eos>, any remaining response audio is immediately truncated, allowing Diffuplex to yield naturally to the user. If the user interrupts within one second, we discard this unspoken canvas rather than appending it to the KV cache for the next interaction. Consequently, only content that has effectively entered the conversation is retained in the context, preventing frequent interruptions or hesitations from corrupting conversation context.

4 Experimental Setup

4.1 Training Details

Training follows the three stages described in Section 3. Appendix A summarizes the training data, configurations, computing resource, and optimization settings for all stages.

4.2 Benchmark and Evaluation

VoiceBench (Chen et al., 2024) evaluates the spoken intelligence of voice assistants across open-ended dialogue, knowledge and reasoning, instruction following, and safety. It comprises nine evaluation subsets with both real and synthetic speech, incorporating variations in speakers, acoustic environments, and linguistic content. The metric contains both LLM judge score and accuracy, and we report them on average for comparison.

Full-Duplex-Bench v1 (Lin et al., 2025) evaluates fundamental real-time conversational behaviors under four controlled scenarios: pause handling, backchanneling, smooth turn-taking, and user interruption. We also add the "turn-taking dynamics" track from **Full-Duplex-Bench v3** (Lin et al., 2026a), as all recording is collected from human speakers with conversational disfluencies.

4.3 Baseline

We compare Diffuplex against three classes of spoken models: (1) **Turn-based AR models** represent the dominant high-capability paradigm, and we enable thinking where supported. As the intermediate reasoning trace is not presented to the user, enabling thinking in AR models would result in relatively high first-audio latency. (2) **Turn-based diffusion models** generate response tokens through parallel iterative refinement and can therefore reduce decoding latency relative to

Table 1: VoiceBench results across turn-based AR and full-duplex models. Open-ended QA columns use a 5-point judge scale; the remaining columns and Overall use a 0–100 scale.

Model	Open-ended QA			Knowledge and Reasoning				Ins. and Safety		**Overall**
	Alp.	Com.	Wil.	SDQA	MMSU	OBQA	BBH	IFE.	Adv.	
Turn-based AR Models										
LFG-3	4.73	4.40	4.45	78.1	85.5	94.7	92.2	88.5	98.3	**89.9**
Nemotron-3 Omni	4.75	4.57	4.58	71.4	82.3	93.0	91.1	88.7	100	89.4
Qwen3-Omni-Thinking	4.82	4.53	4.53	78.1	83.0	94.3	88.9	80.6	97.2	88.8
Whisper-L + GPT-4o	4.80	4.47	4.62	75.8	81.7	93.0	87.2	76.5	98.3	87.8
GPT-4o-Audio	4.78	4.49	4.58	75.5	80.3	89.2	84.1	76.0	98.7	86.8
Kimi-Audio	4.46	3.97	4.20	63.1	62.2	83.5	69.7	61.1	100	76.9
Turn-based Diffusion Models										
DIFFA-2	3.86	3.67	2.80*	40.8	38.1	61.5	53.1*	40.8	85.6	58.5
Full-duplex AR Models										
Nemotron-3 VoiceChat	3.59	3.12	3.06	43.4	50.5	64.4	52.7	17.1	99.6	58.1
Lychee-FD	3.31	2.28	1.75	34.4	22.9	29.2	51.0	22.6	84.4	43.5
Freeze-Omni	4.03	3.46	3.15	53.5	28.1	31.0	50.7	23.4	97.3	55.2
Moshi	2.01	1.60	1.30	15.6	24.0	25.9	47.4	10.1	44.2	29.5
Stage-2										
Diffuplex *w/o* thinking	4.69	4.29	4.44	62.9	76.4	89.7	81.3	77.9	99.6	84.0
Diffuplex *w.* thinking	4.69	4.24	4.45	61.8	81.8	94.3	87.5	85.5	98.8	86.4
Stage-3, $K{=}4$										
Diffuplex (*full-duplex*)	4.65	4.29	4.30	59.3	71.9	85.9	83.1	75.6	100.0	82.3

autoregressive generation. (3) **Full-duplex AR models** interleave speech perception and generation along a fixed-rate temporal stream, enabling low first-audio latency and fluid turn-taking. Because AR steps are tied to perceived audio chunks, these systems do not have a separate thinking phase. Original publications and model cards for every compared system are listed in Appendix B.

5 Evaluation and Results

5.1 Results on Spoken Intelligence

Table 1 reports two evaluation settings for Diffuplex. Stage 2 evaluates the speech-conditioned response generator on complete spoken requests, with and without thinking. Stage 3 starts from the Stage 2 checkpoint, freezes the response generator, and adds the streaming interaction policy; its row evaluates the full-duplex system by committing the first $n{=}8$ tokens within $K{=}4$ denoising passes. Results marked with * were reproduced by us when the corresponding numbers were unavailable in the source paper (Zhou et al., 2026b).

As shown in Table 1, the Stage 2 response generator scores 84.0 without thinking and 86.4 with thinking. The largest gains from enabling thinking occur on IFEval ($+7.6$), BBH ($+6.2$), MMSU ($+5.4$), and OpenBookQA ($+4.6$), while the three open-ended QA subsets change little. This suggests that additional inference-time reasoning helps most on instruction following and knowledge-intensive questions. In the full-duplex Stage 3 setting, the non-thinking score is 82.3, 1.7 points below Stage 2. The largest decreases are on MMSU (-4.5), OpenBookQA (-3.8), and SDQA (-3.6), while BBH improves by 1.8 points. Because Stage 3 freezes the response generator, this decrease does not indicate forgetting from further language-model training. It is consistent with the change to streaming turn decisions and early-prefix decoding, although these contributions are not isolated by this comparison. Appendix D shows the measured latency–quality trade-off as K varies with $n{=}8$. Stage 3 still exceeds the strongest full-duplex AR baseline in the table by 24.2 points.

Generality across diffusion backbones. We apply the same speech adaptation and text-anchored training strategy to Nemotron-Labs-Diffusion-8B (Fu et al., 2026). The resulting speech-conditioned model reaches a VoiceBench overall score of approximately 78.7 with thinking enabled. This result shows that the strategy transfers to a second diffusion backbone; we attribute the lower score primarily to its weaker text-model capability relative to DiffusionGemma.

Table 2: Conversation behavior results."TOR-v1" and "TOR-v3" denote the Full-Duplex-Bench v1 and v3 (with disfluency), respectively. "Syn." and "Cand." stand for Synthetic and Candor audio. Closed-source models (e.g., GPT-Realtime) are excluded due to API-induced latency overheads.

Model	Turn Taking			User Interruption		Pause Handling	
	TOR-v1 ↑	TOR-v3 ↑	Latency ↓	TOR ↑	Latency ↓	Syn. ↓	Cand. ↓
Freeze-Omni	0.336	0.312	0.953	0.867	1.409	0.642	0.481
Moshi	0.941	0.925	0.265	1.000	0.257	0.985	0.980
PersonaPlex	0.992	0.985	0.070	1.000	0.400	0.584	0.662
Raon-SpeechChat	0.832	0.810	1.034	0.980	1.219	0.212	0.299
Nemotron-VC	0.820	0.798	0.448	1.000	0.480	0.153	0.255
Lychee-FD	0.983	0.971	0.840	0.945	-	0.100	-
Diffuplex (*w.* Fisher)	0.933	0.980	0.396	0.945	0.239	0.679	0.667
Diffuplex	1.000	0.990	0.536	1.000	0.691	0.380	0.431

5.2 Results on Conversation Behavior

We report two variants of our model because we find that adding the Fisher conversational corpus (Cieri et al., 2004) into Stage 3 training noticeably changes the learned conversation policy. Including Fisher favors aggressive, low-latency turn transitions; without Fisher, the model focuses on semantically informed turn-taking. The remaining baselines are representative full-duplex spoken dialogue models proposed in recent years. We evaluate all systems on turn-taking, user interruption, and pause handling, where higher TOR and lower latency or pause-handling scores indicate better performance. First-audio latency spans the Stage 3 AR turn decision, response denoising, and TTS.

As shown in Table 2, Diffuplex achieves the strongest turn-taking accuracy on both benchmark versions while correctly responding to user interruptions. Diffuplex with Fisher substantially reduces both turn-taking and interruption latency, but this responsiveness comes at the cost of weaker pause handling. A similar pattern appears in Moshi and PersonaPlex, both of which use Fisher-style conversational data: they enter turns rapidly, yet are more likely to interpret a temporary user pause as the end of a turn. These results expose a fundamental trade-off between aggressive response timing and semantic turn understanding. Training on fast-paced conversational data encourages the model to respond to acoustic silence, whereas excluding such data allows Diffuplex to wait for stronger semantic evidence before taking the floor. Our two variants therefore provide distinct operating points between interaction latency and robustness to natural hesitation and disfluency.

5.3 TPF and TPS Analysis

We measure decoding efficiency on spoken AlpacaEval, whose relatively long responses make generation differences easier to observe. Figure 3 places tokens finalized per forward pass (**TPF**) and decoded tokens per second (**TPS**) alongside active parameter counts and AlpacaEval scores. All models use the same audio and system/user prompts, batch 1, bf16 weights and KV cache, on one NVIDIA RTX 6000 GPU. We enable thinking where available; DIFFA-2 has no thinking mode. Appendix C explains our choice of measurement hardware.

Figure 3 shows that Diffuplex finalizes 10.15 tokens per forward pass, compared with 1.00 for every evaluated AR baseline and 0.37 for DIFFA-2. Its throughput reaches 364 tokens/s, versus 192 for Nemotron-3 Omni and 147 for Qwen3-Omni-Thinking, while requiring 4B active parameters compared with 3B for each of those two models. The accompanying AlpacaEval scores place this efficiency comparison alongside spoken response quality. The throughput gain is smaller than the TPF gain because each diffusion pass processes a 256-token canvas and costs more than a single-token AR step. Thus, the figure shows both the pass-level parallelism and the realized decoding speed, with model size and response quality visible in the same comparison.

5.4 Ablation and Discussion

We ablate three components that are central to transferring the capabilities of a text DLM to spoken interaction. Removing Stage 1 tests whether explicit speech–text alignment is necessary before

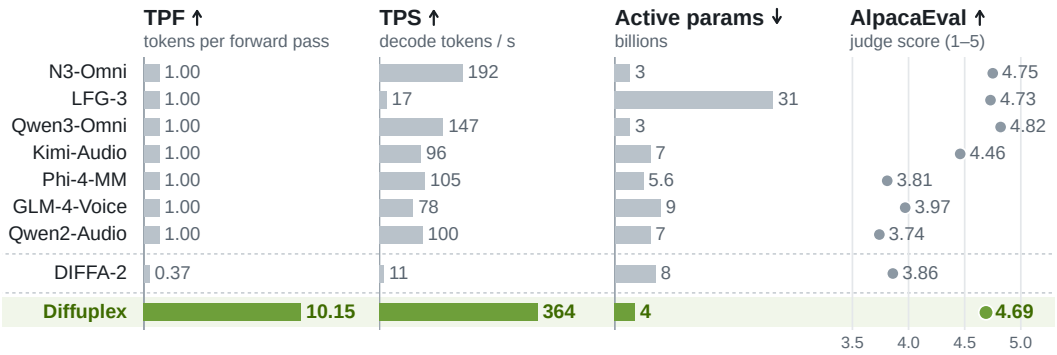


Figure 3: Decoding efficiency and score context on spoken AlpacaEval. TPF and TPS are measured at batch 1 with bf16 weights and KV cache on one NVIDIA RTX 6000 GPU.

Table 3: Ablation study on VoiceBench. "*text input*" denotes that user input is text as the test set has paired transcript. Thinking is disabled for all models.

Setting	Open-ended QA			Knowledge and Reasoning				Ins. and Safety		Overall
	Alp.	Com.	Wil.	SDQA	MMSU	OBQA	BBH	IFE.	Adv.	
Diffuplex	4.69	4.29	4.44	62.9	76.4	89.7	81.3	77.9	99.6	84.0
Diffuplex (*text input*)	4.81	4.53	4.81	73.4	84.7	93.8	94.8	81.3	100	90.1
w/o text anchor	3.96	3.25	3.05	48.4	54.2	64.6	61.5	61.3	100	66.1
w/o Stage 1	1.00	1.00	1.01	0.0	24.3	31.2	44.8	15.1	93.8	29.9
w/o trainable DLM	4.02	3.16	3.51	50.0	52.8	62.5	56.2	49.8	96.9	64.7

instruction tuning, while removing the text anchor evaluates whether the frozen text teacher prevents semantic drift during speech-conditioned adaptation. We additionally freeze the DLM throughout Stage 2 to examine whether adapting only the speech encoder and projection layer is sufficient. The text-input setting serves as an upper-bound diagnostic that separates the model's underlying linguistic capability from errors introduced by speech conditioning.

As shown in Table 3, the text-input model reaches an overall score of 90.1, while Diffuplex retains 84.0 with spoken input, indicating that most of the underlying language capability is successfully transferred to speech. Removing Stage 1 causes performance to collapse to 29.9, confirming that instruction tuning alone cannot reliably ground the DLM in acoustic representations. Removing the text anchor also reduces the score substantially, from 84.0 to 66.1, demonstrating its importance in preserving the response distribution of the pretrained text model. Finally, keeping the DLM frozen yields only 64.7, suggesting that frontend alignment alone is insufficient and that controlled adaptation of the language backbone remains necessary for strong spoken instruction following. Appendix D separately ablates the pass cap K for committing the first n tokens and reports its effect on first-audio latency and VoiceBench scores.

6 Conclusion

We introduced Diffuplex, a full-duplex spoken language model that decouples conversational control from semantic generation. Diffuplex utilize AR decoding to continuously determine *when to speak*, while a speech-conditioned DLM independently generates *what to say* through parallel refinement. This separation preserves the linguistic capabilities of a strong text model without binding semantic computation to the physical audio frame rate. Together with anchored-prefix decoding and modular speech synthesis, Diffuplex enables low-latency interaction while retaining strong spoken intelligence. Experiments across spoken understanding and full-duplex dialogue benchmarks demonstrate that Diffuplex substantially narrows the capability gap between responsive full-duplex systems and intelligent turn-based models. We hope this two-axis formulation provides a useful foundation for future spoken agents that are both conversationally natural and cognitively capable.

REFERENCES

Matthew Baas, Kevin Eloff, and Herman Kamper. TransFusion: Transcribing speech with multinomial diffusion. In *Artificial Intelligence Research (SACAIR)*, 2022.

Jagadeesh Balam, Travis Bartley, Edresson Casanova, Sanjay Chauhan, Chen Chen, et al. NemotronLabs VoiceChat: An open full-duplex speech-to-speech model with tool calling capabilities, 2026. URL https://arxiv.org/abs/2609.21967.

Di Cao, Dongjie Fu, Hai Yu, Siqi Zheng, Xu Tan, and Tao Jin. X-OPD: Cross-modal on-policy distillation for capability alignment in speech LLMs, 2026. URL https://arxiv.org/abs/2603.24596.

Gregg A. Castellucci, Christopher K. Kovach, Matthew A. Howard, Jeremy D. W. Greenlee, and Michael A. Long. A speech planning network for interactive language use. *Nature*, 602:117–122, 2022. doi: 10.1038/s41586-021-04270-z.

Yiming Chen, Xianghu Yue, Chen Zhang, Xiaoxue Gao, Robby T. Tan, and Haizhou Li. VoiceBench: Benchmarking LLM-based voice assistants, 2024. URL https://arxiv.org/abs/2410.17196.

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, 2024. URL https://arxiv.org/abs/2407.10759.

Christopher Cieri, David Miller, and Kevin Walker. The fisher corpus: A resource for the next generations of speech-to-text. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC)*, 2004. URL https://www.ldc.upenn.edu/sites/default/files/lrec2004-fisher-corpus_0.pdf.

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

DiffusionGemma Team, Adrien Ali Taïga, James Assiene, Daniele Calandriello, Rahma Chaabouni, João Gante, Tamara von Glehn, Nate Keating, Chris Knutsen, Martin Kukla, Tianlin Liu, et al. DiffusionGemma technical report, 2026.

Zhihao Du, Changfeng Gao, Yuxuan Wang, Fan Yu, Tianyu Zhao, Hao Wang, Xiang Lv, Hui Wang, Xian Shi, Keyu An, et al. CosyVoice 3: Towards in-the-wild speech generation via scaling-up and post-training, 2025. URL https://arxiv.org/abs/2505.17589.

Harishchandra Dubey, Ashkan Aazami, Vishak Gopal, Babak Naderi, Sebastian Braun, Ross Cutler, Alex Ju, Mehdi Zohourian, Min Tang, Hannes Gamper, Mehrsa Golestaneh, and Robert Aichner. ICASSP 2023 deep noise suppression challenge, 2023. URL https://arxiv.org/abs/2303.11510.

Yonggan Fu, Lexington Whalen, Abhinav Garg, Chengyue Wu, Maksim Khadkevich, Nicolai Oswald, Enze Xie, Daniel Egert, Sharath Turuvekere Sreenivas, Shizhe Diao, Chenhan Yu, Ye Yu, Weijia Chen, Sajad Norouzi, Jingyu Liu, Shiyi Lan, Ligeng Zhu, Jin Wang, Jindong Jiang, Morteza Mardani, Mehran Maghoumi, Song Han, Ante Jukić, Nima Tajbakhsh, Jan Kautz, and Pavlo Molchanov. Nemotron-Labs-Diffusion: A tri-mode language model unifying autoregressive, diffusion, and self-speculation decoding, 2026. URL https://arxiv.org/abs/2607.05722.

glenn2. LFG-3 model card. Hugging Face model card, 2026. URL https://huggingface.co/glenn2/LFG-3.

Fabian Gloeckle, Badr Youbi Idrissi, Baptiste Rozière, David Lopez-Paz, and Gabriel Synnaeve. Better & faster large language models via multi-token prediction. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 15706–15734. PMLR, 2024.

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, 2021. URL https://arxiv.org/abs/2106.09685.

Ke Hu, Ehsan Hosseini-Asl, Chen Chen, Edresson Casanova, Subhankar Ghosh, Piotr Żelasko, Zhehuai Chen, Jason Li, Jagadeesh Balam, and Boris Ginsburg. SALM-Duplex: Efficient and direct duplex modeling for speech-to-speech language model, 2025. URL https://arxiv.org/abs/2505.15670.

Yuxuan Hu, Heng Lu, Ruchao Fan, Yao Qian, Xiaofei Wang, Jian Xue, Heming Wang, Shuohang Wang, Young Jin Kim, Yelong Shen, and Jinyu Li. Preserving speech-to-text LLM capabilities in speech-to-speech generation, 2026. URL https://arxiv.org/abs/2606.30944.

Harsha Vardhan Khurdula, Abhinav Kumar Singh, Yoeven D. Khemlani, and Vineet Agarwal. Audio-native speech recognition with a frozen discrete-diffusion language model, 2026.

Beomsoo Kim, Changho Choi, Dohyun Kim, Dongki Lee, et al. Raon-Speech technical report, 2026. URL https://arxiv.org/abs/2605.23912.

KimiTeam, Ding Ding, Zeqian Ju, Yichong Leng, Songxiang Liu, et al. Kimi-Audio technical report, 2025. URL https://arxiv.org/abs/2504.18425.

Taeyoun Kwon, Junhyuk Ahn, Taegeun Yun, Heeju Jwa, Yoonchae Choi, Siwon Park, Jongchan Kim, Hyungon Ryu, Hyuk-Jae Lee, and Nam-Joon Kim. Whisfusion: Parallel ASR decoding with masked diffusion. *arXiv preprint arXiv:2508.07048*, 2025.

Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B. Hashimoto. Diffusion-LM improves controllable text generation. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://arxiv.org/abs/2205.14217.

Guan-Ting Lin, Jiachen Lian, Tingle Li, Qirui Wang, Gopala Anumanchipalli, Alexander H. Liu, and Hung-yi Lee. Full-Duplex-Bench: A benchmark to evaluate full-duplex spoken dialogue models on turn-taking capabilities, 2025. URL https://arxiv.org/abs/2503.04721.

Guan-Ting Lin, Chen Chen, Zhehuai Chen, and Hung-yi Lee. Full-Duplex-Bench-v3: Benchmarking tool use for full-duplex voice agents under real-world disfluency, 2026a. URL https://arxiv.org/abs/2604.04847.

Qingjian Lin, Yuxin Li, Haoyang Zhang, Jun Chen, Yechang Huang, Feng Tian, Xie Li, Xiangyu Tony Zhang, Daijiao Liu, Yuxin Zhang, Jinglan Gong, Bo Zhao, Fei Tian, Xuerui Yang, Gang Yu, Xiangyu Zhang, and Daxin Jiang. ParaASR: Multi-token prediction for fast and long-context LLM-based speech recognition, 2026b. URL https://arxiv.org/abs/2607.29279.

Zhenyu Liu, Xuanyu Zhang, Yunxin Li, Qixun Teng, Shenyuan Jiang, Haolan Chen, Mingjun Zhao, Fanbo Meng, Yu Xu, Yancheng He, Baotian Hu, Haizhou Li, and Min Zhang. Hierarchical acoustic-semantic modeling: Modality separation and semantic coherence for full-duplex SLMs. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics*, pp. 9264–9280, 2026. doi: 10.18653/v1/2026.acl-long.419. URL https://aclanthology.org/2026.acl-long.419/.

Microsoft, Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, et al. Phi-4-Mini technical report: Compact yet powerful multimodal language models via mixture-of-LoRAs, 2025. URL https://arxiv.org/abs/2503.01743.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models, 2025.

NVIDIA. Parakeet speech recognition models. NVIDIA NeMo documentation, 2025. URL https://docs.nvidia.com/nemo-framework/user-guide/latest/nemotoolkit/asr/models.html.

NVIDIA. Nemotron 3 Nano Omni: Efficient and open multimodal intelligence, 2026. URL https://arxiv.org/abs/2604.24954.

OpenAI. GPT-4o system card. OpenAI technical report, 2024. URL https://openai.com/index/gpt-4o-system-card/.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. LibriSpeech: An ASR corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210, 2015. doi: 10.1109/ICASSP.2015.7178964.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision, 2022. URL https://arxiv.org/abs/2212.04356.

Nithin Rao Koluguri, Monica Sekoyan, George Zelenfroynd, Sasha Meister, Shuoyang Ding, Sofia Kostandian, He Huang, Nikolay Karpov, Jagadeesh Balam, Vitaly Lavrukhin, Yifan Peng, Sara Papi, Marco Gaido, Alessio Brutti, and Boris Ginsburg. Granary: Speech recognition and translation dataset in 25 european languages, 2025. URL https://arxiv.org/abs/2505.13404.

Rajarshi Roy, Jonathan Raiman, Sang-gil Lee, Teodor-Dumitru Ene, Robert Kirby, Sungwon Kim, Jaehyeon Kim, and Bryan Catanzaro. PersonaPlex: Voice and role control for full duplex conversational speech models, 2026. URL https://arxiv.org/abs/2602.06053.

Paul K. Rubenstein, Chulayuth Asawaroengchai, Duc Dung Nguyen, Ankur Bapna, Zalán Borsos, Félix de Chaumont Quitry, Peter Chen, Dalia El Badawy, Wei Han, Eugene Kharitonov, Hannah Muckenhirn, Dirk Padfield, James Qin, Danny Rozenberg, Tara Sainath, Johan Schalkwyk, Matt Sharifi, Michelle Tadmor Ramanovich, Marco Tagliasacchi, Alexandru Tudor, Mihajlo Velimirović, Damien Vincent, Jiahui Yu, Yongqiang Wang, Vicky Zayats, Neil Zeghidour, Yu Zhang, Zhishuai Zhang, Lukas Zilka, and Christian Frank. AudioPaLM: A large language model that can speak and listen, 2023.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T. Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://arxiv.org/abs/2406.07524.

Joachim Thiemann, Nobutaka Ito, and Emmanuel Vincent. The diverse environments multi-channel acoustic noise database (DEMAND): A database of multichannel environmental noise recordings. *Proceedings of Meetings on Acoustics*, 19(1):035081, 2013. doi: 10.1121/1.4799597. URL https://doi.org/10.1121/1.4799597.

Siyin Wang, Zengrui Jin, Changli Tang, Qiujia Li, Bo Li, Chen Chen, Yuchen Hu, Wenyi Yu, Yixuan Li, Jimin Zhuang, et al. Towards general auditory intelligence for machine listening and speaking. *Nature Machine Intelligence*, pp. 1–20, 2026a.

Siyin Wang, Wenyi Yu, Xianzhao Chen, Xiaohai Tian, Jun Zhang, Lu Lu, Yuxuan Wang, and Chao Zhang. End-to-end listen, look, speak and act. In *International Conference on Learning Representations*, pp. 66804–66829, 2026b.

Xiong Wang, Yangze Li, Chaoyou Fu, Yunhang Shen, Lei Xie, Ke Li, Xing Sun, and Long Ma. Freeze-Omni: A smart and low-latency speech-to-speech dialogue model with frozen LLM, 2024.

Yuhao Wang, Heyang Liu, Ziyang Cheng, Ronghua Wu, Qunshan Gu, Yanfeng Wang, and Yu Wang. VocalNet: Speech LLMs with multi-token prediction for faster and high-quality generation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 19584–19601, Suzhou, China, 2025.

Donghang Wu, Tianyu Zhang, Yuxin Li, Hexin Liu, Chen Chen, Eng Siong Chng, and Yoshua Bengio. The silent thought: Modeling internal cognition in full-duplex spoken dialogue models via latent reasoning, 2026. URL https://arxiv.org/abs/2603.17837.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. Qwen2.5-Omni technical report, 2025a.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, et al. Qwen3-Omni technical report, 2025b. URL https://arxiv.org/abs/2509.17765.

Hao Yen, Pin-Jui Ku, Ante Jukić, and Sabato Marco Siniscalchi. MDM-ASR: Bridging accuracy and efficiency in ASR with diffusion-based non-autoregressive decoding. In *Proceedings of Interspeech*, 2026.

Aohan Zeng, Zhengxiao Du, Mingdao Liu, Kedong Wang, Shengmin Jiang, Lei Zhao, Yuxiao Dong, and Jie Tang. GLM-4-Voice: Towards intelligent and human-like end-to-end spoken chatbot, 2024. URL https://arxiv.org/abs/2412.02612.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. SpeechGPT: Empowering large language models with intrinsic cross-modal conversational abilities, 2023.

Luoyuan Zhang, Bokai Xu, Junbo Cui, Weiyue Sun, Yingjing Xu, Hanyu Liu, and Yuan Yao. Liberating LLM capabilities in full-duplex speech models, 2026. URL https://arxiv.org/abs/2606.07547.

Qinglin Zhang, Luyao Cheng, Chong Deng, Qian Chen, Wen Wang, Siqi Zheng, Jiaqing Liu, Hai Yu, Chaohong Tan, Zhihao Du, and Shiliang Zhang. OmniFlatten: An end-to-end GPT model for seamless voice conversation, 2024.

Jiaming Zhou, Hongjie Chen, Shiwan Zhao, Jian Kang, Jie Li, Enzhi Wang, Yujie Guo, Haoqin Sun, Hui Wang, Aobo Kong, Yong Qin, and Xuelong Li. DIFFA: Large language diffusion models can listen and understand. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 35112–35120, 2026a. doi: 10.1609/aaai.v40i41.40817.

Jiaming Zhou, Xuxin Cheng, Shiwan Zhao, Yuhang Jia, Cao Liu, Ke Zeng, Xunliang Cai, and Yong Qin. DIFFA-2: A practical diffusion large language model for general audio understanding. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 4789–4805, San Diego, California, United States, 2026b. Association for Computational Linguistics. doi: 10.18653/v1/2026.findings-acl.235. URL https://aclanthology.org/2026.findings-acl.235/.

A Training Details

We train the speech-conditioned response model in three phases: Stage 1a, Stage 1b, and Stage 2. The backbone is DiffusionGemma-26B-A4B-it (DiffusionGemma Team et al., 2026), paired with a Parakeet speech encoder (NVIDIA, 2025) and a projection layer that maps speech features into the backbone's embedding space. These response-model phases use a 256-token diffusion canvas and an auxiliary autoregressive (AR) next-token objective. Audio is converted to 16-kHz mono waveforms before feature extraction. The training seed is 42.

A.1 Stage 1a: Projector Warm-up

We train the projection layer while freezing the speech encoder and language model. The data comprise the LibriSpeech (Panayotov et al., 2015) train-clean-100, train-clean-360, and train-other-500 partitions, totaling 281,099 utterances (approximately 960 hours). Targets are lowercase transcripts without punctuation. We train for two epochs, or 1,098 optimizer updates, with global batch size 512. The AR loss has weight 0.1 relative to the diffusion loss. We use AdamW with peak learning rate 3×10^{-4}, 28 warm-up updates, and cosine decay to 3×10^{-5}.

A.2 Stage 1b: Speech Encoder Adaptation

Starting from Stage 1a, we unfreeze the speech encoder and continue training it with the projector while keeping the language model frozen. The data are a 32-source English ASR mixture drawn primarily from Granary (Rao Koluguri et al., 2025), with additional internal ASR and LibriSpeech data. Unlike Stage 1a, the transcription targets retain punctuation and capitalization. We train for

4,032 updates with global batch size 512, corresponding to approximately 2.06 million sampled presentations. The AR loss weight is 0.05. We use AdamW with peak learning rate 2×10^{-5}, 403 warm-up updates, and cosine decay to 10^{-5}.

A.3 Stage 2: Spoken Instruction Tuning

Starting from the Stage 1b speech frontend, Stage 2 adapts the model to spoken requests. Its training pool contains 600,000 speech-instruction examples selected from Nemotron-Omni SFT (NVIDIA, 2026) and 37,500 ASR replay examples. The two tasks are sampled with probabilities 95% and 5%. We optimize the speech encoder, projector, and diffusion language model, keeping the MoE router frozen. The diffusion and AR loss weights are both 1.0. To retain the pretrained model's textual capabilities, a frozen copy of the language model receives the written form of each speech-instruction example; forward KL losses between teacher and student are applied to the diffusion canvas and AR readout with weights 3.0 and 1.0, respectively.

Stage 2 uses a maximum padded sequence length of 4,096 tokens and global batch size 128. We use AdamW with peak learning rates 10^{-5} for the speech frontend and 3×10^{-6} for the language model, 200 warm-up updates, and cosine decay. The selected joint-training checkpoint is after 2,266 optimizer updates. Stages 1a, 1b, and 2 use 64 GPUs with BF16 computation, FP32 optimizer parameters, and gradient-norm clipping at 1.0.

A.4 Stage 3: Full-duplex Interaction Training

Data. The main Stage 3 mixture combines 120 hours of clean synthetic conversations with 70 hours of internal conversational corpus. The clean data provide clearly defined turns and pauses; the internal data add natural hesitations, backchannels, and user interruptions. We also train a variant with Fisher telephone conversations (Cieri et al., 2004) to expose the policy to faster, spontaneous turn exchanges. For the noise-augmented recipe, we add stationary fan and office ambience selected from DNS5 (Dubey et al., 2023) and DEMAND (Thiemann et al., 2013) to the clean synthetic recordings. We also mix quiet pink/white noise and short nonspeech events such as doors, typing, and chair movement. A small fraction of the recordings retains its original silence.

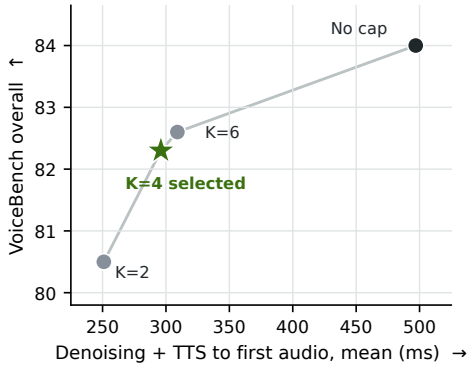
Optimization. Starting from the Stage 2 checkpoint, we train the four-state interaction head and a listening-mode LoRA adapter (Hu et al., 2021), while keeping the response generator fixed. The policy is trained with cross-entropy on 80-ms state labels, with extra weight on the sparse start- and stop-speaking transitions. We use AdamW with peak learning rate 10^{-3}, 50 warm-up updates, and cosine decay to 10^{-4} over 3,000 updates. Global batch size is 32 on two nodes with eight GPUs each.

B Compared Models and Their Sources

We cite the original reports or public model cards for every model used in the comparisons. The systems address different operating regimes, and their reported scores and latencies should be read together with the evaluation settings in Sections 5.1–5.3.

Turn-based speech and audio models. LFG-3 is documented in its public model card (glenn2, 2026). Nemotron-3 Omni (NVIDIA, 2026), Qwen3-Omni-Thinking (Xu et al., 2025b), and Kimi-Audio (KimiTeam et al., 2025) provide the other high-capability comparisons in Table 1. The Whisper-L + GPT-4o cascade uses Whisper (Radford et al., 2022) and GPT-4o (OpenAI, 2024); the GPT-4o-Audio row is likewise attributed to GPT-4o's audio-capable system (OpenAI, 2024). The efficiency comparison in Figure 3 additionally includes Phi-4-Multimodal (Microsoft et al., 2025), GLM-4-Voice (Zeng et al., 2024), and Qwen2-Audio (Chu et al., 2024). These models differ in parameter activation and speech-output design, which is why Figure 3 displays active parameters and a quality measure alongside decoding speed.

Diffusion-based models. DIFFA-2 (Zhou et al., 2026b) is the speech-conditioned diffusion baseline. Our response backbone is DiffusionGemma (DiffusionGemma Team et al., 2026); the



Time to commit the first eight tokens

Pass cap	Passes	LM prefix mean / med	First audio mean / med
K=2	2.0	83 / 83	251 / 252
K=4	**3.8**	**127 / 129**	**296 / 297**
K=6	4.3	141 / 131	309 / 300
No cap	11.5	328 / 195	497 / 363

Times in ms; Stage 3 AR decision delay is excluded.

Figure 4: Measured latency–quality trade-off for committing the first $n{=}8$ tokens. Latency is measured from a supplied turn boundary to first audio, **excluding** the Stage 3 AR turn-decision delay. The left panel shows mean post-trigger latency and VoiceBench overall score; the right panel reports passes to prefix commitment and mean/median timing. The green star marks the reported $K{=}4$ configuration; "No cap" uses the stock decoder.

backbone-transfer experiment uses Nemotron-Labs-Diffusion (Fu et al., 2026). These citations credit the respective model authors independently of our speech adaptation and evaluation.

Full-duplex systems. The conversational baselines include Moshi (Défossez et al., 2024), Freeze-Omni (Wang et al., 2024), PersonaPlex (Roy et al., 2026), Lychee-FD (Liu et al., 2026), Raon-SpeechChat (Kim et al., 2026), and Nemotron VoiceChat (Balam et al., 2026); "Nemotron-VC" is the abbreviated label in Figure 1. Figure 1 also includes SALM-Duplex (Hu et al., 2025) and FLAIR (Wu et al., 2026). Our implementation uses a Parakeet speech encoder (NVIDIA, 2025) and CosyVoice 3 speech synthesis (Du et al., 2025). Each system is included to show a distinct point in the space of spoken intelligence, decoding speed, or conversational responsiveness.

C Efficiency Measurement Hardware

We measure single-GPU decoding efficiency on an NVIDIA RTX 6000 rather than relying on H100-, GB200-, or GB300-class server accelerators. Our motivation is deployment relevance: spoken assistants may ultimately need to run under the tighter compute and power budgets of portable devices, so a workstation-class GPU offers a more accessible reference point than a large server accelerator. The RTX 6000 is still substantially different from a portable-device processor; these measurements compare decoding methods on common hardware and do not claim on-device deployment or predict mobile latency. Faster server accelerators could lower absolute times, but would not change the hardware used for the comparisons reported here.

D Sensitivity of Early Prefix Commitment

We fix the committed prefix length at $n{=}8$ tokens and vary the maximum number of denoising passes K before that prefix is finalized. The reported Stage 3 setting uses $K{=}4$; the uncapped decoder uses the Stage 2 decoding configuration in a separate evaluation run. To isolate the effect of the decoding schedule, we supply the turn boundary and measure the time from that boundary to first audio. These measured post-trigger times include denoising and TTS but **exclude** the Stage 3 AR turn-decision delay.

With $K{=}2$, mean post-trigger first-audio latency is 251 ms and the overall score is 80.5. At $K{=}4$, the first eight tokens are fixed after 3.8 passes on average; latency is 296 ms (297 ms median) and the score rises to 82.3. Increasing the cap to $K{=}6$ yields 309 ms and 82.6, while uncapped decoding takes 497 ms and scores 84.0. Across these settings, higher K is associated with better VoiceBench scores and longer post-trigger latency. Figure 5 gives the corresponding subset scores.

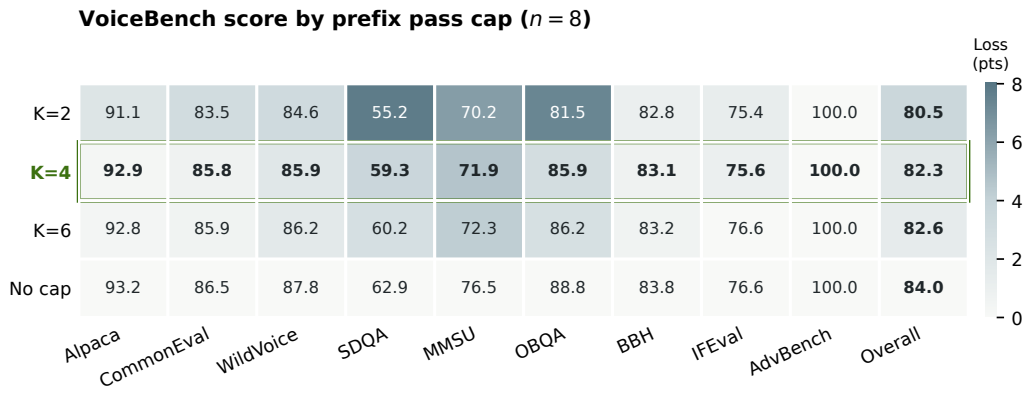


Figure 5: VoiceBench subset scores under the same n=8 prefix setting. Printed values are absolute scores on the 0–100 scale; cell shading indicates the loss relative to unconstrained decoding in the same column. The green outline identifies the K=4 setting used for the Stage 3 result.

E More Discussion

Can Diffuplex generate autoregressively? Yes. The backbone retains a causal AR readout, and the auxiliary AR objective used during speech adaptation allows the same speech-conditioned model to generate a response token by token. In our evaluation, this AR path achieves spoken-response quality comparable to diffusion decoding and can be slightly better in some settings. Its limitation is generation efficiency: each AR backbone step finalizes one answer-text token, whereas diffusion can commit multiple positions in parallel. The AR readout can also serve as an optional verifier for the first n=8 tokens proposed by the diffusion path. Verifying this prefix improves the VoiceBench overall score by approximately one point relative to early-prefix decoding without verification, at the cost of additional sequential computation.

Can Stage 3 be trained without the listening-mode LoRA? The interaction head can be trained without the adapter, but the resulting policy mainly learns an acoustic voice-activity detector: it reacts to speech and silence while failing to distinguish a within-turn hesitation from a true turn ending. In particular, it does not handle pauses reliably. The listening-mode LoRA gives the causal speech pathway enough flexibility to use conversational context for turn decisions while the diffusion response pathway remains fixed.

Why is there no direct multi-token-prediction (MTP) baseline? Our efficiency metric counts finalized *answer-text* tokens per backbone forward pass. Diffuplex reaches 10.15 on spoken AlpacaEval, above the fixed proposal horizons of representative text MTP configurations, such as four-token prediction (Gloeckle et al., 2024); this is a comparison with those configurations, not a universal upper bound on MTP. Among the speech-input MTP studies most relevant to our setting, ParaASR applies a six-token proposal to acoustically constrained transcription (Lin et al., 2026b), whereas VocalNet and PRIME-Speech apply MTP to *speech or codec tokens* rather than parallelizing the answer-text stream (Wang et al., 2025; Hu et al., 2026). Their output units and tasks differ from open-ended spoken question answering, so their reported token rates cannot be placed directly alongside our answer-text TPF. A promising solution is using MTP to replace the early prefix in Diffuplex, which might result in lower first-audio latency. We view it as our future work.

F Limitations

Our current decoder does not yet exploit the full potential of parallel token commitment. Although Diffuplex finalizes 10.15 tokens per forward pass on spoken AlpacaEval, we have not systematically optimized the denoising schedule, token-commitment rule, or implementation to maximize TPF; the reported TPF should therefore not be viewed as the ceiling of this approach. Our training and evaluation also focus on human speech, so we do not establish performance on non-speech audio,

including animal vocalizations such as dog barks, environmental sounds, and music. Autoregressive audio-language models have obtained broader audio understanding by including diverse sound and music data in training (Chu et al., 2024; KimiTeam et al., 2025). We expect that adding comparable non-speech audio data and supervision could extend Diffuplex to these tasks, but this remains an empirical question.