Stabilizing Instruction Supervision for Instruct-TTS via Controllable Diversification and Drift Filtering

Yizhong Geng[1], *Kecan Mao*[1], *Qifei Li*[1], *Cong Wang*[1], *Yingming Gao*[1], *Ruimin Wang*[2], *Chunfeng Wang*[2], *Hao Li*[2], *Ya Li*[1,**]

[1] Beijing University of Posts and Telecommunications, China [2] Li Auto, China

yzgeng@bupt.edu.cn

Abstract

Instruct-TTS systems expand structured style labels into natural-language training instructions through LLM rewriting, yet we find that over 40% of unconstrained rewrites contain semantic drift that corrupts supervision and weakens generalization. We formalize this problem as instruction supervision instability and propose a data-centric stabilization recipe that jointly improves coverage and fidelity through three mechanisms: controllable instruction diversification for systematic expansion, LLM-based drift filtering for quality control, and attribute-aligned supervision that grounds prosody control in acoustic perturbations. On the Chinese split of InstructTTSEval, our recipe raises instruction-following from 34.5% without fine-tuning and 51.0% with naive fine-tuning to 56.4%, while constrained rewriting reduces drift from 40.4% to 15.4%. Ablations confirm the three mechanisms are complementary, and the drift taxonomy may generalize to instruction-driven generation beyond TTS. [1]

Index Terms: Instruct-TTS, Label-to-instruction, Instruction instability, Drift taxonomy, Data-centric stabilization

1. Introduction

Text-to-speech synthesis is shifting from fixed tags and numeric controls to Instruct-TTS, where users specify style, emotion, and prosody through free-form natural language [1, 2, 3, 4]. While this interface broadens applicability in audiobooks, assistants, and content creation, it hinges on large-scale instruction–speech supervision with broad coverage and high semantic fidelity [5], which is difficult to collect at scale.

A common workaround is to convert structured style or attribute labels into natural-language instructions using templates or LLM rewriting [1, 5, 2], a process we refer to as the *label-to-instruction* pipeline. In NLP, Self-Instruct and Evol-Instruct have shown that LLM-generated instructions can effectively train instruction-following models [5, 6], and LLM-as-judge methods are widely adopted for data filtering [7, 8]. However, these pipelines largely assume that LLM output quality is manageable through standard prompting. In the TTS setting, the label-to-instruction pipeline suffers from two coupled weaknesses. First, generated prompts are homogeneous in phrasing and viewpoint, providing *limited coverage* of real user expressions [5, 6]. Second, LLM rewrites are prone to *semantic drift*: they silently change the intended control meaning [6, 9, 10], e.g., producing an execution-like utterance instead of a control instruction, adopting an in-character persona, or altering seed attributes (Fig. 1). These two weaknesses together corrupt training signals, which may explain why naive fine-tuning on such data can underperform a no-SFT baseline [11, 12].

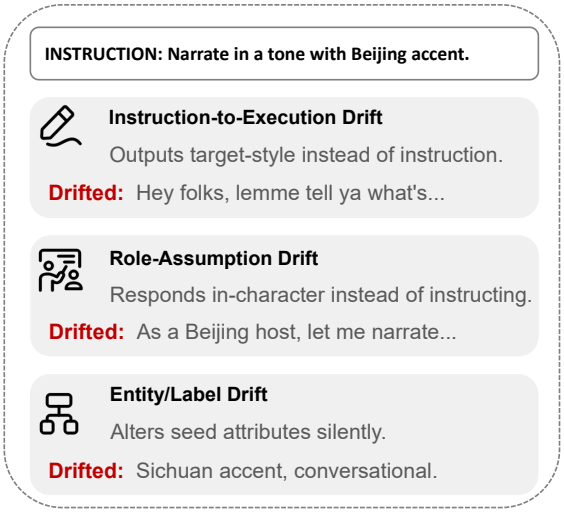

Figure 1: *Three drift types in LLM-rewritten instructions: a drifted output versus the expected faithful instruction from the same seed attributes.*

We unify the two weaknesses above under the concept of *instruction supervision instability* and argue that it is fundamentally a data quality problem rather than a model architecture limitation: the model itself is capable of following instructions, but the training signal is too noisy and narrow to teach it reliably. This perspective suggests a *data-centric* remedy: improving the instruction data along both coverage and fidelity dimensions simultaneously. To quantify the problem, we annotate LLM-expanded instructions and identify three recurring drift patterns—*instruction-to-execution*, *role-assumption*, and *entity/label* drift—finding that over 40% of unconstrained rewrites are supervision-corrupting. Crucially, expanding diversity alone does not ensure fidelity, and filtering alone does not expand coverage, so the two must be addressed jointly.

Guided by this analysis, we assemble a data-centric stabilization recipe (Fig. 2) from three complementary mechanisms. *Controllable instruction diversification* generates semantically equivalent variants by constraining persona, syntactic pattern, and attribute slots, yielding systematic coverage expansion without free-form drift. *Drift filtering* removes residual supervision-corrupting rewrites via an LLM verifier

**indicates the corresponding author.

[1]Audio Samples are available at https://anonymous.4open.science/api/repo/instruct-demo-301C/file/index.html?v=2c0101a1

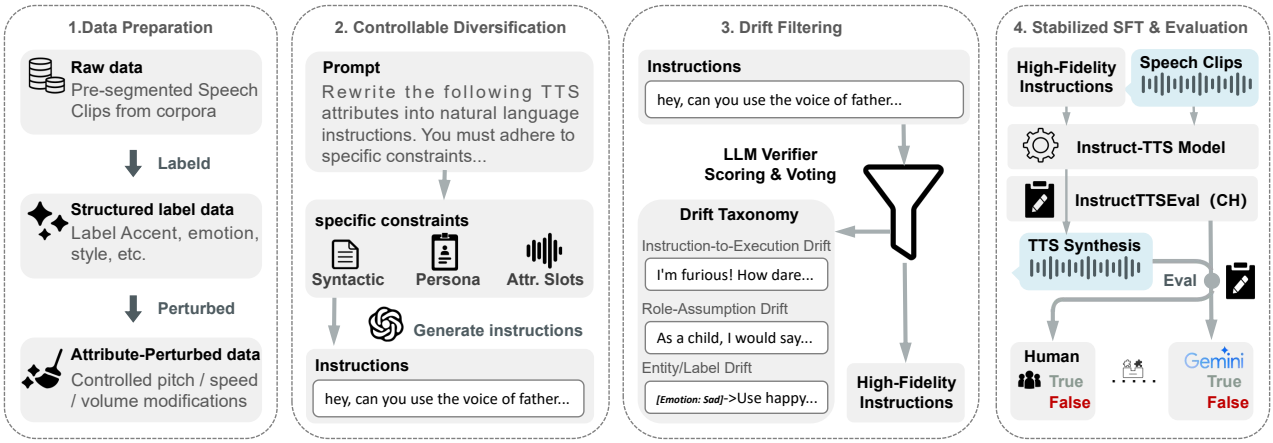

Figure 2: *Overview of our data-centric stabilization recipe: (1) data preparation with structured labeling and pitch/speed/volume perturbation; (2) controllable instruction diversification with persona, syntactic pattern, and attribute slot constraints; (3) drift filtering via LLM verifier scoring and voting; (4) stabilized fine-tuning and evaluation on InstructTTSEval.*

with scoring and voting [7, 8], guided by our drift taxonomy. *Attribute-aligned supervision* pairs pitch/speed/volume perturbations with natural-language attribute prompts [13, 14], grounding low-level prosody controllability that label-to-instruction rewrites alone cannot reliably cover. On the Chinese split of InstructTTSEval [15], the full recipe consistently outperforms both a no-SFT baseline and naive fine-tuning on unfiltered rewrites. Our contributions are:

- **Problem characterization.** We define *instruction supervision instability* and quantify it with a three-category drift taxonomy, revealing that over 40% of unconstrained LLM rewrites corrupt supervision.
- **Data-centric stabilization.** We combine controllable diversification, drift filtering, and attribute-aligned supervision into a unified recipe and show through ablation that the three mechanisms are complementary.
- **Empirical validation.** On InstructTTSEval (Chinese) [15], our recipe yields consistent gains in instruction-following accuracy and controllability, with drift filtering as the single most impactful component.

2. Related Work

Instruct-TTS and instruction data quality. PromptTTS [1], PromptTTS 2 [2], InstructTTS [3], and VoxInstruct [4] progressively extend natural-language control from speaking style to multi-dimensional instructions covering emotion, accent, and prosody, while ParlerTTS scales the paradigm with large annotated corpora [16]. These efforts focus on model architectures and training objectives, treating the upstream instruction data as given. Our work is orthogonal: we target the coverage and fidelity of the label-to-instruction supervision itself, systematically characterizing domain-specific semantic drift that standard NLP instruction pipelines [5, 6, 7] do not address.

Controllable speech synthesis. Conventional controllable TTS relies on reference audio embeddings or discrete style tags [17, 18], while low-level attributes such as pitch, speed, and volume are typically adjusted through signal processing or model-internal controls [19, 20]. Our attribute-aligned supervision bridges this gap by pairing parameterized perturbations with natural-language prompts, providing high-fidelity grounded signal for dimensions that free-form rewrites cover with limited reliability.

3. Method

3.1. Attribute-Aligned Supervision

We start from a speech corpus pre-segmented into short clips, each annotated with structured seed attributes such as accent, emotion, gender, and speaking style (Fig. 2, Stage 1). Since these high-level labels alone provide limited coverage of low-level prosody controls, we additionally apply parameterized perturbations along pitch, speaking rate, and volume to each clip [13, 14, 19]. Perturbation settings are mapped to concise natural-language attribute prompts via lightweight templates, yielding paired (perturbed speech, attribute prompt) examples that ground low-level controllability in high-fidelity acoustic manipulations rather than potentially drifted LLM outputs.

3.2. Controllable Instruction Diversification

To expand the limited coverage of template-based instructions, we generate instruction variants from seed attributes a under three explicit constraints (Fig. 2, Stage 2): *persona* specifies the speaker viewpoint to diversify pragmatic intent; *syntactic pattern* constrains the surface form to mitigate template over-reliance; and *attribute slots* require all target attributes as non-droppable slots, preventing entity/label drift through omission. These constraints are instantiated as hard requirements in an LLM rewrite prompt [8, 6], and for each seed we enumerate persona $\times$ pattern combinations, making coverage gains systematic and bounded. While constrained rewriting also reduces semantic drift substantially (40.4% $\rightarrow$ 15.4%; Table 2), the residual drift motivates the subsequent filtering stage.

3.3. Drift Filtering

While diversification expands coverage by generating more candidates, ensuring fidelity requires removing those that have drifted from the intended semantics.

Drift taxonomy. We define three drift categories to operational-

Table 1: *Main results on InstructTTSEval (Chinese). Acc. (%) denotes Gemini-judged instruction-following accuracy across APS, DSD, and RP tasks. Avg. is the arithmetic mean of the three. CER (%) is character error rate of synthesized speech. NMOS and CMOS are human-rated naturalness and controllability scores (1–5 scale, 95% CI). Upper block: external baseline; lower block: internal recipe variants on the same CosyVoice 2 backbone. Best in our systems in* ***bold****.*

		Instruction Following Acc. (%)					Human Eval (MOS)	
	Setting	**APS**	**DSD**	**RP**	**Avg.**	**CER↓**	**NMOS↑**	**CMOS↑**
Ext.	VoxInstruct	47.5	52.3	42.6	47.5	22.5	3.18±.08	3.12±.09
Ours	Base (no SFT)	20.2	45.5	37.7	34.5	35.0	3.30±.09	3.17±.09
	Naive SFT	45.2	62.2	45.5	51.0	19.2	3.46±.08	3.22±.09
	Full Recipe	**49.5**	**65.4**	**54.4**	**56.4**	**17.7**	**4.16**±.06	**4.16**±.06

Table 2: *Drift rates (%) of LLM-expanded instructions.* Unconstrained*: free-form LLM rewriting;* Constrained*: rewriting with persona, pattern, and slot controls (800 annotated samples).*

Drift Type	**Unconstrained**	**Constrained**
Instruction-to-Execution	18.3	7.1
Role-Assumption	12.7	4.5
Entity/Label	9.4	3.8
Total	40.4	15.4

Table 3: *Ablation on InstructTTSEval (Chinese), removing each component from the Full Recipe. Metrics follow Table 1.*

	Instruction Following Acc. (%)				
Setting	**APS**	**DSD**	**RP**	**Avg.**	**CER↓**
Full Recipe	49.5	65.4	54.4	56.4	17.7
-- Ctrl. Div.	45.8	60.1	49.6	51.8	18.4
-- Drift Filt.	43.1	57.3	46.2	48.9	19.8
-- Attr. Sup.	42.7	63.8	52.7	53.1	18.1

ize semantic fidelity (Fig. 1). *Instruction-to-execution drift*: the rewrite becomes an execution-like utterance that "says the line" rather than specifying how to speak; this is the most prevalent category and is particularly harmful for tasks where the boundary between instructing a style and performing it is subtle. *Role-assumption drift*: the rewrite adopts an in-character persona instead of issuing a control instruction, producing supervision-corrupting training pairs that teach the model persona imitation rather than controllability. *Entity/label drift*: the rewrite silently alters seed attributes such as emotion, accent, or style, directly corrupting the attribute-level fidelity that attribute-aligned supervision seeks to establish.

LLM verifier with scoring and voting. An LLM verifier, drawn from a different model family than the instruction generator to prevent self-evaluation bias [7, 21], receives the seed specification together with a candidate instruction and outputs a drift/no-drift decision, a drift category, and a confidence score. We discard candidates whose confidence falls below a threshold, and for borderline cases query the verifier multiple times with independent sampling, retaining an instruction only if it receives a majority "no-drift" vote [8]. This scoring-and-voting mechanism exposes a controllable coverage–fidelity trade-off, consistent with prior instruction data filtering pipelines [5, 6]. The output is a set of high-fidelity instructions used to form stabilized instruction–speech pairs for fine-tuning, directly addressing the supervision-corrupting drift that causes instruction supervision instability.

4. Experiments

4.1. Experimental Setup

Data and perturbation. We fine-tune CosyVoice 2.0-0.5B[22] on a Chinese speech collection of approximately 90 h (12k clips, $<$30 s each) covering eight accent and ten speaker-style categories with structured attribute annotations. For attribute-aligned supervision, each clip is perturbed along pitch ($\pm$1/2/3 semitones), speed ($\times$0.8/0.9/1.1/1.2/1.3), and volume ($\pm$3/6/9 dB), yielding 17 variants per clip.

Instruction generation and filtering. We compare GPT-4o[23], DeepSeek-R1[24], and QwQ-32B as instruction generators; DeepSeek-R1 produces the most diverse and attribute-faithful outputs, generating up to 24 candidates per seed (6 personas $\times$ 4 syntactic patterns). GPT-4o serves as drift verifier (threshold $\leq$5/10, self-consistency voting[8] for borderline cases), retaining $\approx$61% of candidates. Full prompt templates, persona/pattern specifications, filtering hyperparameters, and a quantitative generator comparison are in the supplementary material.

Training and evaluation. SFT uses AdamW (lr 2×10^{-5}, batch 16, 3 epochs, cosine schedule, 500-step warmup). We evaluate on the Chinese split of InstructTTSEval across three tasks, namely Attribute-controlled Pronunciation and Style (APS), Dialogue Scene Description (DSD), and Role-Playing (RP), reporting Gemini 3 Pro[25]-judged instruction-following accuracy where the judge listens to synthesized audio alongside the instruction text. Since InstructTTSEval uses unseen instructions distinct from training prompts, it directly tests generalization to diverse real-user phrasings. For human evaluation, 20 native Mandarin listeners rate 20 utterances per setting on naturalness (NMOS) and controllability (CMOS) on a 1–5 scale. VoxInstruct[4] is evaluated from its official checkpoint with the same instructions.

4.2. Drift Prevalence and Downstream Impact

Drift prevalence. We manually annotate 800 LLM-expanded instructions, half from unconstrained rewriting and half from constrained rewriting, to estimate drift rates under our taxonomy. As shown in Table 2, unconstrained rewriting yields 40.4% total drift (instruction-to-execution 18.3%, role-assumption 12.7%, entity/label 9.4%). Constrained generation

reduces total drift to 15.4% (≈half per category). However, the residual 15.4% remains non-trivial and motivates the subsequent filtering stage.

Downstream impact. The link between drift rate and downstream performance is supported by comparing across Tables 1 and 3. Naive SFT trains on unfiltered rewrites with an estimated 40% drift and achieves 51.0% average accuracy. Removing drift filtering from the Full Recipe reintroduces residual drift into training and drops the average to 48.9%, while the Full Recipe with filtering applied reduces drift to roughly 5% (validated by post-filtering human audit in supplementary material) and reaches 56.4%. This monotonic pattern across three operating points with decreasing drift rates confirms that supervision corruption directly degrades instruction-following performance and that filtering is the primary driver of improvement.

4.3. Main Results and Ablation

Main results. Table 1 compares instruction-following accuracy, intelligibility, and human evaluation scores on InstructTTSEval. VoxInstruct achieves 47.5% average accuracy but obtains the lowest human ratings (NMOS 3.18, CMOS 3.12). Within our CosyVoice 2 backbone, Naive SFT improves over the no-SFT baseline in both instruction following (34.5% $\rightarrow$ 51.0%, CER 35.0% $\rightarrow$ 19.2%) and human ratings (NMOS 3.30 $\rightarrow$ 3.46). The Full Recipe achieves substantially higher human scores (NMOS 4.16, CMOS 4.16), outperforming all other systems by over 0.7 on both scales, confirming that supervision quality is the primary bottleneck for naturalness and controllability. Human CMOS rankings are largely consistent with Gemini-judged accuracy. The three pipeline stages rely on distinct model families (DeepSeek-R1, GPT-4o, Gemini), mitigating systematic bias from any single LLM. Judge stability analysis is provided in the supplementary material.

Ablation. Table 3 decomposes contributions by removing each component from the Full Recipe. Drift filtering has the largest impact: removing it drops the average from 56.4% to 48.9% despite *increasing* the training set size, confirming that the gain stems from removing drifted samples rather than a data volume effect (size-matched control in supplementary material). Removing controllable diversification yields an average of 51.8%, indicating that systematic coverage expansion contributes beyond what filtering alone provides. Removing attribute-aligned supervision has a moderate effect on the average at 53.1%, but disproportionately impacts APS, which drops from 49.5% to 42.7%, consistent with its targeted role in low-level attribute control. The three components are complementary: filtering ensures fidelity, diversification expands coverage, and attribute alignment strengthens grounded prosody control.

4.4. Effect of Attribute-Aligned Supervision

Table 4 decomposes APS accuracy by attribute. The Full Recipe outperforms Base and Naive SFT across four dimensions. Pitch and Speed show the largest absolute gains over Base at roughly +32 each, reflecting the benefit of attribute-aligned supervision for these well-defined acoustic dimensions. Volume also improves by +31.8, a notable result given that Base accuracy on Volume is only 8.7%, the lowest among all attributes; this confirms loudness is difficult to learn from label-to-instruction supervision alone, and that attribute-aligned supervision provides critical grounded signal through parameterized perturbations. Emotion shows a smaller gain of +21.4, consistent with its reliance on high-level style labels rather than low-level perturbations. Interestingly, Naive SFT achieves higher Emotion accuracy than the Full Recipe on this dimension (50.0 vs. 42.6), suggesting unfiltered rewrites may over-represent emotional descriptors at the expense of other attributes, a pattern corrected by our balanced recipe.

Table 4: *Per-attribute accuracy (%) on the APS task, isolating control over individual acoustic dimensions.* Δ *denotes absolute gain of Full Recipe over Base.*

Setting	Pitch	Speed	Volume	Emotion
Base (no SFT)	24.6	26.3	8.7	21.2
Naive SFT	50.1	52.4	28.3	50.0
Full Recipe	**56.7**	**58.2**	**40.5**	**42.6**
Δ	+32.1	+31.9	+31.8	+21.4

Table 5: *Effect of drift filtering strategy on DSD accuracy and data retention rate.* Scoring & Voting *is the combined strategy used in our full recipe.*

Filtering Strategy	DSD Acc. (%)	Retention (%)
No Filtering	57.3	100.0
Threshold Only	63.8	72.3
Voting Only	63.5	68.1
Scoring & Voting	**65.4**	61.5

4.5. Drift Filtering Strategy Comparison

Table 5 compares drift filtering strategies on DSD accuracy and data retention. Without filtering, DSD accuracy is 57.3%, consistent with the ablation in Table 3. Applying confidence thresholding alone improves accuracy to 63.8% while retaining 72.3% of candidates, and self-consistency voting alone achieves a comparable 63.5% at 68.1% retention. Combining the two strategies yields the best accuracy at 65.4% but retains only 61.5% of data, reflecting the inherent coverage–fidelity tradeoff: stricter filtering improves semantic fidelity at the cost of reduced diversity. In our setting, the combined strategy provides the strongest balance, and the threshold and voting strength can be tuned based on data scale and desired supervision strictness.

5. Conclusion

In this work, we analyze instruction supervision instability in label-to-instruction pipelines for Instruct-TTS, where semantic drift in LLM rewrites corrupts supervision and weaken generalization. We propose a data-centric stabilization recipe that jointly improves coverage and fidelity via controllable instruction diversification, drift filtering with an LLM verifier, and attribute-aligned supervision from pitch/speed/volume perturbations paired with prompts. Experiments on the Chinese split of InstructTTSEval show that constrained rewriting reduces drift substantially (40.4%$\rightarrow$15.4%), and the full recipe achieves the best instruction-following accuracy with particularly strong gains on low-level controllability. Our key contribution is a reproducible framework for stabilizing instruction supervision: we formalize supervision corruption with a drift taxonomy, and show that jointly improving coverage and fidelity is necessary for reliable generalization. We believe this analytical perspective and the drift taxonomy are transferable to other instruction-driven generative tasks beyond speech synthesis.

6. Generative AI Use Disclosure

During the preparation of this manuscript, generative AI tools were used for language refinement and minor editing. In addition, large language models were employed as part of the experimental methodology, including instruction generation, drift verification, and automatic evaluation, as described in the main text. All experimental design, analysis, and conclusions were developed and verified by the authors, who take full responsibility for the content of this paper.

7. References

[1] Z. Guo, Y. Leng, Y. Wu, S. Zhao, and X. Tan, "Prompttts: Controllable text-to-speech with text descriptions," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[2] Y. Leng, Z. Guo, K. Shen, X. Tan, Z. Ju, Y. Liu, Y. Liu, D. Yang, L. Zhang, K. Song *et al.*, "Prompttts 2: Describing and generating voices with text prompt," *arXiv preprint arXiv:2309.02285*, 2023.

[3] D. Yang, S. Liu, R. Huang, C. Weng, and H. Meng, "Instructtts: Modelling expressive tts in discrete latent space with natural language style prompt," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 2913–2925, 2024.

[4] Y. Zhou, X. Qin, Z. Jin, S. Zhou, S. Lei, S. Zhou, Z. Wu, and J. Jia, "Voxinstruct: Expressive human instruction-to-speech generation with unified multilingual codec language modelling," in *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 554–563.

[5] Y. Wang, Y. Kordi, S. Mishra, A. Liu, N. A. Smith, D. Khashabi, and H. Hajishirzi, "Self-instruct: Aligning language models with self-generated instructions," in *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, 2023, pp. 13 484–13 508.

[6] C. Xu, Q. Sun, K. Zheng, X. Geng, P. Zhao, J. Feng, C. Tao, and D. Jiang, "Wizardlm: Empowering large language models to follow complex instructions," *arXiv preprint arXiv:2304.12244*, 2023.

[7] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. Xing *et al.*, "Judging llm-as-a-judge with mt-bench and chatbot arena," *Advances in neural information processing systems*, vol. 36, pp. 46 595–46 623, 2023.

[8] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, and D. Zhou, "Self-consistency improves chain of thought reasoning in language models," *arXiv preprint arXiv:2203.11171*, 2022.

[9] J. Fu, G. Zhao, Y. Deng, Y. Mi, and X. Qian, "Learning to paraphrase for alignment with llm preference," in *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024, pp. 2394–2407.

[10] J. Sun, C. Shaib, and B. C. Wallace, "Evaluating the zero-shot robustness of instruction-tuned language models," *arXiv preprint arXiv:2306.11270*, 2023.

[11] A. Alajrami, X. Tan, and N. Aletras, "Fine-tuning on noisy instructions: Effects on generalization and performance," in *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, 2025, pp. 728–742.

[12] D. Zhang, Q. Dai, and H. Peng, "The best instruction-tuning data are those that fit," *arXiv preprint arXiv:2502.04194*, 2025.

[13] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition." in *Interspeech*, vol. 2015, 2015, p. 3586.

[14] N. Jaitly and G. E. Hinton, "Vocal tract length perturbation (vtlp) improves speech recognition," in *Proc. ICML workshop on deep learning for audio, speech and language*, vol. 117, 2013, p. 21.

[15] K. Huang, Q. Tu, L. Fan, C. Yang, D. Zhang, S. Li, Z. Fei, Q. Cheng, and X. Qiu, "Instructttseval: Benchmarking complex natural-language instruction following in text-to-speech systems," *arXiv preprint arXiv:2506.16381*, 2025.

[16] D. Lyth and S. King, "Natural language guidance of high-fidelity text-to-speech with synthetic annotations," *arXiv preprint arXiv:2402.01912*, 2024.

[17] Y. Wang, D. Stanton, Y. Zhang, R.-S. Ryan, E. Battenberg, J. Shor, Y. Xiao, Y. Jia, F. Ren, and R. A. Saurous, "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," in *International conference on machine learning*. PMLR, 2018, pp. 5180–5189.

[18] R. Valle, J. Li, R. Prenger, and B. Catanzaro, "Mellotron: Multispeaker expressive voice synthesis by conditioning on rhythm, pitch and global style tokens," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6189–6193.

[19] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech 2: Fast and high-quality end-to-end text to speech," *arXiv preprint arXiv:2006.04558*, 2020.

[20] B. Bai, Y. Geng, F. Wang, C. Wang, P. Guo, Y. Gao, and Y. Li, "Hq-svc: Towards high-quality zero-shot singing voice conversion in low-resource scenarios," *arXiv preprint arXiv:2511.08496*, 2025.

[21] J. Gu, X. Jiang, Z. Shi, H. Tan, X. Zhai, C. Xu, W. Li, Y. Shen, S. Ma, H. Liu *et al.*, "A survey on llm-as-a-judge," *The Innovation*, 2024.

[22] Z. Du, Y. Wang, Q. Chen, X. Shi, X. Lv, T. Zhao, Z. Gao, Y. Yang, C. Gao, H. Wang *et al.*, "Cosyvoice 2: Scalable streaming speech synthesis with large language models," *arXiv preprint arXiv:2412.10117*, 2024.

[23] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford *et al.*, "Gpt-4o system card," *arXiv preprint arXiv:2410.21276*, 2024.

[24] D. Guo, D. Yang, H. Zhang, J. Song, P. Wang, Q. Zhu, R. Xu, R. Zhang, S. Ma, X. Bi *et al.*, "Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning," *arXiv preprint arXiv:2501.12948*, 2025.

[25] G. Comanici, E. Bieber, M. Schaekermann, I. Pasupat, N. Sachdeva, I. Dhillon, M. Blistein, O. Ram, D. Zhang, E. Rosen *et al.*, "Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities," *arXiv preprint arXiv:2507.06261*, 2025.