Rethinking Progress and Evaluation in Automatic Speech Recognition

Anonymous ACL submission

Abstract

Automatic speech recognition (ASR) is increasingly deployed in safety-critical settings where transcription errors can carry operational consequences. However, standard metrics such as Word Error Rate, BERTScore, and semantic distance measure surface accuracy rather than operational correctness. Using ASR in air traffic control as a case study, we show that fine-tuning improves these metrics but increases the concentration of operationally critical errors among outputs deemed correct by those same metrics. We call this the *Finetuning Paradox*: benchmark performance improves while operational safety improvement is substantially overstated, particularly under distribution shift, when correctness depends on sparse, high-impact tokens such as callsigns, runways, headings, and altitudes. To address this gap, we propose Contextual Error Rate (CxER), a multi-judge LLM evaluation method that evaluates whether ASR outputs preserve operational meaning. We show that CxER is robust across prompts and model families and aligns strongly with aviation expert annotations (F1 = 0.866, κ = 0.587) approaching human agreement levels (Cohen's κ = 0.65). These findings show that benchmark gains may not reflect real-world reliability in domains where operational meaning is concentrated in rare but safety-critical tokens. (code available at: `https://anonymous.4open.science/r/cxer-release-79B1`)

1 Introduction and Related Work

Automatic speech recognition (ASR) systems are commonly evaluated by how closely their outputs match a reference transcript. But in safety-critical environments, the central question is not lexical similarity, it is whether the transcription preserves operational meaning. In aviation, emergency dispatch, and clinical documentation, a single incorrect token may alter the meaning of an instruction despite minimal lexical deviation from the reference. For example, changing "*taxi to runway two-seven* ***left***" to "*taxi to runway two-seven* ***right***" yields a near-perfect Word Error Rate (WER) (Watanabe et al., 2018) and BERTScore (Zhang et al., 2019), yet fundamentally corrupts an aviation clearance in a way that could cause a runway incursion. The core problem is not model capability: the metrics used to evaluate these systems were never designed to measure operational correctness.

Existing metrics implicitly assume that transcription quality is proportional to semantic similarity. This assumption is reasonable for conversational transcription, but breaks down in operational settings where meaning depends disproportionately on a small number of high-impact tokens. WER (Watanabe et al., 2018) treats all substitutions, insertions, and deletions equally. More structured alternatives, including Character Error Rate (Xu et al., 2021), Part-of-Speech Error Rate (Bañeras-Roux et al., 2026), and Named Entity Error Rate (Garofolo et al., 1999), introduce linguistic structure but remain agnostic to operational consequence. Semantic metrics, such as BERTScore (Zhang et al., 2019) and SemDist (Kim et al., 2021), extend evaluation beyond surface overlap but are context-agnostic: "*Delta 123*" and "*Delta 121*" receive highly similar representations despite the fact that confusing these identifiers constitutes a mission-critical failure in ATC communication.

Standard next-token training objectives do not explicitly optimize for operational correctness. During fine-tuning, optimization may therefore prioritize high-frequency lexical patterns over sparse safety-critical entities such as callsigns and runway identifiers, leading benchmark improvements to diverge from operational reliability, a structural metric-safety divergence we term the *Finetuning Paradox*.

The LLM-as-a-Judge paradigm (Zheng et al., 2023) offers a qualitatively different approach: instruction-tuned LLMs can reason over domain-

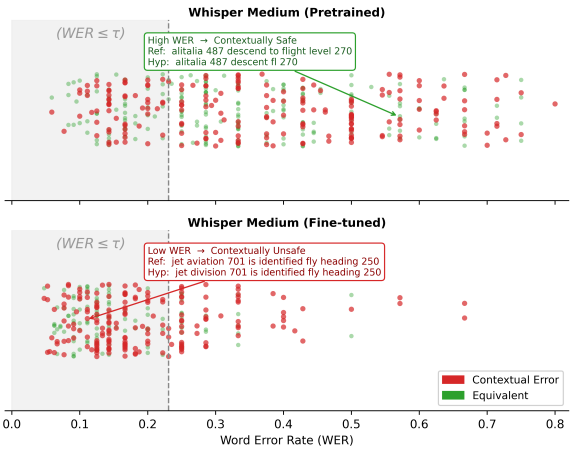

Figure 1: WER distribution for 200 representative ATCOSIM test samples before and after Whisper fine-tuning. Contextual Errors (red) become more concentrated after fine-tuning in τ derived from bottom 25% of the pretrained WER distribution.

specific criteria under structured prompts, combining scalability with context-sensitive judgment (Sun et al., 2026; Li et al., 2024; Wang et al., 2024; Zhu et al., 2025). Aditionally, Verga et al. (2024) show that panels of diverse LLM evaluators consistently outperform individual judges, motivating heterogeneous ensembles where annotation is costly and domain expertise is required.

Motivated by the Finetuning Paradox, we propose the **Contextual Error Rate (CxER)**, an LLM ensemble framework that assesses whether ASR outputs preserve operational meaning. Our contributions are threefold: (1) demonstrating empirically that current ASR evaluation metrics fail to capture operational risk and systematically overestimate correctness during fine-tuning; (2) proposing CxER, validated against aviation subject-matter experts with ensemble approaching moderate-to-substantial agreement (κ = 0.587 to 0.618); and (3) establishing robustness across multiple prompt variants, ASR systems, and three open-weight LLM families. More broadly, our findings suggest that any evaluation paradigm aggregating errors uniformly including entity-weighted metrics that lack a model of operational context will systematically underestimate risk in domains where the criticality of a token depends on its communicative function, not its surface form.

2 Methodology

2.1 ASR Systems and Corpora

We consider two ASR architectures. Whisper Medium (Radford et al., 2023) is a 769M-parameter encoder-decoder Transformer trained on 680k hours of weakly supervised multilingual audio. NVIDIA Parakeet-TDT-0.6B (Sekoyan et al., 2025) is a FastConformer-based transducer. Both models are trained using supervised fine-tuning jointly on the official training splits of two public ATC corpora: ATCOSIM (Hofbauer et al., 2008), consisting of approximately 10 hours of simulated ATC speech from non-native speakers, and UWB-ATCC (Šmídl et al., 2019), containing approximately 20 hours of real controller-pilot communications. Evaluation is performed separately on the official test splits of both corpora. To assess robustness under domain shift, we also evaluate on ATCO2 (Zuluaga-Gomez et al., 2022), a held-out corpus collected under substantially different recording conditions and speaker distributions, used exclusively for out-of-distribution (OOD) evaluation and excluded from fine-tuning.

We evaluate using WER, BERTDistance ($1 -$ BERTScore), and SemDist. To assess whether structured entity-aware metrics can capture operational failures without contextual reasoning, we also include an entity-weighted WER baseline that assigns higher penalties to safety-critical ATC entities, including callsigns, runways, altitudes, frequencies, and headings, using a deterministic ATC entity tagger. Full implementation details and code are provided in the supplementary material.

2.2 Contextual Error Rate (CxER)

We propose CxER, a multi-judge LLM evaluation method that assesses whether ASR outputs preserve operational meaning rather than surface similarity. Given a reference transcript and an ASR prediction, each sample is assigned one of two labels: *Equivalent*, indicating that operational intent is preserved, or *Contextual_Error*, indicating that operationally critical information has been altered or lost. CxER is defined as the proportion of outputs classified as *Contextual_Error*, i.e., CxER $= |\textit{Contextual_Error}|/N$, where N is the total number of evaluated samples. For *Contextual_Error* samples, CxER additionally assigns a fine-grained error category reflecting ATC safety priorities: callsign, runway, altitude, heading, frequency, clearance, navigation, weather, and other.

CxER evaluations are produced by three open-weight LLMs running independently: Llama-3.3-70B-Instruct (Meta, 2024), Qwen2.5-72B-Instruct (Yang and et al., 2024), and DeepSeek-R1-Distill-Llama-70B (DeepSeek-AI, 2025), selected to differ

Metric	Model	ATCO2	ATCC	ATCOSim
Δ_{WER}	Whisper	+24.1	+34.9	+53.7
	Parakeet	+14.2	+22.8	+31.9
$\Delta_{WeightedWER}$	Whisper	+27.5	+38.3	+50.6
	Parakeet	+15.2	+20.4	+33.9
$\Delta_{BERTDist}$	Whisper	+19.8	+40.5	+51.7
	Parakeet	+8.0	+17.3	+31.2
$\Delta_{SemDist}$	Whisper	+19.3	+38.5	+46.0
	Parakeet	+13.3	+23.2	+35.6

Table 1: Finetuning Paradox results across all metrics, architectures, and corpora

Rater / Judge	Avg. κ	Avg. Wtd. κ	F1
Human Agreement (Ceiling)			
Human_1 – Human_2	0.650	0.665	–
Individual LLM vs Human			
Llama-3.3-70B	0.529	0.591	0.825
Qwen-2.5-72B	0.488	0.573	0.766
DeepSeek-R1-70B	0.592	0.636	0.845
LLM Consensus Ensemble			
Consensus – Human	**0.593**	**0.641**	**0.866**

Table 2: CxER validation results (avg. over 2 annotators).

substantially in training data, alignment procedures, and reasoning behavior. Final labels are obtained through majority voting over individual predictions. We evaluate four prompt variants to assess robustness; full prompts are provided in the Appendix.

3 Experiments and Results

3.1 The Finetuning Paradox

Fine-tuning dramatically improves all standard metrics relative to pretrained baselines (Figure 2). Yet CxER improvements are consistently smaller, and with the divergence widening substantially under distribution shift. For example, on the OOD ATCO2 corpus, Parakeet's WER improves by 28.4% while its CxER improves by only 4.0%. The same pattern holds for Whisper and across all three corpora, where standard metrics overestimate operational improvement relative to CxER. This is Goodhart's Law in ASR evaluation: when a metric becomes the target, it ceases to measure what matters.

To assess whether standard benchmark metrics reliably identify operationally safe predictions, we analyze the subset of outputs assigned the strongest metric scores. We first compute a threshold τ corresponding to the best-performing quartile (lowest 25%) of the pretrained model's metric distribution for each metric and dataset. We then apply this same threshold to the fine-tuned predictions. In Figure 1, this region corresponds to the shaded gray area for the WER case. Within this metric-favorable region, we measure the fraction of all samples classified as *Contextual_Error* by CxER that fall below the threshold τ. This analysis is repeated across all four baseline metrics (WER, Weighted WER, BERTDist, and SemDist), both ASR architectures, and all three corpora.

Table 1 reports the change Δ in critical error prevalence within these high-confidence metric regions after fine-tuning. For example, Δ_{WER} denotes the difference between the fraction of contextual errors before and after fine-tuning within the WER-defined region; concretely, for Whisper on ATCOSim, this fraction rose from 28.5% to 82.2%, a +53.7 pp increase. Across all datasets and architectures, this proportion increases substantially after finetuning. On UWB-ATCC, Whisper increases by +34.9 pp, and on ATCO2 by +24.1 pp. The same pattern appears consistently across all metrics, architectures, and datasets (Appendix). These results show that operationally critical failures become increasingly concentrated in regions of metric space that standard benchmarks interpret as high quality.

3.2 CxER Validation

We validate CxER through inter-rater agreement studies with two aviation-domain doctoral researchers working on FAA-related ATC projects, who independently annotate 500 ASR predictions spanning all datasets and models. Agreement between human annotators and LLM evaluators is measured using Cohen's κ (Cohen, 1960), while multi-rater consistency among LLM judges is measured using Fleiss' κ (Fleiss, 1971).

To ensure sufficient coverage of operationally ambiguous cases, we intentionally oversample disagreement cases among LLM evaluators. The evaluation set contains 150 samples with unanimous *Equivalent* judgments, 150 with unanimous *Contextual_Error* judgments, and 200 with 2-vs-1 disagreement among judges. Because this sampling strategy differs from the natural corpus distribution, we additionally report population-reweighted κ scores using inverse probability weights derived from the full evaluation pool (9,988 critical, 4,729 equivalent, and 5,822 disagreement cases).

Samples with WER = 0 are excluded, as exact lexical matches provide no signal for contextual

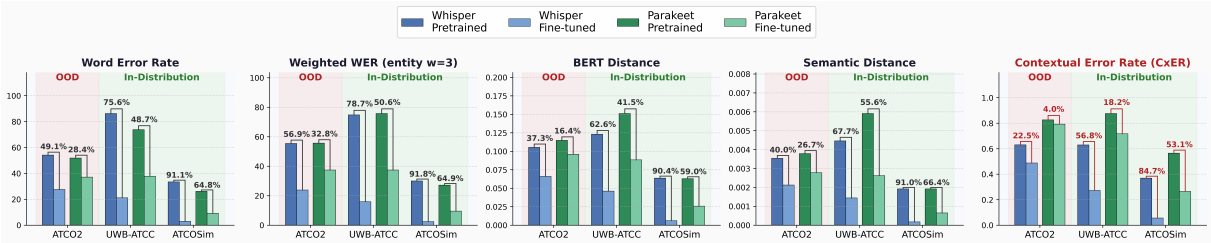

Figure 2: Improvement gap between conventional metrics and CxER across ASR models (Brackets show % reduction in metric), smaller reduction in CxER indicates that surface metrics overestimate real improvement.

evaluation. Approximately 2–2.5% of samples failing JSON parsing are excluded from all analyses.

We report CxER validation results in Table 2. Human-human agreement reaches $\kappa = 0.650$, establishing the practical agreement ceiling for this task. Individual LLM judges achieve moderate agreement with human annotators ($\kappa = 0.529$–0.592), with population-reweighted agreement consistently higher ($\kappa = 0.591$–0.636). This increase reflects the fact that disagreement cases were intentionally oversampled and are substantially more difficult than the natural corpus distribution.

The three LLM judges exhibit complementary error profiles. Qwen achieves high precision (0.920) but relatively low recall (0.656), tending to underdetect contextual errors, while DeepSeek achieves the highest recall (0.872). Llama provides the most balanced individual performance (F1 = 0.825). Aggregating judgments through majority voting improves overall robustness, yielding ensemble recall of 0.851 and agreement approaching the human-human ceiling ($\kappa = 0.593$, weighted $\kappa = 0.641$). Full per-judge results are provided in the Appendix.

3.3 Prompt Robustness

To assess robustness to prompt formulation, we construct a stratified benchmark of 300 samples drawn from the 500 SME-annotated pool: 100 critical, 100 Equivalent, and 100 disagreement cases, sampled with a fixed random seed for reproducibility. Each LLM judge is evaluated across four prompt variants differing in reasoning structure, example usage, and decision strictness. We measure agreement with human annotations using accuracy and F1, and quantify prompt stability using inter-prompt Fleiss' κ.

Results are summarized in Table 3. Qwen exhibits the highest prompt robustness (Fleiss' $\kappa = 0.836$), maintaining stable performance across all variants (Acc = 0.869–0.902). Llama shows moderate robustness ($\kappa = 0.764$), with the largest degradation under the `no_examples` variant (Acc = 0.781, F1 = 0.724), while DeepSeek achieves intermediate stability ($\kappa = 0.695$). Across all judges and prompt formulations, accuracy remains above 0.78 and best-prompt performance exceeds 0.89, indicating that CxER captures a stable operational signal rather than prompt-specific behavior. Full per-prompt results are provided in the Appendix.

Judge	Best Prompt	Acc	F1	Fleiss' κ
Qwen	cot	0.902	0.916	0.836
Llama	cot	0.892	0.908	0.764
DeepSeek	compressed	0.879	0.893	0.695

Table 3: Best-prompt performance and cross-prompt robustness. Full results across all four prompt variants are in the Appendix.

Catastrophic failures, where a judge disagrees with human annotations across all prompt variants, remain below 8.1% and are concentrated in operationally ambiguous samples where human annotators themselves exhibit lower agreement.

4 Conclusion

Standard ASR evaluation metrics share a structural blind spot: they measure surface accuracy, not operational correctness. Fine-tuning improves benchmark scores while concentrating critical errors precisely where metrics signal better transcription, a Goodhart's Law dynamic that worsens under domain shift. CxER exposes this divergence through independent LLM judges aggregated via majority vote, achieving F1 = 0.866 against human expert annotations with robustness across prompt variations and model families. The implications extend beyond aviation: in any domain where operational meaning concentrates in low-frequency tokens, current metrics create a false sense of progress. These findings suggest that metric-safety divergence may emerge naturally under token-level optimization objectives when operational correctness depends disproportionately on sparse, high-impact entities. Addressing this issue may require training and evaluation paradigms that incorporate contextual correctness as an explicit objective.

5 Limitations

CxER relies on high-capacity open-weight LLMs (70B+ parameters) as evaluators, which introduces non-trivial computational cost. We use open-weight models to ensure reproducibility and avoid dependence on proprietary APIs whose behavior may change over time. Preliminary experiments with smaller models (8B–13B parameters) yielded substantially lower agreement with human experts, consistent with broader observations that LLM-as-a-Judge reliability scales with model capacity. In practice, however, inference cost can be amortized because CxER is intended for offline evaluation and benchmarking rather than real-time deployment.

Human validation is currently limited to the aviation domain due to the need for specialized operational expertise. The Finetuning Paradox results are computed using CxER labels over the full test sets across all three corpora. Due to the cost of exhaustive SME annotation, human labels are instead collected on a stratified subset of samples to validate CxER reliability rather than full-corpus labeling. While we observe consistent effects across two ASR architectures, three corpora, and multiple standard metrics, we cannot fully exclude residual evaluator noise or sampling effects as a contributing factor. Finally, approximately 2–2.5% of samples were excluded due to JSON parsing failures. Parsing errors were concentrated in the `compressed` prompt variant for Llama-3.3, where 41.4% of outputs failed to conform to the required schema.

References

Thibault Bañeras-Roux, Mickael Rouvier, Jane Wottawa, and Richard Dufour. 2026. Qualitative evaluation of language model rescoring in automatic speech recognition. *arXiv preprint arXiv:2604.27533*.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378–382.

John S Garofolo, Ellen M Voorhees, Cedric GP Auzanne, Vincent M Stanford, and Bruce A Lund. 1999. 1998 trec-7 spoken document retrieval track overview and results. *NIST SPECIAL PUBLICATION SP*, pages 79–90.

Konrad Hofbauer, Stefan Petrik, and Horst Hering. 2008. The atcosim corpus of non-prompted clean air traffic control speech. In *LREC*, volume 3, page 8.

Suyoun Kim, Abhinav Arora, Duc Le, Ching-Feng Yeh, Christian Fuegen, Ozlem Kalinli, and Michael L Seltzer. 2021. Semantic distance: A new metric for asr performance analysis towards spoken language understanding. *arXiv preprint arXiv:2104.02138*.

Junlong Li, Shichao Sun, Weizhe Yuan, Run-Ze Fan, Pengfei Liu, and 1 others. 2024. Generative judge for evaluating alignment. In *International Conference on Learning Representations*, volume 2024, pages 27547–27574.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Meta. 2024. Llama-3.3-70b-instruct. Large language model.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.

Monica Sekoyan, Nithin Rao Koluguri, Nune Tadevosyan, Piotr Zelasko, Travis Bartley, Nikolay Karpov, Jagadeesh Balam, and Boris Ginsburg. 2025. Canary-1b-v2 & parakeet-tdt-0.6 b-v3: Efficient and high-performance models for multilingual asr and ast. *arXiv preprint arXiv:2509.14128*.

Luboš Šmídl, Jan Švec, Daniel Tihelka, Jindřich Matoušek, Jan Romportl, and Pavel Ircing. 2019. Air traffic control communication (atcc) speech corpora and their use for asr and tts development. *Language Resources and Evaluation*, 53(3):449–464.

Bian Sun, Zhenjian Wang, Orvill de la Torre, and Zirui Wang. 2026. When metrics disagree: Automatic similarity vs. llm-as-a-judge for clinical dialogue evaluation. *arXiv preprint arXiv:2603.00314*.

Pat Verga, Sebastian Hofstatter, Sophia Althammer, Yixuan Su, Aleksandra Piktus, Arkady Arkhangorodsky, Minjie Xu, Naomi White, and Patrick Lewis. 2024. Replacing judges with juries: Evaluating llm generations with a panel of diverse models. *arXiv preprint arXiv:2404.18796*.

Yidong Wang, Zhuohao Yu, Wenjin Yao, Zhengran Zeng, Linyi Yang, Cunxiang Wang, Hao Chen, Chaoya Jiang, Rui Xie, Jindong Wang, and 1 others. 2024. Pandalm: An automatic evaluation benchmark for llm instruction tuning optimization. In *International Conference on Learning Representations*, volume 2024, pages 43573–43593.

Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, and 1 others. 2018. Espnet: End-to-end speech processing toolkit. *arXiv preprint arXiv:1804.00015*.

Menglong Xu, Shengqiang Li, and Xiao-Lei Zhang. 2021. Transformer-based end-to-end speech recognition with local dense synthesizer attention. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5899–5903. IEEE.

An Yang and et al. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. 2025. Judgelm: Fine-tuned large language models are scalable judges. In *International Conference on Learning Representations*, volume 2025, pages 51257–51296.

Juan Zuluaga-Gomez, Karel Veselỳ, Igor Szöke, Petr Motlicek, and 1 others. 2022. Atco2 corpus: A large-scale dataset for research on automatic speech recognition and natural language understanding of air traffic control communications. *arXiv preprint arXiv:2211.04054*.

A Appendix

A.1 Baseline Evaluation Metrics

We use four standard metrics plus a rule-based entity-weighted variant as baselines. **WER** is the normalised Levenshtein edit distance at the word level, aggregated corpus-wide (Watanabe et al., 2018). **BERTDist** (Zhang et al., 2019) is $1 -$ BERTScore F1 computed with RoBERTa-large (Liu et al., 2019); lower is better. **SemDist** (Kim et al., 2021) is the cosine distance between mean-pooled RoBERTa-large sentence embeddings; lower is better.

Weighted WER (WWER) is a deterministic entity-aware baseline motivated by a cheaper alternative to CxER. An ATC-specific rule-based tagger identifies entity tokens: callsigns, runways, flight levels, altitudes, frequencies, and headings using regular expressions and an ICAO airline vocabulary, and assigns substitution/deletion cost $w{=}3$ relative to non-entity tokens ($w{=}1$) during edit-distance alignment. We conduct a sensitivity analysis over $w \in \{1, 3, 5, 7, 10\}$ and find that the concentration of critical errors in low-metric regions remains largely stable across all weight values. This confirms that surface-level entity weighting cannot resolve the Finetuning Paradox and motivates the need for a meaning-aware metric such as CxER 4.

A.2 CxER Prompt Variants

CxER uses four prompt variants to assess robustness of LLM judge behaviour across different instruction formulations. All variants share the same JSON output schema and binary classification task (`Equivalent` vs. `Critical_Errors`). The **CoT** variant is canonical and is used for all main paper results; the remaining three are used exclusively for the robustness analysis reported in Section 3.3 Appendix A.5.

Each variant isolates a specific dimension of prompt design: CoT provides maximum structure with ICAO normalisation, concrete examples, entity priority ordering, and chain-of-thought reasoning; Compressed reduces verbosity while preserving structure; No-Examples removes few-shot exemplars while keeping all rules; Lenient removes the structured taxonomy entirely and relies on pure semantic judgment. This design forms a controlled ablation spectrum.

Weight (entity emphasis)	Whisper ATCO2	Whisper UWB-ATCC	Whisper ATCOSim	Parakeet ATCO2	Parakeet UWB-ATCC	Parakeet ATCOSim
1	+23.6	+34.9	+52.3	+14.2	+22.8	+31.9
3	+27.5	+38.3	+50.6	+15.2	+20.4	+33.9
5	+31.1	+38.4	+45.0	+16.5	+21.7	+33.5
7	+31.2	+38.5	+44.1	+17.1	+22.4	+32.9
10	+30.8	+37.9	+42.4	+17.6	+22.3	+32.3

Table 4: Sensitivity analysis: Weighted WER improvement with increasing entity weight. Whisper benefits most on ATCOSim at low weights (weight=1: +52.3%), while Parakeet shows smaller but more stable gains. Peak performance often at weight=5–7 for Whisper, weight=10 for Parakeet.

A.2.1 CoT (Canonical)

Provides: (1) a full ICAO phonetics normalisation table; (2) concrete examples of critical errors across all operational entity types; (3) explicit equivalence rules covering format variants, spoken abbreviations, and fillers; (4) an entity priority ordering; and (5) structured chain-of-thought reasoning inside `<reasoning>` tags before producing JSON output.

```
You are an expert ATC evaluator. Compare reference
      transcript against
transcription. Identify ONLY operationally significant
      errors (meaning
changed). Ignore equivalent variants and filler words.

REFERENCE: {reference}
TRANSCRIPTION: {transcription}

--- ICAO PHONETICS ---
Numbers: wun/one=1, too/two=2, tree/three=3, fower/four=4,
      fife/five=5,
six=6, seven=7, ait/eight=8, niner/nine=9, zero=0
Letters: alfa/alpha=A, bravo=B, charlie=C, delta=D, echo=E,
foxtrot/fox-trot=F, golf=G, hotel=H, india=I,
      juliett/juliet=J,
kilo=K, lima=L, mike=M, november=N, oscar=O, papa=P,
      quebec=Q,
romeo=R, sierra=S, tango=T, uniform=U, victor=V, whiskey=W,
x-ray/xray=X, yankee=Y, zulu=Z

--- CRITICAL ERRORS (meaning changed) ---
Any of the following differences are CRITICAL:
- Callsign: "united two three four" vs "united two three
      five"
- Direction: "turn left" vs "turn right"
- Altitude: "eight thousand" vs "one eight thousand"
- Heading: "heading two seven zero" vs "heading two seven"
- Runway: "runway two seven left" vs "runway two niner
      right"
- Frequency: "one two one decimal five" vs "one two two
      decimal five"
- Navigation: "direct DINKU" vs "direct POKER"
- Clearance: "cleared for takeoff" vs "hold position"
- Weather: "wind two seven zero at one five" vs "wind two
      niner zero
  at one five"

--- CONTEXTUALLY EQUIVALENT (same meaning - NOT errors) ---
- ICAO phonetics (use table): "tree"=3, "fife"=5, "niner"=9
- Format variants: "FL350"="flight level three five zero"
- Number formatting: "4091"="four zero niner one",
      "27"="two seven"
- Minor grammar: "descend to altitude"="descend altitude"
- Spoken abbreviations: "QNH"="QNH", "ILS"="I L S"
- Word order preserving meaning: "runway two seven cleared
      to land"=
  "cleared to land runway two seven"
- Minor synonyms: "proceed"="continue"

--- IGNORE COMPLETELY ---
Filler/acknowledgements: roger, wilco, affirmative,
      negative, okay,
thank you, good day, copy that, understood, will do, with
      you,
checking in. Spacing: "csa"="c s a". Articles when meaning
      unchanged.

--- ENTITY TYPES & PRIORITY (most specific first) ---
1. callsign   2. runway   3. altitude   4. frequency
5. heading    6. direction   7. clearance   8. weather
9. navigation   10. other

--- REASONING ---
<reasoning>
1. Normalize both transcripts using ICAO and equivalence
      rules.
2. Extract: callsign, direction, altitude, heading,
      runway, frequency,
   navigation, clearance, weather.
3. Compare element by element. Any non-equivalent
      difference = CRITICAL.
4. For each critical error: assign highest-priority
      entity_type; copy
   exact words as reference_value / transcribed_value; set
      safety_risk:
   high=runway/altitude/clearance/direction,
   medium=callsign/frequency/heading,
      low=navigation/weather.
5. Critical error exists -> "Critical_Errors",
      meaning_preserved=false.
   Else -> "Equivalent", meaning_preserved=true.
6. operational_safety: any high-risk -> "unsafe";
   any medium-risk -> "potentially_unsafe"; else -> "safe".
</reasoning>

--- OUTPUT JSON (no extra text, no markdown) ---
{
  "transcription_quality": "Equivalent|Critical_Errors",
  "meaning_preserved": true|false,
  "operational_safety": "safe|potentially_unsafe|unsafe",
  "explanation": "One concise sentence.",
  "transcription_errors": [
    {
      "entity_type":
      "runway|altitude|clearance|heading|direction|
        callsign|frequency|navigation|weather|other",
      "error_name": "wrong_{entity_type}",
      "reference_value": "exact words from reference",
      "transcribed_value": "exact words from
      transcription",
      "safety_risk": "high|medium|low",
      "explanation": "One sentence explaining this error."
    }
  ]
}
If no critical errors, output "transcription_errors": [].
```

A.2.2 Compressed

Retains the same logical structure as CoT but removes all concrete examples and condenses the ICAO table and rules to minimal bullet points. Tests whether instruction verbosity affects judgment reliability.

```
You are an expert ATC evaluator. Compare reference vs
      transcription
and detect ONLY operationally significant meaning-changing
      errors.
Ignore filler words and equivalent ICAO variants.

REFERENCE: {reference}
TRANSCRIPTION: {transcription}

--- ICAO PHONETICS ---
1=one/wun, 2=two/too, 3=tree/three, 4=fower/four,
      5=fife/five,
6=six, 7=seven, 8=ait/eight, 9=niner/nine, 0=zero
A=alfa/alpha, B=bravo, C=charlie, D=delta, E=echo,
      F=foxtrot,
G=golf, H=hotel, I=india, J=juliett, K=kilo, L=lima,
      M=mike,
N=november, O=oscar, P=papa, Q=quebec, R=romeo, S=sierra,
      T=tango,
U=uniform, V=victor, W=whiskey, X=xray, Y=yankee, Z=zulu

--- CRITICAL ERROR TYPES ---
Callsign, runway, altitude, heading, frequency, direction
      (left/right),
clearance, navigation, weather. Differences are CRITICAL
      if they change
any of the above.

--- IGNORE ---
Filler words, spacing/tokenization, and ICAO equivalents
      (tree=3,
fife=5, niner=9). Wording/grammar changes that preserve
      meaning.

--- RULES ---
1. Normalize ICAO + equivalents.
2. Extract key entities.
3. Compare meaning.
4. Mark ONLY non-equivalent differences as errors.
5. Assign entity_type to highest priority category.
6. Critical error -> "Critical_Errors",
      meaning_preserved=false.
   Else -> "Equivalent", meaning_preserved=true.
7. Safety: runway/altitude/clearance/direction -> unsafe;
   callsign/frequency/heading -> potentially_unsafe;
      others -> safe.

--- OUTPUT JSON ONLY ---
{
  "transcription_quality": "Equivalent|Critical_Errors",
  "meaning_preserved": true|false,
  "operational_safety": "safe|potentially_unsafe|unsafe",
  "explanation": "one sentence summary",
  "transcription_errors": [
    {
      "entity_type":
      "callsign|runway|altitude|heading|frequency|
        direction|clearance|navigation|weather|other",
      "error_name": "snake_case_label",
      "reference_value": "exact phrase",
      "transcribed_value": "exact phrase",
      "safety_risk": "high|medium|low",
      "explanation": "one sentence"
    }
  ]
}
If no critical errors: return empty transcription_errors
      array.
```

A.2.3 No-Examples

Retains full rule structure and ICAO normalisation instructions but removes all concrete examples of critical errors and equivalences. Tests whether few-shot exemplars are necessary for reliable judgment or whether rules alone suffice.

```
You are an expert ATC evaluator. Compare reference vs
      transcription
and identify ONLY operationally significant
      meaning-changing
differences. Ignore fillers and context-preserving
      variants.

REFERENCE: {reference}
TRANSCRIPTION: {transcription}

--- ICAO NORMALIZATION ---
Convert spoken ICAO words into standard digits/letters and
      treat
equivalent pronunciations as identical.

--- ERROR DEFINITIONS ---
A critical error is any difference that changes
      operational meaning in:
callsign, runway, altitude, heading, frequency, direction
      (left/right),
clearance, navigation, weather.

--- IGNORE ---
Fillers, acknowledgements, spacing differences, ICAO
      phonetic variants,
and wording differences that preserve meaning.

--- DECISION RULE ---
1. Normalize both inputs.
2. Extract operational entities.
3. Compare semantic meaning at entity level.
4. Mark differences as critical ONLY if meaning changes.
5. Assign each error to the highest-priority entity type.
6. Critical error -> "Critical_Errors",
      meaning_preserved=false.
   Else -> "Equivalent", meaning_preserved=true.

--- SAFETY RULE ---
runway/altitude/clearance/direction -> unsafe;
callsign/frequency/heading -> potentially_unsafe;
navigation/weather -> safe.

--- OUTPUT (STRICT JSON ONLY) ---
{
  "transcription_quality": "Equivalent|Critical_Errors",
  "meaning_preserved": true|false,
  "operational_safety": "safe|potentially_unsafe|unsafe",
  "explanation": "one sentence summary",
  "transcription_errors": [
    {
      "entity_type":
      "callsign|runway|altitude|heading|frequency|
        direction|clearance|navigation|weather|other",
      "error_name": "snake_case_label",
      "reference_value": "exact phrase from reference",
      "transcribed_value": "exact phrase from
      transcription",
      "safety_risk": "high|medium|low",
      "explanation": "one sentence"
    }
  ]
}
If no critical errors, return an empty
      transcription_errors list.
```

A.2.4 Lenient

Removes structured entity categories, ICAO normalisation, entity priority rules, and examples entirely. Instructs the model to rely on pure semantic judgment without predefined error types. This variant supports binary classification only and cannot produce fine-grained error categorisation by design.

```
You are an expert ATC evaluator. Compare reference and
      transcription
```

```
and determine whether the meaning is preserved or changed.
      Focus only
on whether the two utterances would lead to different
      operational
understanding.

REFERENCE: {reference}
TRANSCRIPTION: {transcription}

--- TASK ---
Determine if the transcription preserves the same meaning
      as the
reference in an aviation context. Ignore minor wording
      differences.
Focus on whether any change could alter instructions,
      intent, or
safety-critical meaning.

--- DECISION RULE ---
1. Read both utterances.
2. Compare overall meaning in context.
3. Decide if any meaningful difference exists:
   - Meaning preserved -> "Equivalent"
   - Meaning changes   -> "Critical_Errors"
4. Infer operational risk based on severity of meaning
      change.

--- OUTPUT RULES ---
Do NOT rely on predefined categories. Do NOT assume
      structured error
types. Do NOT normalize phonetics explicitly. Use only
      semantic
judgment.

--- OUTPUT (STRICT JSON ONLY) ---
{
  "transcription_quality": "Equivalent|Critical_Errors",
  "meaning_preserved": true|false,
  "operational_safety": "safe|potentially_unsafe|unsafe",
  "explanation": "one sentence summary",
  "transcription_errors": []
}
If no meaningful error exists, return an empty
      transcription_errors
list.
```

A.3 Full Finetuning Paradox Results

Table 5 extends the main paper results by reporting absolute percentages of critical errors that fall in the low-metric region (below the 25th percentile threshold derived from pretrained distributions) for every metric, model, and dataset combination presented in Table 1. **Pre** and **FT** columns show what fraction of critical errors are hidden in the "good" region of each metric before and after finetuning. The Δ column quantifies how much worse each metric becomes at detecting critical errors after finetuning. Large positive Δ values indicate that finetuning causes standard metrics to become systematically overconfident. WWER tracks WER closely across all conditions, confirming that rule-based entity weighting does not resolve this failure mode.

A.4 Full Inter-Rater Agreement

Tables 6 and 7 provide the complete inter-rater agreement results summarised in Table 2, main paper . Table 6 reports Cohen's κ and population-reweighted weighted κ for all rater pairs. Weighted

Dataset	Model	Metric	Pre (%)	FT (%)	Δ (pp)
ATCO2	W	WER	14.1	38.1	+24.1
		WWER	10.1	37.6	+27.5
		BERTDist	13.9	33.7	+19.8
		SemDist	17.3	36.5	+19.3
	P	WER	11.2	25.4	+14.2
		WWER	8.7	23.9	+15.3
		BERTDist	10.5	18.5	+8.0
		SemDist	12.0	25.4	+13.3
ATCC	W	WER	6.5	41.4	+34.9
		WWER	6.2	44.5	+38.3
		BERTDist	8.3	48.8	+40.5
		SemDist	8.6	47.1	+38.5
	P	WER	3.0	25.8	+22.8
		WWER	2.6	23.0	+20.4
		BERTDist	3.1	20.4	+17.3
		SemDist	2.5	25.7	+23.2
ATCOSim	W	WER	28.5	82.2	+53.7
		WWER	30.6	81.2	+50.6
		BERTDist	31.0	82.7	+51.7
		SemDist	33.3	79.3	+46.0
	P	WER	47.2	79.1	+31.9
		WWER	39.1	73.0	+33.9
		BERTDist	36.0	67.2	+31.2
		SemDist	38.0	73.7	+35.6

Table 5: Full Finetuning Paradox results. W = Whisper-medium, P = Parakeet-0.6B. **Pre/FT**: percentage of critical errors hidden in the low-metric region (below 25th-percentile threshold) before/after finetuning. Δ: percentage-point increase after finetuning. Large Δ indicates the metric becomes less reliable at flagging safety-critical errors after finetuning. WWER tracks WER closely, confirming that rule-based entity weighting does not resolve the paradox.

κ corrects for the deliberate oversampling of disagreement cases in our annotation protocol by applying inverse probability weights derived from the natural corpus distribution (9,988 agreed-critical, 4,729 agreed-equivalent, 5,822 disagreed). Table 7 reports precision, recall, F1, and accuracy for each individual judge and the ensemble against the human consensus gold standard.

A.5 Prompt Robustness Full Results

The main paper Table 3 reports aggregate robustness scores; Table 8 gives the full per-variant breakdown and Table 9 decomposes each judge's output into five mutually exclusive outcome categories. *Consistent correct*: all prompt variants agree and are correct. *Consistent wrong*: all agree and are wrong. *Recoverable*: at least one variant is correct (prompt-sensitive but not catastrophic). *Catastrophic*: all variants are wrong. *Incomplete*: not all variants yielded parseable JSON output. Qwen achieves the highest consistency (Fleiss $\kappa = 0.836$,

Rater Pair	κ	Wtd. κ
Human Agreement (Ceiling)		
Human_1 – Human_2	0.650	0.665
Human – LLM		
Human_1 – Llama	0.511	0.560
Human_1 – Qwen	0.514	0.579
Human_1 – DeepSeek	0.538	0.574
Human_2 – Llama	0.546	0.622
Human_2 – Qwen	0.463	0.567
Human_2 – DeepSeek	0.646	0.699
Human_1 – LLM Consensus	0.587	0.618
Human_2 – LLM Consensus	0.599	0.665
LLM – LLM		
Llama – Qwen	0.536	0.653
Llama – DeepSeek	0.524	0.621
Qwen – DeepSeek	0.365	0.509
Fleiss κ (3 LLMs)	0.506	0.607

Table 6: Full pairwise inter-rater agreement on 500 SME-annotated samples. Weighted κ corrects for oversampling bias using inverse probability weights from the natural corpus distribution. Human_1 vs LLM Consensus ($\kappa = 0.587$, wtd. $\kappa = 0.618$) approaches the human-human ceiling ($\kappa = 0.650$).

Judge	P	R	F1	Acc
Llama-3.3-70B	0.841	0.809	0.825	0.806
Qwen-2.5-72B	0.920	0.656	0.766	0.774
DeepSeek-R1-70B	0.820	0.872	0.845	0.820
LLM Consensus	**0.882**	**0.851**	**0.866**	**0.852**
Human_1 vs Consensus	0.779	0.831	0.805	0.794

Table 7: Judge performance against human consensus gold standard ($n = 500$). P/R/F1 are computed for the *Critical_Errors* class. Qwen's low recall indicates systematic under-detection, motivating ensemble over any single judge. The final row reports Human_1 against LLM Consensus as a symmetric view of agreement.

73.4% consistently correct) and the lowest incomplete rate, making it the most robust individual judge. Llama has a high incomplete rate (41.4%), primarily from the compressed variant failing to return valid JSON on a subset of samples.

B Fine-Grained Error Categorisation

The main paper reports binary (Equivalent / Critical_Errors) agreement; here we examine whether LLM judges also agree with human annotators on *which type* of error was made. Tables 10 and 11 are computed on the 150 agreed-critical samples from our human annotation set. Table 12 reports each judge's error-type distribution across all critical predictions in the full evaluation corpus.

Frequent classes (callsign 70.5%, clearance 16.8%, runway 18.1%) achieve moderate-to-substantial agreement with human annotators; rare classes (heading and altitude at 1.3% each) show weak or near-zero κ due to insufficient sample support. Among frequent classes, frequency ($\kappa = 0.846$, Qwen) and runway ($\kappa = 0.684$, Qwen) achieve the strongest human agreement; navigation and other show the weakest, reflecting genuinely overlapping failure modes. LLM–LLM pairwise agreement (Table 11) is generally higher than human–LLM agreement, particularly for structured categories. The heading anomaly ($\kappa = 1.000$, Llama–Qwen) is an artefact: both models consistently missed the same two annotated instances, producing spurious perfect agreement.

Judge	Prompt	Acc	F1	N	Fleiss' κ
Qwen	`cot`	**0.902**	**0.916**	297	0.836
	`no_examples`	0.875	0.884	297	
	`lenient`	0.876	0.878	274	
	`compressed`	0.869	0.874	297	
Llama	`cot`	**0.892**	**0.908**	297	0.771
	`lenient`	0.831	0.834	296	
	`compressed`	0.855	0.780	179	
	`no_examples`	0.781	0.724	247	
DeepSeek	`compressed`	**0.879**	**0.893**	297	0.695
	`lenient`	0.864	0.880	295	
	`no_examples`	0.832	0.844	297	
	`cot`	0.825	0.841	297	

Table 8: Per-variant prompt robustness results. Fleiss' κ measures inter-prompt consistency for each judge (higher = more stable across prompt variants). The CoT variant achieves the highest accuracy and F1 for Qwen and Llama; DeepSeek performs best with Compressed. N is the number of samples with parseable output.

Judge	Cons. Correct	Cons. Wrong	Recoverable	Catastrophic	Incomplete
Qwen	73.4%	5.4%	6.4%	7.1%	7.7%
Llama	45.8%	3.4%	4.4%	5.1%	41.4%
DeepSeek	67.7%	4.0%	19.5%	8.1%	0.7%

Table 9: Robustness breakdown per judge across all prompt variants. *Recoverable*: at least one prompt variant is correct. *Catastrophic*: all prompt variants are wrong. *Incomplete*: not all variants yielded valid JSON output. DeepSeek shows the highest recoverability (19.5%) but also higher catastrophic errors (8.1%) than Qwen.

Class	H%	Ll κ	Qw κ	DS κ	Best F1	Judge
Callsign	70.5	0.398	0.275	**0.469**	0.874	DS
Runway	18.1	0.125	**0.684**	0.504	0.746	Qw
Clearance	16.8	0.339	**0.634**	0.362	0.694	Qw
Navigation	13.4	0.197	0.273	**0.323**	0.387	DS
Other	32.9	0.103	**0.326**	0.200	0.542	Qw
Weather	8.1	0.143	**0.686**	0.608	0.714	Qw
Frequency	6.7	0.554	**0.846**	0.831	0.857	Qw
Altitude	1.3	0.493	**0.661**	0.390	0.667	Qw
Heading	1.3	0.0	0.0	**0.321**	0.333	DS

Table 10: Per-class agreement between each LLM judge and human annotator on 150 agreed-critical samples. H% = human annotation frequency. Ll=Llama, Qw=Qwen, DS=DeepSeek. Bold = best κ per class.

Class	Ll–Qw	Ll–DS	Qw–DS
Callsign	0.605	0.573	0.538
Runway	0.153	0.288	0.476
Clearance	0.353	0.434	0.472
Navigation	0.502	0.476	0.521
Other	0.229	0.160	0.383
Weather	0.106	0.172	0.665
Frequency	0.514	0.601	0.786
Altitude	0.321	0.390	0.561
Heading	1.000	0.394	0.394

Table 11: Pairwise LLM–LLM Cohen's κ per error class on 150 agreed-critical samples. The Llama–Qwen heading value of 1.000 is an artefact of both models missing the same two instances.

Error Type	Llama	Qwen	DeepSeek
Callsign	75.2%	82.8%	83.5%
Clearance	8.2%	11.5%	8.9%
Other	6.8%	31.6%	21.1%
Runway	4.2%	14.4%	10.5%
Navigation	3.4%	10.1%	9.3%
Altitude	1.9%	3.8%	3.6%
Weather	1.6%	7.5%	4.7%
Frequency	1.1%	6.7%	3.9%
Heading	0.8%	1.0%	1.6%
Multi-label	298	6,351	5,426

Table 12: Error-type distribution per judge across all critical predictions in the full evaluation corpus. Percentages may exceed 100% due to multi-label assignments. Qwen assigns multiple error types per utterance substantially more often than Llama (6,351 vs 298 multi-label instances), reflecting a more aggressive secondary-category assignment strategy.