Benchmarking Streaming ASR for Real-time Deployment: A Robustness Scorecard and Error Taxonomy for Vietnamese

Hoai An THAI[1][0009-0004-4093-522X], Duy Tan NGUYEN[1][0009-0000-2832-8647], Vy LE[1][0009-0003-0387-8302], and Thi Thuy Duong NGUYEN[1][0009-0000-3548-2212]

UEH College of Technology and Design, University of Economics Ho Chi Minh City
{anthai.31231025020, tanguyen.31231023384, vyle.31231022128, duongnguyen.31231022904}@st.ueh.edu.vn

Abstract. Vietnamese ASR is increasingly deployed in production, where offline word error rate (WER) alone is insufficient for model selection under streaming latency and runtime budgets. We present a reproducible streaming evaluation framework that standardizes chunking, right context (look-ahead), overlap handling, and metric computation, and reports WER and real-time factor (RTF) under multiple deployment-oriented latency profiles (1200/2400/4000 ms) across representative model families. We evaluate on VIVOS, VLSP2020, Viet YouTube ASR v2, and Speech-MASSIVE_vie to produce a robustness scorecard under streaming stressors and distribution shifts. We further provide a preliminary Vietnamese-specific error taxonomy (numerals, punctuation, named entities, and Vietnamese–English code-switching) to translate benchmark outcomes into deployment guidance.

Keywords: Vietnamese ASR · Streaming evaluation · Latency · Real-time factor · Robustness

1 Introduction

Vietnamese automatic speech recognition (ASR) is widely used in production. Model selection, however, is still often driven by offline word error rate (WER). This is misaligned with streaming deployment, where systems must satisfy latency budgets and limited compute [?,?]. The problem is sharper for Vietnamese because evaluation data often differ from real usage: read speech vs. spontaneous and in-the-wild speech, combined with strong accent variability [?,?,?].

We present a system-oriented benchmark for Vietnamese streaming ASR. Our contributions are:

- We define a reproducible streaming evaluation protocol (with a reference implementation) covering chunking, look-ahead, overlap handling, and metric computation.

- We report WER, ΔWER (streaming minus offline), and RTF under three streaming latency profiles representative of practical deployment.
- We provide a robustness scorecard measuring degradation under streaming stressors and realistic distribution shifts.
- We provide a preliminary Vietnamese-specific error taxonomy and summarize practical implications for deployment.

2 Related Work

Evaluation in automatic speech recognition (ASR) has traditionally been dominated by offline accuracy metrics, most notably word error rate (WER). While WER remains a useful indicator of transcription quality, it abstracts away operational constraints that arise in production systems. In streaming ASR, recognition must be performed under limited right context and bounded computation, yielding an explicit accuracy–latency trade-off. Prior work has shown that design choices such as chunk size, look-ahead, and overlap/context management can have a direct and sometimes non-linear impact on both WER and latency, making it difficult to compare models when streaming configurations are reported inconsistently [**?**,**?**].

To better reflect deployment constraints, several toolkits and evaluation efforts have started to report runtime-related metrics such as end-to-end latency, real-time factor (RTF), and throughput, reframing ASR evaluation as a multi-objective problem rather than a single-metric ranking. Production-oriented toolkits (e.g., WeNet) provide configurable streaming decoding and facilitate reporting of latency/RTF alongside WER, improving reproducibility at the system level [**?**]. More recent initiatives, such as open leaderboards, further emphasize transparency and standardized reporting, but they often remain offline-centric in ranking criteria or provide limited support for streaming-specific controls and stress testing [**?**].

For Vietnamese ASR, the gap between benchmark evaluation and real-world deployment is amplified by distribution shifts and linguistic characteristics. Public Vietnamese datasets and strong pretrained checkpoints have accelerated progress, yet many evaluations are still conducted on relatively clean read speech, while deployed systems must handle spontaneous and in-the-wild conditions and pronounced accent variability [**?**,**?**,**?**]. In addition, Vietnamese ASR outputs are sensitive to text normalization decisions (e.g., numerals, punctuation/casing) and to systematic error patterns involving named entities and Vietnamese–English code-switching, which are not fully captured by aggregate WER.

Robustness under distribution shift has been studied through controlled stress tests involving additive noise, speaking rate variation, reverberation, and domain mismatch, consistently showing substantial degradation outside the training distribution. However, robustness analysis is rarely integrated into streaming-oriented benchmarks, despite the fact that streaming constraints can further exacerbate failures by restricting available context and increasing partial-hypothesis uncertainty. Consequently, existing evaluations often treat streaming, efficiency, and robustness as separate concerns, leaving a need for a unified and reproducible Vietnamese-focused benchmark that jointly characterizes WER, RTF, and streaming degradation under realistic shifts, with actionable error patterns for deployment. Table ?? summarizes these gaps and positions our contribution.

Table 1. Comparison of prior ASR evaluation practices and benchmarks.

Work	Language	Streaming	Latency / RTF	Robustness	Error Analysis	Reproducible
Representative Streaming ASR models [?]	Multi	Yes	Partial	No	No	Partial
WeNet toolkit [?]	Multi	Yes	Yes	No	No	Yes
Open ASR Leaderboard [?]	Multi	No	Yes	No	No	Yes
Vietnamese ASR benchmarks [?,?]	Vietnamese	No	No	Partial	No	Partial
This work	Vietnamese	Yes	Yes	Yes	Yes	Yes

3 Evaluation Pipeline and Setup

The evaluation pipeline is designed to be reproducible, computationally feasible, and representative of real-world streaming ASR scenarios. It consists of four stages: dataset preparation, streaming simulation and inference, multi-axis metric computation, and analysis. Figure ?? provides an overview of the pipeline.

We evaluate models on four publicly available Vietnamese speech datasets selected to reflect diverse deployment conditions. VIVOS serves as a clean read-speech baseline, providing an upper-bound reference under favorable acoustic conditions [?]. VLSP2020 represents a standardized Vietnamese benchmark and includes spontaneous speech, enabling evaluation under controlled domain shift [?]. The Vietnamese YouTube ASR Corpus v2 captures in-the-wild speech across diverse recording conditions and content domains [?], while Speech-MASSIVE (Vietnamese) consists of short, intent-driven utterances representative of assistant-style use cases with strict latency requirements [?]. For all latency-sweep experiments, we use fixed 300-utterance subsets per dataset, yielding: 1155.2 s (VIVOS), 2808.4 s (VLSP2020), 768.4 s (Viet YouTube ASR v2), and 1156.8 s (Speech-MASSIVE_vie), i.e., 5888.8 s in total (∼1.64 h). All audio is resampled to a consistent sampling rate and converted to mono, and reference transcripts are normalized using a unified text normalization procedure to ensure fair comparison across models and settings.

The evaluation focuses on representative ASR model families with publicly available checkpoints. ChunkFormer is included as a streaming-native, chunk-based model and constitutes the primary subject of streaming analysis. wav2vec2-base (Vietnamese, 250h) is used as a lightweight CTC-based baseline amenable to streaming simulation. PhoWhisper-large represents a strong Vietnamese-specific offline upper bound, while Whisper-large-v3 serves as a multilingual baseline. Deployment-oriented variants based on faster-whisper are left as future work due environment/toolchain compatibility constraints in this run. No model is retrained or fine-tuned; all results are obtained through inference-only evaluation to isolate the effects of streaming constraints and evaluation protocol choices.

figures/pipeline.png

Fig. 1. Overview of the evaluation pipeline.

Streaming inference is simulated by segmenting input audio into fixed-size chunks and restricting right context using a configurable look-ahead window. For streaming-capable models (ChunkFormer and wav2vec2), we evaluate three deployment-oriented profiles: 1200 ms, 2400 ms, and 4000 ms chunk size, each with 200 ms overlap and 0 ms look-ahead. Chunking and overlap handling are standardized across streaming-capable models, and offline decoding with full context is performed as a reference. Latency proxy is defined as chunk size plus look-ahead (thus 1200/2400/4000 ms in our sweep).

Word Error Rate (WER) is reported as the primary accuracy metric, together with 95% bootstrap confidence intervals for statistical reliability. Runtime efficiency is quantified using real-time factor (RTF), defined as the ratio of inference time to audio duration, and throughput is reported where applicable. To capture the effect of streaming constraints, accuracy degradation is computed as the difference between streaming and offline WER. All runtime measurements are conducted on a consistent hardware setup.

All components of the evaluation pipeline, including dataset subsets, streaming configurations, normalization rules, and inference scripts, are fully specified. Experiments are executed with fixed random seeds and can be reproduced via a single command-line interface, facilitating reuse and extension of the benchmark by the community.

4 Results and Analysis

Table **??** summarizes the overall Word Error Rate (WER) for all evaluated models under offline decoding and representative streaming configurations. Results are reported together with 95% bootstrap confidence intervals. Across datasets, offline decoding provides an upper-bound reference, while streaming settings exhibit varying degrees of accuracy degradation depending on model family.

Table 2. Overall WER (%) with 95% confidence intervals across datasets and decoding modes. Lower is better. N/A indicates no streaming evaluation for offline-only models.

Model	Dataset	Offline WER (95% CI)	Streaming WER (95% CI)	ΔWER
ChunkFormer-CTC-large (110M)	VIVOS	3.44 [2.68, 4.39]	6.17 [4.85, 7.57]	2.74
	VLSP2020	3.83 [3.26, 4.45]	50.64 [46.27, 54.42]	46.81
	Viet YouTube ASR v2	4.48 [3.83, 5.18]	5.44 [4.45, 6.83]	0.96
	Speech-MASSIVE_vie	18.78 [15.49, 22.28]	21.38 [17.75, 24.98]	2.60
wav2vec2-base-vi (95M)	VIVOS	8.88 [7.69, 10.22]	12.32 [10.54, 13.95]	3.44
	VLSP2020	10.09 [9.14, 11.35]	52.74 [49.03, 56.17]	42.65
	Viet YouTube ASR v2	7.90 [6.81, 8.86]	8.31 [7.02, 9.51]	0.40
	Speech-MASSIVE_vie	27.71 [24.12, 31.29]	30.38 [26.95, 34.14]	2.68
PhoWhisper-large (1.55B; offline)	VIVOS	3.01 [2.32, 3.80]	N/A	N/A
	VLSP2020	17.49 [13.74, 21.60]	N/A	N/A
	Viet YouTube ASR v2	10.55 [8.80, 12.69]	N/A	N/A
	Speech-MASSIVE_vie	17.85 [14.60, 20.78]	N/A	N/A
Whisper-large-v3 (1.55B; offline)	VIVOS	9.49 [8.27, 11.11]	N/A	N/A
	VLSP2020	24.91 [21.89, 28.25]	N/A	N/A
	Viet YouTube ASR v2	36.27 [30.08, 42.86]	N/A	N/A
	Speech-MASSIVE_vie	22.39 [18.98, 25.82]	N/A	N/A

To quantify runtime efficiency, Table **??** reports real-time factor (RTF) and throughput measured under identical hardware conditions. While larger models achieve stronger offline accuracy, they incur substantially higher runtime cost, highlighting the need for cost-aware model selection in deployment settings.

Figure **??** summarizes streaming WER under the latency-sweep setup (1200/2400/4000 ms chunks, 200 ms overlap, 0 ms look-ahead).

4.1 Latency–Accuracy Frontier Analysis

Figure **??** shows a consistent monotonic trend across datasets: moving from 1200 ms to 4000 ms reduces streaming WER for both streaming-capable models. Averaged over datasets, ChunkFormer improves from 34.8% to 20.9% WER

Table 3. Runtime efficiency measured by real-time factor (RTF) and throughput under identical hardware conditions. Lower RTF and higher throughput indicate better efficiency.

Model	RTF (median [IQR])	Throughput (xRT)	Hardware
ChunkFormer-CTC-large (110M)	0.0111 [0.0016]	90.1	1×NVIDIA H100 80GB
wav2vec2-base-vi (95M)	0.0059 [0.0009]	169.5	1×NVIDIA H100 80GB
PhoWhisper-large (1.55B)	0.1034 [0.0170]	9.7	1×NVIDIA H100 80GB
Whisper-large-v3 (1.55B)	0.0903 [0.0186]	11.1	1×NVIDIA H100 80GB

(−13.9 percentage points), while wav2vec2 improves from 43.5% to 25.9% (−17.6 percentage points). This confirms that latency configuration is a first-order factor in deployment quality.

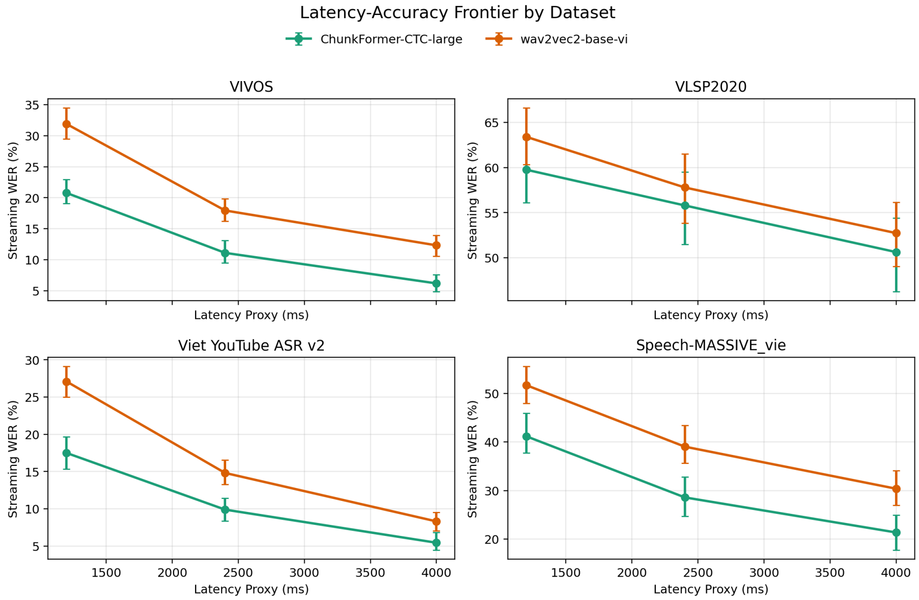

Fig. 2. Latency–accuracy frontier (streaming WER with 95% CI) for ChunkFormer and wav2vec2 under 1200/2400/4000 ms profiles.

Figure **??** reports ΔWER (streaming minus offline) along the same latency sweep. Across all datasets and both models, ΔWER decreases as chunk size increases, indicating more stable context integration at higher latency budgets. VLSP2020 remains the hardest condition at all latency points, with large residual degradation even at 4000 ms.

4.2 VLSP2020 Failure Analysis

The VLSP2020 degradation is systematic rather than an isolated outlier. At 4000 ms, both models remain above 50% streaming WER (ChunkFormer 50.64%, wav2vec2 52.74%), despite low offline WER (3.83% and 10.09%, respectively). This gap is accompanied by strong instability signals: at 4000 ms, VLSP2020 has the highest average chunks per utterance (2.97 versus 1.05–1.52 on other datasets) and the highest change rate (∼80% for both models). At 1200 ms, average chunks per utterance increase to 9.87 and change rate rises to about 91–92%. These results suggest boundary-sensitive failure under multi-chunk decoding, where frequent partial-hypothesis revision amplifies domain mismatch. This behavior should be interpreted as a deployment risk for long or highly variable utterances.

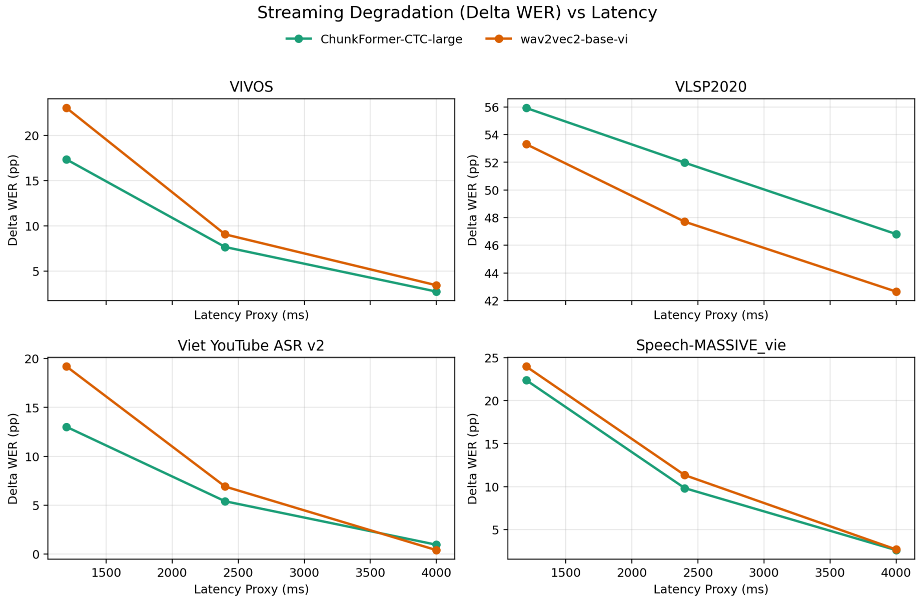

Fig. 3. Latency sweep for streaming degradation (*Δ*WER in percentage points). Lower is better.

4.3 Streaming Stability

We further analyze incremental-hypothesis stability using change rate and edit-overhead metrics. Figure **??** shows that stability improves markedly as latency increases: mean change rate drops from approximately 81% (1200 ms) to 31% (4000 ms), and mean edit overhead drops from approximately 1.55 to 0.44.

VLSP2020 is consistently the least stable condition, with change rates around 0.88–0.92 at 1200/2400 ms and still around 0.80 at 4000 ms.

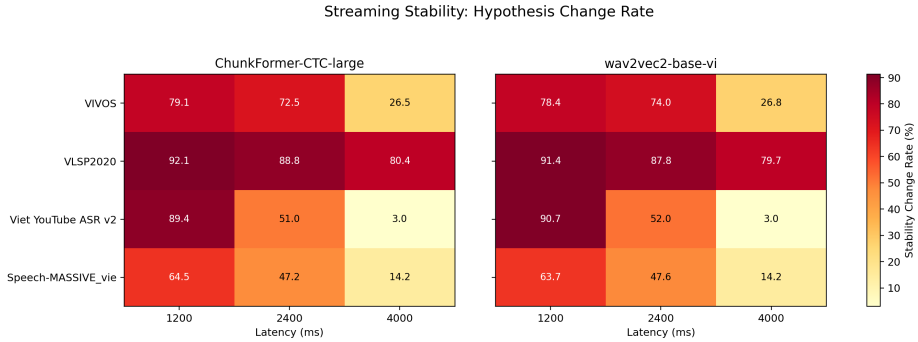

Fig. 4. Streaming stability heatmap (hypothesis change rate, %) across datasets and latency profiles.

Table **??** reports accuracy degradation under distribution shifts, including spontaneous speech, in-the-wild recordings, and assistant-style utterances. For each condition, we report both absolute and relative shifts against the clean read-speech baseline (VIVOS) at 4000 ms. Reporting both metrics avoids over-emphasis from ratio-only views when the clean baseline is small.

4.4 Robustness with Absolute and Relative Shift

To make robustness comparison explicit, we define a model-wise robustness index at fixed latency:

$$\mathrm{RI}(d, m) = \frac{\mathrm{WER}_{\mathrm{stream}}(d, m)}{\mathrm{WER}_{\mathrm{stream}}(\mathrm{VIVOS}, m)}.$$

This is equivalent to $1 + \frac{\text{Relative Shift}}{100}$, and complements absolute shifts (percentage points).

Table 4. Robustness scorecard at 4000 ms streaming profile. Absolute shift is in percentage points (pp) from VIVOS. Relative shift is % change from VIVOS; negative values indicate improvement.

Model	Dataset	WER (%)	Absolute Shift (pp)	Relative Shift (%)
ChunkFormer-CTC-large	VIVOS	6.17	0.00	0.0
	VLSP2020	50.64	+44.46	+720.2
	Viet YouTube ASR v2	5.44	-0.73	-11.9
	Speech-MASSIVE_vie	21.38	+15.21	+246.3
wav2vec2-base-vi	VIVOS	12.32	0.00	0.0
	VLSP2020	52.74	+40.42	+328.2
	Viet YouTube ASR v2	8.31	-4.01	-32.6
	Speech-MASSIVE_vie	30.38	+18.07	+146.7

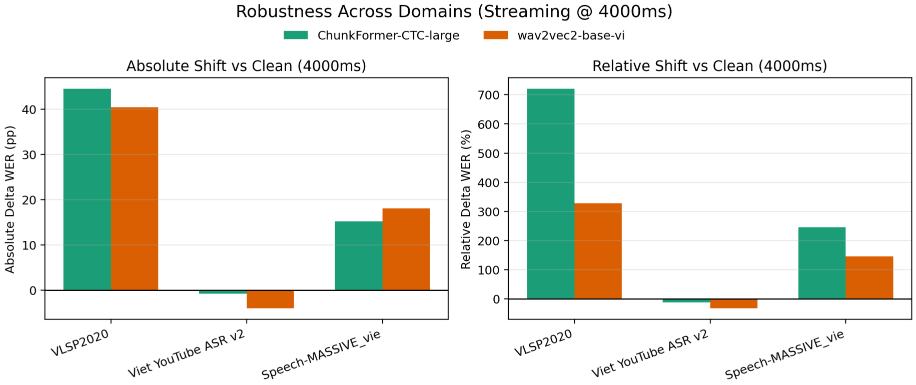

Fig. 5. Robustness comparison at 4000 ms profile: absolute and relative shifts from VIVOS.

Based on the observed latency-sweep results, we derive deployment-oriented insights. ChunkFormer consistently outperforms wav2vec2 across all datasets and all three latency profiles, but both models degrade substantially on VLSP2020. These findings highlight that offline rankings are insufficient for deployment decisions, and latency-profile-aware streaming evaluation should be reported alongside offline WER.

5 Conclusion and Future Work

This paper presents a system-oriented benchmark for Vietnamese streaming ASR that moves beyond offline accuracy-centric evaluation. By introducing a unified and reproducible evaluation pipeline, we systematically assess ASR models under realistic streaming constraints and report recognition accuracy and runtime efficiency under deployment-oriented latency profiles. Our results demonstrate that model selection based solely on offline WER can be misleading in deployment scenarios, as streaming constraints and runtime cost substantially alter the relative performance of different model families.

Through latency-profile comparisons of WER, ΔWER, and RTF, we highlight model configurations that provide favorable accuracy–efficiency behavior for deployment. In addition, the proposed robustness scorecard reveals that streaming constraints often amplify performance degradation under distribution shifts, including spontaneous speech and in-the-wild conditions that are common in Vietnamese applications. Our preliminary Vietnamese-specific error taxonomy (numerals, abbreviations, named entities, and code-switching) exposes likely failure patterns and motivates future quantitative error breakdown.

This work focuses on evaluation rather than model training, and several limitations remain. We do not retrain or fine-tune models under streaming constraints, and robustness analysis is limited to a fixed set of datasets and stress conditions. Although we evaluate three practical latency profiles (1200/2400/4000 ms), we do not claim an exhaustive continuous latency–accuracy frontier. RTF results are measured on a single H100 80GB setup and should be interpreted as throughput upper bounds rather than edge-deployment guarantees. Future work may extend this benchmark by incorporating additional streaming architectures, low-resource adaptation techniques, explicit latency-budget sweeps, and more fine-grained latency measurements that account for end-to-end system effects. We also plan to expand the robustness analysis to cover a broader range of acoustic and linguistic variations, and to integrate the benchmark into an open, continuously updated evaluation platform for Vietnamese ASR.

By providing an open, reproducible, and deployment-aligned evaluation framework, we hope this work serves as a reference point for future research and practical system development in Vietnamese streaming ASR.