Scaling Laws or Threshold Effects: Exploring the Optimal Vocabulary Size for Balancing Performance and Efficiency in Low-Resource Languages

Anonymous ACL submission

Abstract

While vocabulary expansion scaling laws are well-established for high-resource languages, they remain unverified in low-resource settings. This gap is particularly critical for Byte-level BPE (BBPE), where constrained vocabulary sizes often fail to capture the rich morphemes of complex scripts, leading to severe over-segmentation in languages such as Mongolian, Tibetan, and Uyghur. We systematically investigate jointly-scaled trilingual vocabulary for these languages (140 to 195,000 tokens) across BPE (Llama 2) and BBPE (Qwen2.5/3) architectures. Our results reveal that BBPE follows a "decline-then-rise" pattern, requiring a 9,000-token threshold (3,000 per language) to trigger non-linear performance gains and inference acceleration, whereas BPE improves monotonically. Using Pareto Frontier Analysis, we identify an optimal 79,500-token configuration for BBPE that reduces continual pre-training duration by over 71% across 1.5B to 8B parameter models while consistently enhancing downstream performance.[1]

1 Introduction

Despite the rapid advancements in Large Language Models (LLMs), a structural *tokenization disparity* persists, where token counts for identical semantics vary by up to tenfold across languages (Limisiewicz et al., 2023). This issue is particularly acute for morphologically rich, low-resource languages such as Mongolian, Tibetan, and Uyghur, which exhibit significantly higher token fertility than English (Ahia et al., 2023). While vocabulary expansion is the standard remedy to alleviate this fragmentation (Wang et al., 2020; Lu et al., 2025), its implementation in low-resource regimes remains largely heuristic. Lacking a principled empirical basis, researchers often select vocabulary sizes arbitrarily, struggling to balance the mitigation of severe over-segmentation against the risk of representation instability caused by sparsely-distributed, under-trained tokens.

This lack of guidance is primarily rooted in the limitations of existing *Scaling Laws*, which have traditionally focused on the synergy between model parameters and data volume (Kaplan et al., 2020; Hoffmann et al., 2022) while often overlooking the vocabulary dimension. Although recent studies (Takase et al., 2025; Tao et al., 2024) have begun to explore vocabulary scaling, they remain predominantly English-centric, failing to address the unique agglutinative or inflectional structures of low-resource scripts. Crucially, the scaling properties of BBPE—a mainstream tokenizer that eliminates out-of-vocabulary (OOV) issues by using UTF-8 bytes as base units—remain unquantified in data-sparse scenarios. Consequently, whether the scaling behaviors established for high-resource languages generalize to models utilizing BBPE (Radford et al., 2019) in low-resource settings requires rigorous empirical verification, as paradigm-specific scaling properties may necessitate a distinct framework.

To bridge this gap, we systematically investigate vocabulary scaling for three non-Latin-script, low-resource languages: Mongolian, Tibetan, and Uyghur (as illustrated in Figure 1). We propose a joint trilingual expansion strategy to facilitate unified regional-level models, executing controlled experiments across ten scaling levels (140 to 195,000 tokens, with equal allocation per language) on Qwen3-8B (Yang et al., 2025) (representing BBPE) and Llama 2-7B (Touvron et al., 2023) (representing BPE (Sennrich et al., 2016)). To establish generalizability, we replicate these experiments on the Qwen2.5 series (1.5B and 7B) (Bai et al., 2025). We evaluate performance across four key tasks: summarization, text classification, word segmentation, and translation from Mongolian, Tibetan, and

[1] Our code, data, and models will be made publicly available upon acceptance.

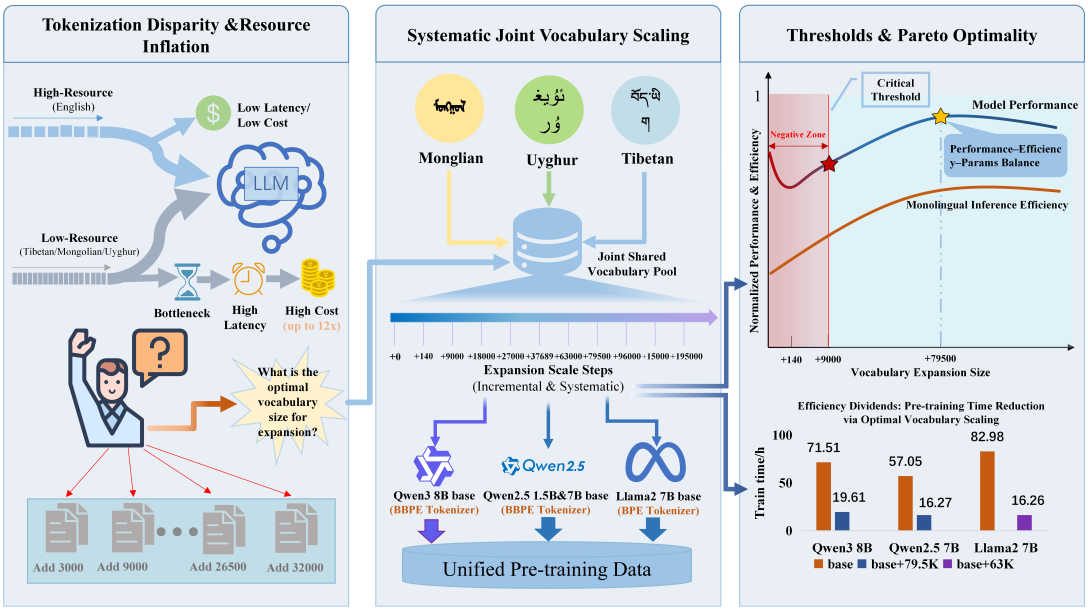


Figure 1: Overview of the research framework: analyzing tokenization disparities, conducting systematic multi-scale vocabulary scaling experiments, and identifying the Pareto-optimal configuration.

Uyghur into Chinese. The results demonstrate that BBPE scaling properties remain robust across varying model capacities and architectural iterations. Our core contributions are as follows:

(1) **Quantification of the BBPE Threshold Effect:** We identify a critical initiation threshold (∼ 9,000 total tokens) and a subsequent *performance-decline region*. Below this boundary, insufficient vocabulary disrupts pre-trained byte-level representations, leading to performance degradation. This provides a principled guide for the initial stage of BBPE expansion.

(2) **Efficiency-Performance Pareto Optimality:** We pinpoint a universal "optimal point" at 79,500 tokens for BBPE architectures. This configuration curtails continual pre-training duration by over 71% across 1.5B–8B models while enhancing downstream task performance and delivering significant inference efficiency gains, achieving an average of over two-fold throughput gains across various monolingual tasks.

(3) **Method-Dependent Scaling Framework:** We demonstrate that vocabulary scaling dynamics are contingent upon the choice of tokenization method. While BPE-based models exhibit monotonic improvements that yield diminishing marginal returns beyond ∼27,000 tokens, BBPE requires surpassing a critical threshold to unlock non-linear benefits. Our framework provides empirical guidance for optimizing expansion scales, offering actionable insights tailored to the underlying tokenization method.

2 Related Work

The performance of large language models (LLMs) is traditionally described by power laws relative to model parameters and training data volume (Kaplan et al., 2020; Hoffmann et al., 2022). However, the role of vocabulary size (V) as a critical determinant of computational efficiency and model efficacy has recently gained prominence. Tao et al. (2024) theoretically established a FLOPs-optimal matching relationship between vocabulary size and parameters, while Takase et al. (2025) provided empirical evidence that expanding vocabularies can directly enhance downstream performance and training efficiency. Despite these insights, existing investigations remain predominantly English-centric and benchmarked against standard BPE algorithms. The scaling properties of BBPE in morphologically rich, low-resource contexts remain largely underexplored, leaving an empirical gap in providing a principled basis for determining expansion scales in data-sparse scenarios.

To bridge the representation gap for low-resource languages, vocabulary expansion has emerged as a primary adaptation strategy (Hangya

et al., 2022; Lu et al., 2025). Current engineering practices, however, typically rely on heuristic injections of new tokens. For instance, Zhuang et al. (2024) introduced approximately 32,000 Tibetan tokens to Llama 2-7B, whereas Zhang et al. (2024) opted for a more conservative 3,000 tokens per language across four minority languages, Mongolian, Tibetan, and Uyghur. Such practices exhibit significant limitations, as expansion scales are often determined by intuition rather than a systematic analysis of the trade-offs between performance gains and inference overhead (Nozaki et al., 2025). Furthermore, prior research has focused almost exclusively on monolingual adaptation, leaving the co-optimization of vocabulary in multilingual joint expansion scenarios—essential for regional-level application models—inadequately studied.

The challenges are further compounded by the limitations of mainstream BBPE implementations. While BBPE ensures universality via byte-level decomposition (Kudo and Richardson, 2018), the lack of language-specific optimization in mainstream tokenizers—combined with inherent morphological complexity—frequently shatters words into semantically opaque fragments in languages like Mongolian, Tibetan, and Uyghur (Rust et al., 2021). This phenomenon leads to dual challenges: it substantially inflates sequence lengths, thereby elevating inference latency and potentially inducing generation instability due to long-range dependencies (Rust et al., 2021); and as highlighted by Hangya et al. (2022), it risks disrupting pre-trained byte-sequence representations when data is insufficient. Ill-chosen expansion scales can thus trigger unintended performance degradation even falling below the pre-expansion baseline. Consequently, identifying an optimal scale that balances sequence compression with vocabulary sparsity is essential to navigating the efficiency-performance trade-off in low-resource LLM adaptation.

3 Analytical Framework for Vocabulary Scaling

To quantify the non-linear impact of vocabulary scale on low-resource LLMs, we construct a controlled evaluation framework spanning multi-scale expansion, multi-architecture validation, and Pareto-optimal decision-making.

3.1 Multi-scale Vocabulary Expansion

We propose a trilingual joint expansion strategy for Mongolian (mn), Tibetan (bo), and Uyghur (ug). To rectify over-segmentation, we utilize `SentencePiece` to extract high-frequency morphemes from the Mongolian, Tibetan, and Uyghur data within the MC^2 dataset (Zhang et al., 2024). These units are prioritized during tokenization via the *longest-match principle*.

Vocabulary Scales. We establish ten vocabulary scales ($L1$–$L10$) spanning four orders of magnitude (Table 1) for the complete pre-training and evaluation pipeline. This range covers the spectrum from character-level representations ($L1$) to a practical expansion ceiling ($L10$). To further probe the intrinsic scaling limits of BBPE tokenization—a dimension less explored compared to established studies on BPE—we extend the analysis to ultra-large vocabularies of 300,000 and 1,200,000 tokens (allocating 100,000 and 400,000 per language, respectively). These extreme scales are utilized *exclusively* for tokenization quality assessment (e.g., in Figure 3a) rather than model training.

Notably, the vocabulary budget is equally partitioned among the three target languages (i.e., $V_{bo} = V_{mn} = V_{ug}$) to maintain cross-linguistic capacity parity for most levels, with two exceptions: the minimal expansion scale ($L1$) and the theoretically predicted optimal size ($L5$) derived from Tao et al. (2024). Table 1 lists the targeted expansion scales; actual increments are adjusted for token overlap during the union with the base vocabulary to maintain target scale precision (details in Appendix A.1).

Level	Expected Expansion
L1	140
L2	9,000
L5	37,689
L7	76,500
L10	195,000

Table 1: Key expansion levels. See Appendix A.1 for full L1–L10 and per-language counts.

Distribution-Aware Initialization. To stabilize new embeddings, we compute the mean μ and variance σ^2 of the base embeddings E_{base}. New tokens are sampled from $\mathcal{N}(\mu, \sigma^2)$, ensuring geometric consistency with the pre-trained manifold (see Appendix A.2 for derivation).

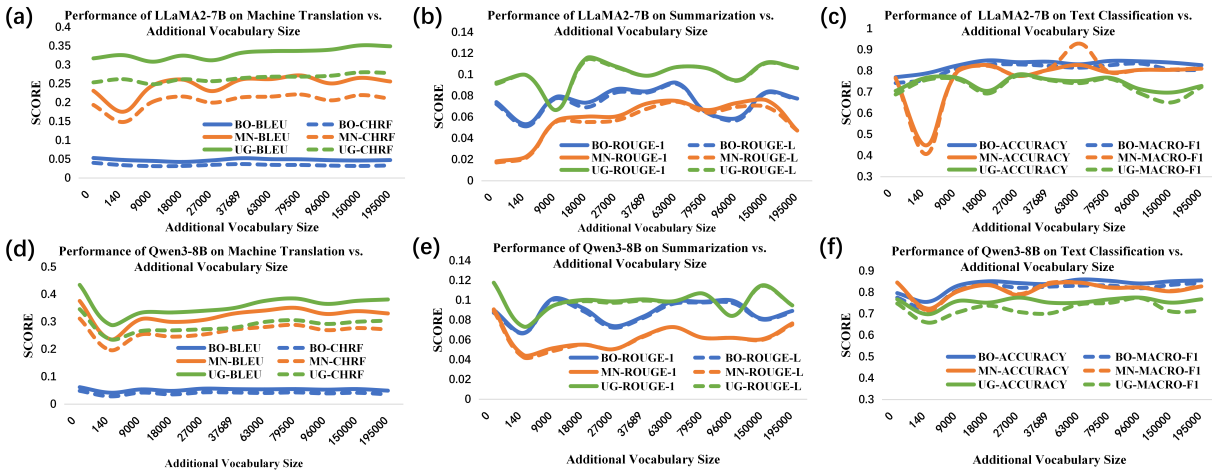


Figure 2: Performance scaling comparison. (a-c) Llama 2-7B (BPE) shows monotonic improvement. (d-f) Qwen3-8B (BBPE) exhibits a "decline-then-rise" threshold effect. Qwen2.5 results are in Appendix B.

3.2 Model Architectures

To establish the generalizability of our findings across tokenization paradigms, we evaluate three representative models:

- LLaMA-2 (7B): A classic BPE-based baseline used to observe standard scaling behavior.
- Qwen3 (8B): The primary BBPE-based model for quantifying threshold effects.
- Qwen2.5 (1.5B & 7B): Included to verify the robustness of BBPE scaling properties across different model iterations and parameter scales.

3.3 Experimental Protocol

Training. Training is conducted on the Mongolian, Tibetan, and Uyghur corpora from the MC^2 dataset (Zhang et al., 2024). We employ Parameter-Efficient Incremental Pre-training (PEFT-IP) followed by Supervised Fine-tuning (SFT). Both stages utilize Low-Rank Adaptation (LoRA) (Hu et al., 2022) to adapt the backbone while unfreezing the embedding layers to accommodate the expanded vocabulary. All training and SFT hyperparameters are summarized in Appendix A.3.

Benchmarks. Evaluation covers four dimensions: Machine Translation (CCMT 2019)(Yang et al., 2019), Text Classification (MITC)(Deng et al., 2023), Summarization (MIMO)(WENG et al., 2024), and Word Segmentation (MLWS 2021)(ZHAO, 2022). Specific metrics, dataset statistics, and the inference configurations employed to ensure experimental reproducibility are provided in Appendix A.4.

3.4 Multi-Objective Pareto Framework

To reconcile the competing demands of task performance and computational efficiency, we formulate vocabulary selection as a Multi-Objective Optimization (MOO) problem (Branke et al., 2004), providing a mathematical basis for identifying the Pareto-optimal configuration. We define four meta-objectives categorized into two primary dimensions—*Utility* and *Cost*—to balance the trade-offs: (1) **Quality Utility** (U_Q), the normalized average performance across tasks; (2) **Fairness Index** (I_F), the performance of the worst-performing language (Rawlsian Maximin Principle); (3) **Efficiency Cost** (C_E), the harmonic mean of normalized latency and parameter growth; and (4) **Resource Cost** (C_R), the normalized incremental training time. We identify the *knee point* (Satopaa et al., 2011) where marginal gains are maximized using the Kneedle algorithm. The hierarchical normalization of scores, the formulation of the balanced objective $B(v)$, and the robustness metrics (Ranking Stability and Jaccard Sensitivity) are detailed in Appendix A.5.

4 Experimental Results

4.1 Non-linear Impact of Vocabulary Scale

Figure 2 illustrates the performance trajectories of representative BPE (Llama 2-7B) and BBPE (Qwen3-8B) models across three diverse downstream tasks: Machine Translation (MT), Text Summarization (TS), and Text Classification (TC). Our findings reveal that the vocabulary-performance relationship is paradigm-specific, fundamentally governed by the underlying tokenization architec-

ture.

Monotonicity in BPE: Llama 2-7B As shown in Figure 2(a-c), standard BPE exhibits a generally monotonic improvement trajectory.Performance rises steadily until saturating at approximately 27,000 tokens (9,000 per language), largely aligning with the theoretical predictions of Tao et al. (2024) derived from high-resource settings. Beyond this, further expansion yields diminishing marginal utility, confirming that moderate expansion effectively mitigates over-segmentation in the BPE paradigm.

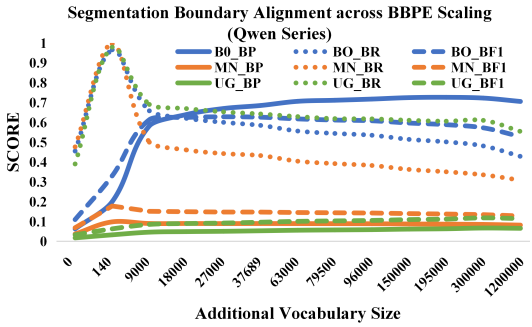
The BBPE Threshold Effect: Qwen3 and Qwen2.5 In stark contrast to the monotonic trajectory of BPE, BBPE-based models exhibit a distinct "decline-then-rise" U-shaped pattern. Crucially, cross-model validation (detailed in Appendix B.1) confirms that this inflection point remains consistently anchored at the 9,000-token threshold (3,000 per language), regardless of both model scale (1.5B vs. 7B) and architectural iteration (covering both the pre-iteration Qwen2.5 and the evolved Qwen3 architectures). This non-linear behavior can be divided into three stages:

(1) **Performance Degradation Regime ($<$ 9,000 tokens)**: Performance initially degrades below the baseline. This confirms our hypothesis regarding *representation space disruption*: when the number of new tokens is insufficient, the model fails to bridge the gap between fragmented byte-level units and stable linguistic morphemes.

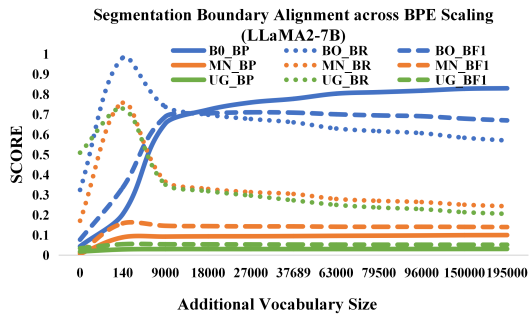
(2) **The Inflection Point ($\sim$ 9,000 tokens)**: Surpassing this critical threshold allows performance to finally recover to baseline levels and even exhibit further improvements, marking the point where the negative impacts of over-segmentation are effectively neutralized.

(3) **Diminishing Returns and Backlash**: While gains continue, they exhibit diminishing marginal utility. Critically, for parameter-constrained models like Qwen2.5-1.5B, we observe a *Scaling penalty* beyond 150,000 tokens. In this regime, performance collapses as the disproportionate Embedding Parameter Proportion "Constrains" the core Transformer capacity, indicating that the upper bound for BBPE expansion is strictly constrained by total model parameters (see Appendix B.1).

Tokenization Quality Analysis MLWS analysis (Figure 3) shows BBPE (a) exhibits high boundary volatility in the Performance Degradation Regime ($<$9,000 tokens), whereas BPE (b) stabilizes earlier. We further extended the BBPE analysis to ultra-large scales (300,000 and 1,200,000 tokens) *exclusively* for tokenization quality to probe its theoretical ceiling. Figure 3a reveals that marginal gains in segmentation precision plateau beyond 150,000 tokens, confirming that BBPE requires a significantly larger vocabulary than BPE to achieve morphological stability.



(a) Qwen Series (BBPE)



(b) LLama 2-7B (BPE)

Figure 3: MLWS Boundary Analysis: (a) BBPE shows high instability in the Performance Degradation Regime ($<$ 9,000 tokens); (b) BPE shows stable alignment with linguistic boundaries.

4.2 Efficiency Trade-offs: Compression, Parameters, and Costs

This section explores the synergistic impact of vocabulary expansion on tokenization efficiency, physical model attributes, and core computational costs.

4.2.1 Tokenization Compression Ratio (CR)

As illustrated in Figure 4, the compression ratio (CR) exhibits a distinct non-linear evolution. During the initial expansion phase (0 to 9,000 tokens), the CR rises steeply, particularly for Tibetan (BO). This stage marks the transition of BBPE from processing *semantically impoverished byte-sequences* to *language-specific subwords*. At approximately 3,000 tokens per language (the 9k threshold), the

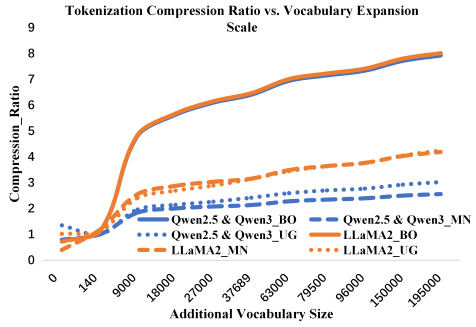


Figure 4: Tokenization Compression Ratio (CR) vs. Vocabulary Scale for Low-Resource Languages.

tokenizer captures the most frequent morphemes, yielding the most dramatic reduction in sequence length. Beyond this *inflection point*, the marginal utility of further sequence shortening diminishes, entering an asymptotic saturation phase around 150,000 tokens. This suggests a physical limit to how much a discrete vocabulary can compress natural language.

4.2.2 Parameter Inflation and Training Duration

While a larger vocabulary reduces sequence length, it induces linear parameter inflation concentrated in the embedding and output layers. For Qwen3-8B, expansion to 195,000 tokens results in a 19.5% parameter increase (from 8.19B to 9.79B). In smaller models like Qwen2.5-1.5B, this increase reaches 33.7%, creating a "top-heavy" architecture that diminishes the capacity available for Transformer backbone.

Interplay between compression (gains) and inflation (costs) creates two distinct phases (see summary in Table 2; full results in AppendixB.3):

Model	Metric	Base(0)	9k	79.5k	195k
Llama 2-7B	T (h)	82.98	20.17	16.04	15.46
	P (B)	6.74	6.81	7.39	8.33
Qwen3-8B	T (h)	71.51	24.35	19.61	19.20
	P (B)	8.19	8.26	8.84	9.79
Qwen2.5-7B	T (h)	57.05	19.69	16.27	16.61
	P (B)	7.62	7.68	8.18	9.01
Qwen2.5-1.5B	T (h)	19.93	6.90	5.64	**9.39**
	P (B)	1.78	1.80	2.02	2.38

Table 2: Impact of vocabulary expansion on training duration (T, hours) and parameter counts (P, billions).

(1) **Initial Dividend Phase ($V < 9k$):** While both models experience a precipitous drop in training time (e.g., Llama 2-7B from 82.98h to 20.17h), an *efficiency-performance paradox* emerges specifically for BBPE-based architectures. In this phase, the substantial efficiency gains from sequence shortening coincide with the *Performance Degradation Regime* (Section 4.1), where performance degrades due to representation instability—a phenomenon notably absent in the monotonic improvement of BPE-based models.

(2) **Saturation and Reversal Phase ($V > 9k$):** Training time stabilizes as Softmax and gradient overhead begin to offset compression gains. Critically, Qwen2.5-1.5B exhibits a severe *efficiency reversal* at $195k$ tokens, with training time surging to 9.39h. This confirms that the computational burden of an oversized vocabulary eventually outweighs any benefits of sequence reduction (see Appendix B.2).

4.2.3 Inference Efficiency and the Paradox

In monolingual understanding tasks (e.g., Text Classification), inference speed remains synchronized with the CR. However, generative tasks like Machine Translation (MT) to Chinese reveal a counter-intuitive paradox: despite significantly compressing the source sequences, MT throughput eventually trends downward (Figure 5).

This stems from a fundamental trade-off: although an expansive vocabulary optimizes source-side encoding, the escalating computational cost of matrix multiplications in the oversized output Softmax layer—which projects into the full trilingual space even when generating Chinese—eventually outweighs the gains. At 150,000 tokens, the overhead of the expanded output projection dominates the generation cycle, identifying an efficiency ceiling where the computational burden of the vocabulary size compromises the overall speed-performance balance.

4.3 Validation of Pareto Optimality and Robustness

To identify the optimal balance between performance (U_Q), equity (I_F), efficiency (C_E), and resource cost (C_R), we filter configurations through a Pareto framework. We focus on Qwen3-8B as the primary BBPE representative, with results for other models in Appendix C.

Knee Point and Cross-Model Generalizability The Pareto *knee point* analysis reveals a significant evolution in vocabulary efficiency. For our primary model, Qwen3-8B, the optimal balance

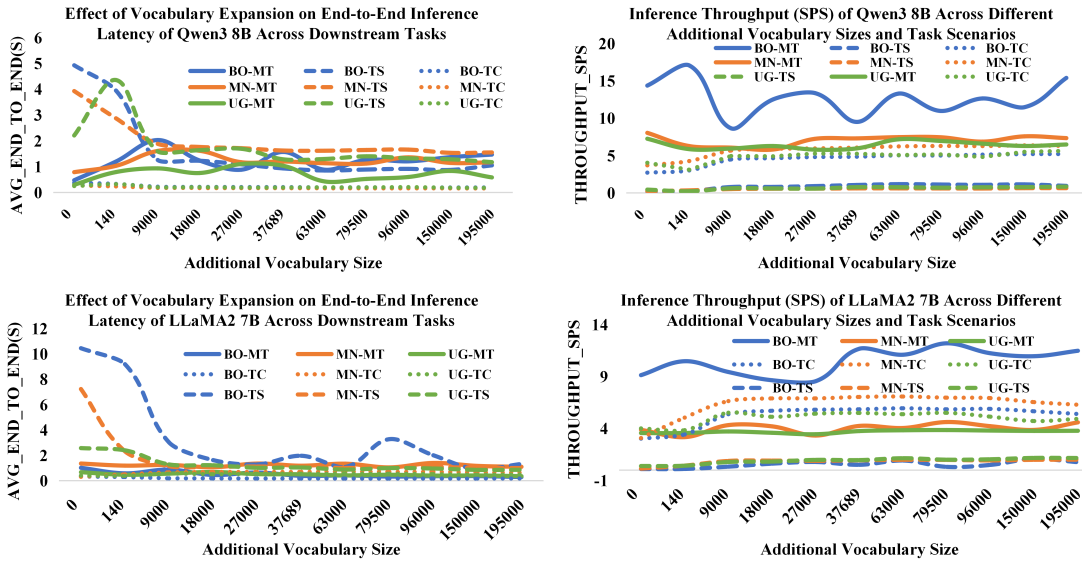


Figure 5: Inference Throughput vs. Vocabulary Scale: Efficiency Paradox in Generative Tasks.

Strategy	MT (BLEU)			Classification (Acc.)			Summarization (R-L)		
	BO	MN	UG	BO	MN	UG	BO	MN	UG
Qwen3-8B (JTE)	**0.0548**	**0.3389**	**0.3765**	0.851	0.805	0.752	**0.081**	0.061	**0.115**
Qwen3-8B (IME)	0.0532	0.2793	0.3585	**0.862**	**0.839**	**0.760**	0.076	**0.064**	0.111
Relative Gain (%)	+3.0	+21.3	+5.0	-1.3	-4.1	-1.1	+6.6	-4.7	+3.6

Table 3: JTE vs. IME at 150k. JTE excels in generation; IME leads marginally in understanding.

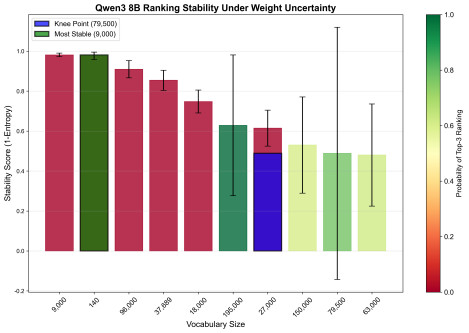


Figure 6: Ranking stability under weight uncertainty.

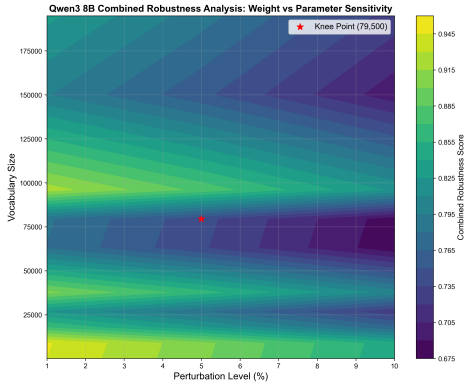


Figure 7: Robustness heatmap across noise levels.

is identified at 79,500 tokens. Crucially, due to space constraints, results for the BPE-based Llama 2-7B baseline and the generalizability analysis across the Qwen2.5 series (1.5B and 7B) are deferred to Appendix C. the 79.5k configuration also exhibits remarkable efficacy on the previous-generation Qwen2.5-7B. While Qwen2.5-7B requires 195,000 tokens to reach its theoretical performance peak, the 79.5k scale already captures over 92% of the maximum potential gains in downstream tasks while saving nearly 830 million parameters compared to the 195k version (Table 2). This demonstrates that 79.5k serves as a robust, cross-generational optimal configuration: it provides a highly competitive performance-to-cost ratio for modern architectures while offering a "Reliable baseline" configuration for older ones that prevents excessive parameter bloat.

Ranking Stability and Engineering Robustness To ensure the 79,500-token recommendation is not biased by specific evaluation metrics, we simulated 2,000 random objective weight combinations (Fig. 6). While the 9,000-token scale shows the highest absolute stability, it remains a low-

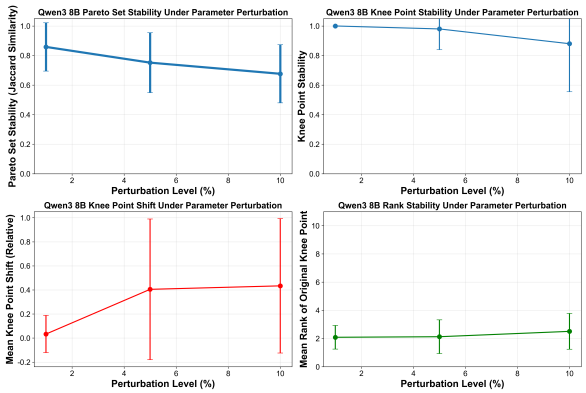


Figure 8: Jaccard sensitivity to parameter noise.

performance baseline. The 79,500-token configuration maintains a high Top-3 ranking probability under weight perturbations across different models. This confirms its status as a robust engineering choice that consistently outperforms both minimal expansions (which suffer from representation disruption) and ultra-large expansions (which suffer from diminishing marginal utility).

Robustness and Engineering Recommendation Robustness experiments (Fig. 7, 8) reveal that the 79.5k configuration resides in a stable "plateau zone" across architectures, whereas expanding toward the 195k peak induces significant *ranking volatility* as the oversized embedding layer becomes hypersensitive to sparse updates. This mid-range scale effectively mitigates the *representation fragility* observed in smaller models (e.g., Qwen2.5-1.5B). Synthesizing the Pareto validation and robustness analysis, we establish 79,500 tokens as the universal engineering guideline for trilingual BBPE expansion, providing an ideal equilibrium between task performance, computational efficiency, and representational stability.

5 Comparative Analysis: Joint vs. Independent Monolingual Expansion

To evaluate the efficacy of our joint expansion strategy, we compare *Joint Trilingual Expansion* (JTE) against an *Independent Monolingual Expansion* (IME) baseline at the 150k scale. To ensure a controlled comparison, the IME sub-vocabularies (50k tokens each for bo, mn, and ug) are directly extracted as subsets from the JTE 150k trilingual pool. Three separate models are then trained under the IME setting to isolate the impact of the shared embedding space. We benchmark these strategies across three key metrics: Machine Translation (MT), Text Classification (TC), and Summarization (TS) (see Table 3).

As illustrated in Table 3, JTE significantly outperforms IME in generative tasks, particularly in Machine Translation where Mongolian (MN) achieves a +21.3% relative gain. This evidence suggests that a *shared trilingual embedding space* acts as a "semantic bridge," facilitating cross-lingual semantic alignment and knowledge transfer. By mapping these morphologically distinct languages into a unified manifold, JTE allows the model to leverage common structural features and high-frequency subword units, effectively mitigating data-sparsity issues that often plague independent adaptation.

Conversely, IME maintains a marginal advantage in text classification (+1.1% to +4.1%). This is likely due to the higher *monolingual representation density* and the absence of cross-lingual token interference within the embedding space, allowing the model to focus exclusively on single-language discriminative features. However, the generative paradox remains: JTE's substantial gains in sequence generation, coupled with its drastically lower deployment overhead (maintaining a single model instead of three), confirm that joint expansion is a more scalable and efficient framework for regional multilingual LLM adaptation.

6 Conclusion

This yields three primary insights into vocabulary scaling for low-resource languages. First, we identify a critical "Performance Degradation Regime" below 3,000 tokens per language, where the disruption of byte-level representations outweighs sequence compression gains, leading to performance degradation. Second, Pareto Frontier Analysis pinpoints optimal knee points at 79,500 tokens for BBPE(Qwen3-8B) and 63,000 tokens for BPE(Llama 2-7B), which improve downstream performance and inference speed while balancing associated parameter growth. Third, we demonstrate that scaling is tightly constrained by model capacity; in smaller architectures, an excessive vocabulary triggers a performance decline as a high *Embedding-to-Core Ratio* reallocates the parameter budget away from core Transformer layers. This empirical study provides quantitative evidence for balancing performance, efficiency, and architectural stability in low-resource LLM adaptation, navigating the fundamental trade-off between parameter distribution and inference throughput.

7 Limitations

This study has several limitations. First, the inherent scarcity of low-resource corpora precludes training and validation of our findings on ultra-large architectures (e.g., 70B+), which require substantially higher data volumes to converge; thus, our experiments are constrained to the 1.5B–8B parameter range. Second, standard statistical tokenizers (BPE/BBPE) may not fully capture the complex morphology of agglutinative languages compared to specialized morphological parsers. Finally, the impact of vocabulary expansion on the models' original high-resource capabilities (e.g., original English or Chinese) remains to be extensively assessed to ensure no performance regression in base knowledge.

References

Orevaoghene Ahia, Sachin Kumar, Hila Gonen, Jungo Kasai, David Mortensen, Noah Smith, and Yulia Tsvetkov. 2023. Do all languages cost the same? tokenization in the era of commercial language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9904–9923, Singapore. Association for Computational Linguistics.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Jürgen Branke, Kalyanmoy Deb, Henning Dierolf, and Matthias Osswald. 2004. Finding knees in multi-objective optimization. In *International conference on parallel problem solving from nature*, pages 722–731. Springer.

Junjie Deng, Hanru Shi, Xinhe Yu, Wugedele Bao, Yuan Sun, and Xiaobing Zhao. 2023. Milmo:minority multilingual pre-trained language model. In *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 329–334.

Viktor Hangya, Hossain Shaikh Saadi, and Alexander Fraser. 2022. Improving low-resource languages in pre-trained multilingual language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11993–12006, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, and 3 others. 2022. Training compute-optimal large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *Preprint*, arXiv:2001.08361.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Tomasz Limisiewicz, Jiří Balhar, and David Mareček. 2023. Tokenization impacts multilingual language modeling: Assessing vocabulary allocation and overlap across languages. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5661–5681, Toronto, Canada. Association for Computational Linguistics.

Kaiwen Lu, Yating Yang, Fengyi Yang, Rui Dong, Bo Ma, Aihetamujiang Aihemaiti, Abibilla Atawulla, Lei Wang, and Xi Zhou. 2025. Low-resource language expansion and translation capacity enhancement for LLM: A study on the Uyghur. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 8360–8373, Abu Dhabi, UAE. Association for Computational Linguistics.

Yuta Nozaki, Dai Nakashima, Ryo Sato, Naoki Asaba, and Shintaro Kawamura. 2025. Efficient vocabulary reduction for small language models. In *Proceedings of the 31st International Conference on Computational Linguistics: Industry Track*, pages 771–783, Abu Dhabi, UAE. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, and 1 others. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. 2021. How good is your tokenizer? on the monolingual performance of multilingual language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3118–3135, Online. Association for Computational Linguistics.

Ville Satopaa, Jeannie Albrecht, David Irwin, and Barath Raghavan. 2011. Finding a" kneedle" in a

haystack: Detecting knee points in system behavior. In *2011 31st international conference on distributed computing systems workshops*, pages 166–171. IEEE.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 1715–1725.

Sho Takase, Ryokan Ri, Shun Kiyono, and Takuya Kato. 2025. Large vocabulary size improves large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 1015–1026, Vienna, Austria. Association for Computational Linguistics.

Chaofan Tao, Qian Liu, Longxu Dou, Niklas Muennighoff, Zhongwei Wan, Ping Luo, Min Lin, and Ngai Wong. 2024. Scaling laws with vocabulary: Larger models deserve larger vocabularies. *arXiv preprint arXiv:2407.13623*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Zihan Wang, Karthikeyan K, Stephen Mayhew, and Dan Roth. 2020. Extending multilingual BERT to low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2649–2656, Online. Association for Computational Linguistics.

Yu WENG, Tianjiao XING, Xuming YE, Zheng Liu, Rilige CHAOMU, and Xuan LIU. 2024. A dataset of chinese-mongolian-tibetan-uyghur multi-document summaries. *China Scientific Data*, 9:1–12.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Muyun Yang, Xixin Hu, Hao Xiong, Jiayi Wang, Yiliyaer Jiaermuhamaiti, Zhongjun He, Weihua Luo, and Shujian Huang. 2019. Ccmt 2019 machine translation evaluation report.

Chen Zhang, Mingxu Tao, Quzhe Huang, Jiuheng Lin, Zhibin Chen, and Yansong Feng. 2024. MC^2: Towards transparent and culturally-aware NLP for minority languages in China. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8832–8850, Bangkok, Thailand. Association for Computational Linguistics.

Xiaobing ZHAO. 2022. A dataset of word segmentation technology evaluation for minority languages (mlws2021). *China Scientific Data*, 7.

Wenhao Zhuang, Yuan Sun, and Xiaobing Zhao. 2024. (TiLamb: A Tibetan large language model based on incremental pre-training). In *Proceedings of the 23rd Chinese National Conference on Computational Linguistics (Volume 1: Main Conference)*, pages 254–267, Taiyuan, China. Chinese Information Processing Society of China.

A Implementation Details

A.1 Detailed Vocabulary Expansion levels

To systematically investigate the scaling effect, we design ten expansion levels (L_1–L_{10}). As shown in Table 4, L_1 represents the minimal vocabulary size extracted by `SentencePiece` to ensure full coverage of unique characters and basic graphemes. Our levels design combines key theoretical anchor points (e.g., L_5, L_8) with equidistant linear interpolation points. Specifically, L_5 is derived from theoretical scaling laws that correlate optimal vocabulary size with model parameters and training data volume (Tao et al., 2024). To facilitate a consistent horizontal comparison across models of varying scales, we utilize Qwen2.5-7B as the benchmark architecture to determine the L_5 value and apply this fixed target size uniformly to LLaMA-2, Qwen3, and Qwen2.5-1.5B.

The Expected column indicates the target increments produced by the `SentencePiece` tokenizer. In contrast, the Actual columns show the final count after a *union* operation with the model's base vocabulary. Notably, since Qwen2.5 and Qwen3 share identical base representations and character sets for Mongolian, Tibetan, and Uyghur, the resulting trilingual token increments are identical for both architectures; thus, they are consolidated under a unified Qwen column. Minor discrepancies between the expected and actual values arise from tokens already present in the original models, such as common UTF-8 byte sequences or shared symbols. Finally, L_{10} serves as the prescribed upper bound of our experimental expansion range.

A.2 Tokenizer Training and Initialization

Language-specific subword models are trained using `SentencePiece` (BPE) with `character_coverage=0.995` and `byte_fallback=True`. To eliminate interference from non-target scripts during frequency estimation, we implement a rigorous filtering strategy before training. Specifically, we apply the regex rule `re.sub(r'[a-zA-Z0-9]', '', line)` to exclude all Latin letters and Arabic numerals. This procedure ensures that the BPE algorithm prioritizes the morphological characteristics

	Per-Language Count				Actual (Union)		
Level	**bo**	**mn**	**ug**	**Expected**	**Qwen**	**Llama**	**Description**
L1	66	34	40	140	124	78	Minimal character-level set (`SentencePiece`)
L2	3,000	3,000	3,000	9,000	8,978	8,920	Linear interpolation point
L3	6,000	6,000	6,000	18,000	17,975	17,915	Linear interpolation point
L4	9,000	9,000	9,000	27,000	26,972	26,913	Linear interpolation point
L5	11,996	12,101	13,592	37,689	37,657	37,596	Theoretically optimal size (Tao et al., 2024)
L6	21,000	21,000	21,000	63,000	62,963	62,900	Linear interpolation point
L7	26,500	26,500	26,500	79,500	79,462	79,399	Linear interpolation point
L8	32,000	32,000	32,000	96,000	95,962	95,899	TiLamb benchmark scale (Zhuang et al., 2024)
L9	50,000	50,000	50,000	150,000	149,958	149,892	Linear interpolation
L10	65,000	65,000	65,000	195,000	194,953	194,888	Upper bound of experimental scaling

Table 4: Detailed configuration of vocabulary expansion levels. The "bo", "mn", and "ug" columns represent counts for Tibetan, Mongolian, and Uyghur. "Expected" denotes the raw output from `SentencePiece`, while "Actual" reflects the unique tokens after merging with base vocabularies.

of low-resource languages and prevents alphanumeric characters from being over-segmented into fragmented tokens.

To mitigate representational disruption, we use Distribution-Aware Initialization. Let $\mathbf{E}_{\text{base}} \in \mathbb{R}^{V_{\text{base}} \times d}$ be the base embedding matrix. We compute the dimension-wise mean μ_j and standard deviation σ_j:

$$\mu_j = \frac{1}{V_{\text{base}}} \sum_{i=1}^{V_{\text{base}}} \mathbf{E}_{i,j}, \qquad \sigma_j = \sqrt{\frac{1}{V_{\text{base}}} \sum_{i=1}^{V_{\text{base}}} (\mathbf{E}_{i,j} - \mu_j)^2}. \tag{1}$$

New embeddings $\mathbf{e}_{\text{new}}$ are sampled from $\mathcal{N}(\mu_j, \sigma_j^2)$, ensuring the new tokens reside on the pre-trained manifold.

A.3 Training Configurations and Hardware

The hyperparameters for pre-training (PT) and supervised fine-tuning (SFT) are detailed in Table 5, while the hardware platforms used for different model architectures are listed in Table 6.

Table 5: Hyperparameters for PT and SFT stages.

Parameter	**Pre-training (PT)**	**Fine-tuning (SFT)**
Optimizer	AdamW	AdamW
Learning Rate	1e-4	2e-5
LR Scheduler	Cosine	Cosine
Trainable	LoRA, Emb., Head	
LoRA Targets	All Linear	
LoRA Rank (r)	128 (64 for 1.5B)	64
LoRA Alpha (α)	256 (128 for 1.5B)	128
Epochs	3.0	2.0
Batch Size	48 (Effective)	64 (Effective)
Max Length	2048	512 / 1516 (MT)

Table 6: Hardware environment per architecture.

Model Architecture	**Hardware Platform**
Qwen3-8B / Qwen2.5-7B	Huawei Ascend 910B4 NPU
Llama 2-7B	Huawei Ascend 910B4 NPU
Qwen2.5-1.5B	NVIDIA A800 GPU
Downstream Evaluation	NVIDIA A800

A.4 Evaluation Metrics and Rationale

To comprehensively evaluate the impact of vocabulary scaling on both computational efficiency and task-specific quality, we establish a multi-dimensional metric system. This system captures the non-linear trade-offs between sequence compression and representational stability.

For efficiency benchmarking, inference is conducted using the vLLM framework on NVIDIA A800 GPUs. To ensure consistent and reproducible results, we set the GPU memory utilization to 0.9, use a fixed random seed of 42, and configure the generation with temperature set to 0 and Top-k sampling disabled.Notably, to accommodate the extensive scale of translation test sets and optimize evaluation efficiency, we set the batch size to 4 for translation tasks to maximize VRAM utilization and accelerate throughput. Each experiment is repeated three times, and we report the average values of latency and throughput. Table 7 provides a detailed summary of these metrics and their linguistic rationales.

A.5 Pareto Optimization and Robustness Analysis Details

To identify the optimal vocabulary size V^* through the Multi-Objective Optimization (MOO) framework described in Section 3.3, we implement a

Category	Metrics	Description and Linguistic Rationale
Efficiency	Latency (E2E)	Total time for the full pipeline, including tokenization, model forward pass, and detokenization. Averaged over three runs to ensure stability.
	Throughput	Measured in Tokens Per Second (TPS) and Samples Per Second (SPS) to quantify generation speed and processing capacity under the vLLM framework.
	CR	Compression Ratio (CR): Ratio of raw UTF-8 bytes to generated tokens. Reflects the tokenizer's ability to condense morphologically complex text.
Task-Specific	Machine Translation	BLEU / chrF: Standard translation accuracy. chrF is prioritized as it accounts for character n-gram matches, making it more robust for agglutinative languages.
	Text Summarization	ROUGE-1/L / chrF: Measures information coverage and semantic density. ROUGE-L captures longest common subsequences to evaluate fluency.
	Text Classification	Accuracy / Macro-F1: Evaluates the model's discriminative performance in a generative setting, ensuring fairness across imbalanced low-resource classes.
	Tokenizer Quality	Boundary F1 / IV Rate: Specifically used to diagnose the "Performance Degradation Regime." Boundary F1 measures segmentation precision, while IV Rate tracks vocabulary coverage.

Table 7: Detailed summary of evaluation metrics for efficiency and downstream tasks, including their specific linguistic rationales for Mongolian, Tibetan, and Uyghur.

three-stage mathematical pipeline: normalization, knee point identification, and stability validation.

Hierarchical Normalization To ensure *Scale Invariance* across diverse metrics (e.g., BLEU, Accuracy) and languages, we first apply min-max normalization to raw scores $S_{l,t}$ for each language l and task t, followed by language-level aggregation P_l:

$$\begin{aligned} \bar{S}_{l,t} &= \frac{S_{l,t} - \min_{v \in V}(S_{l,t})}{\max_{v \in V}(S_{l,t}) - \min_{v \in V}(S_{l,t}) + \epsilon}, \\ P_l &= \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \bar{S}_{l,t}, \end{aligned} \tag{2}$$

where $\epsilon = 10^{-8}$ is added for numerical stability.

Knee Point Identification (Kneedle) We define a balanced score $B(v)$ by weighting the aggregated quality utility and fairness against computational and resource costs:

$$B(v) = \frac{U_Q(v) + I_F(v)}{2} - \alpha \cdot \frac{C_E(v) + C_R(v)}{2}, \tag{3}$$

where $\alpha = 0.5$ balances the performance–efficiency trade-off. Following the Kneedle Algorithm (Satopaa et al., 2011), we map both the vocabulary scales v and scores $B(v)$ to the unit interval $[0, 1]$, resulting in normalized coordinates $(\tilde{v}, \tilde{B}(\tilde{v}))$. The optimal knee point V^* is the scale that maximizes the difference curve $D(\tilde{v})$ before the onset of diminishing marginal utility:

$$D(\tilde{v}) = \tilde{B}(\tilde{v}) - \tilde{v}, \quad V^* = \arg\max_{\tilde{v}} D(\tilde{v}). \tag{4}$$

Robustness and Stability Metrics To verify the engineering reliability of the identified V^*, we employ two sensitivity indicators:

1. **Ranking Stability**: We use Shannon Entropy $H(v)$ to measure the stability of a configuration's rank under $N = 2000$ Dirichlet-sampled weight perturbations. The stability score $\mathrm{Stab}(v)$ is defined as:

$$\mathrm{Stab}(v) = 1 - \frac{-\sum_r p_r \log_2(p_r)}{\log_2(|V|)}. \tag{5}$$

2. **Pareto Set Sensitivity**: We use the Jaccard Index J to measure the overlap between the original Pareto-optimal set $\mathcal{P}_{\text{orig}}$ and the set under δ noise perturbation $\mathcal{P}_{\text{pert}}$:

$$J = \frac{|\mathcal{P}_{\text{orig}} \cap \mathcal{P}_{\text{pert}}|}{|\mathcal{P}_{\text{orig}} \cup \mathcal{P}_{\text{pert}}|}. \tag{6}$$

B Cross-Architecture Robustness (Qwen2.5 Series)

This appendix provides a comprehensive analysis of the Qwen2.5 series (1.5B and 7B) to validate the generalizability of our findings regarding the BBPE threshold effect and efficiency bottlenecks.

B.1 Performance Scaling and Scaling Penalty

As illustrated in Figure 9, the Qwen2.5 series replicates the "decline-then-rise" performance trajectory observed in Qwen3-8B.

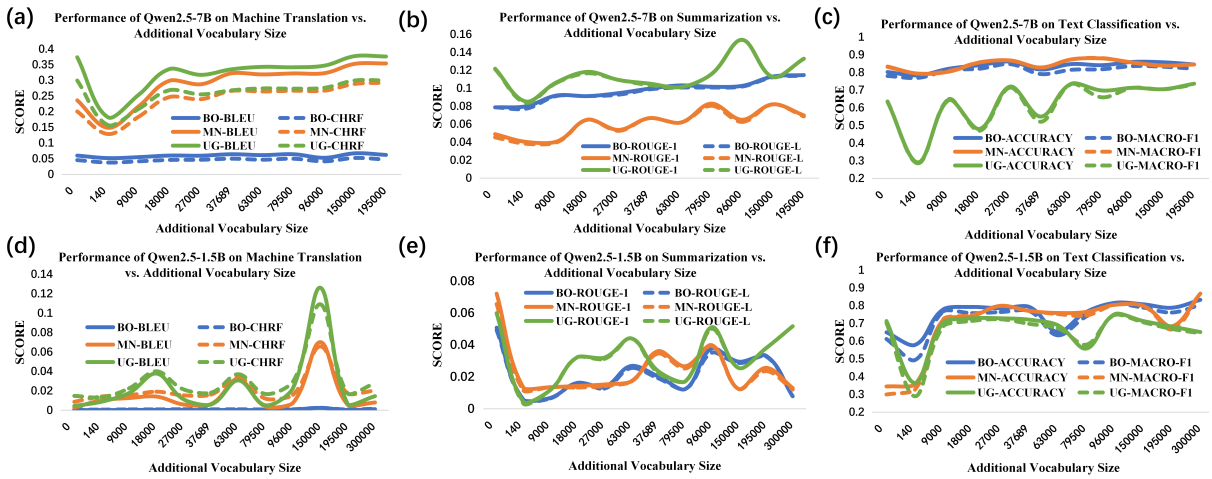


Figure 9: Performance trajectories for Qwen2.5-7B and 1.5B across MT, TS, and TC tasks, confirming the robustness of the BBPE threshold effect.

Universal Inflection Point Regardless of model size (1.5B vs. 7B), the inflection point remains anchored at the 9,000-token threshold. This confirms that the *Performance Degradation Regime* is a paradigm-level constraint inherent to BBPE, rather than a model-specific artifact.

The Backlash Theory We observe a significant performance collapse in the 1.5B model at ultra-large scales ($>$ 150k). We attribute this to the Embedding-to-Core Ratio: at 195k tokens, the embedding layer accounts for over 33% of the 1.5B model's parameters, severely "squeezing" the core Transformer capacity and leading to unstable representation updates in data-sparse scenarios.

B.2 Inference Throughput and Efficiency Reversal Analysis

The Qwen2.5 series highlights the hardware-level constraints of vocabulary scaling. Figure 10 illustrates the end-to-end inference latency and throughput (SPS) across Mongolian, Tibetan, and Uyghur downstream tasks.

Task-Specific Volatility and the 1.5B Collapse Inference throughput reveals a stark contrast between understanding (TC) and generative (MT, TS) tasks. For the Qwen2.5-1.5B model, understanding tasks exhibit a severe *efficiency backlash*: while throughput initially rises due to sequence compression, it collapses and becomes highly volatile beyond the 27k-token scale. This oscillatory pattern in SPS (Figure 10, bottom-right) confirms that for small-parameter models, the memory and compute overhead of the oversized embedding/Softmax layers creates a bottleneck that negates the benefits of shorter sequences.

Generative Paradox in Decoder-only Models For generative tasks (MT and TS), throughput remains significantly lower and less responsive to vocabulary expansion compared to classification. In the 1.5B model, MT and TS throughput remains nearly flat or trends slightly downward at ultra-large scales (>150k). This reinforces the paradox discussed in Section 4.2: although larger vocabularies reduce the number of decoding iterations, the linear increase in per-step Softmax computation cost across the full trilingual space eventually dominates the inference cycle, especially when the core Transformer capacity is restricted.

Identification of the Stable Engineering Zone In contrast to the 1.5B model's volatility, the Qwen2.5-7B architecture maintains a relatively stable throughput plateau between 27k and 79.5k tokens. Beyond 79.5k, even the 7B model begins to exhibit diminishing returns in inference speed. This empirical evidence validates our recommendation of the 79.5k configuration as a robust engineering "optimal trade-off point" that maximizes sequence compression while maintaining representational and computational stability across both model scales.

B.3 Detailed Training Efficiency and Parameter Statistics

To provide a granular view of the trade-off between sequence compression and architectural overhead, we present the full experimental results for all ten vocabulary expansion levels in Table 8.

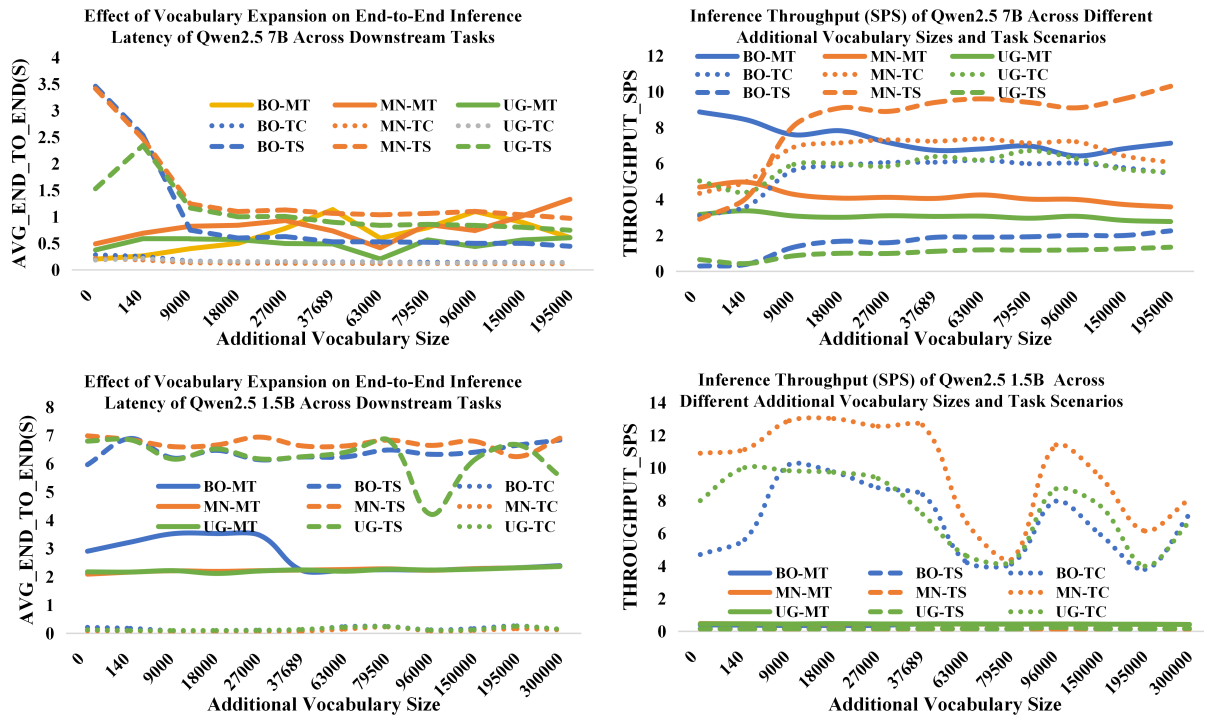


Figure 10: Inference throughput (SPS) and latency for Qwen2.5 series. The 1.5B model shows a clear reversal in efficiency at the largest vocabulary sizes.

Parameter Inflation. As the vocabulary size (V) increases, the parameters in the embedding layer and the language modeling head grow linearly. For the 7B/8B class models, expanding to 195,000 tokens adds approximately 1.5B to 1.6B parameters. While this represents a manageable increase ($\sim$19–24%) for large models, it is disproportionately high for the 1.5B model, where the parameter count nearly doubles from 1.78B to 2.38B. This data supports our "top-heavy" architecture hypothesis discussed in Section 4.2.2.

Training Duration and the Efficiency Paradox. The training duration (measured in hours) reveals two distinct phases:

- **Initial Dividend Phase:** A precipitous drop in training time is observed across all models when moving from 0 to 9,000 tokens. This is driven by the rapid increase in the Tokenization Compression Ratio (CR), which significantly reduces the total number of training tokens (and thus the number of forward/backward passes).
- **Saturation and Reversal:** Beyond 79,500 tokens, the reduction in training time plateaus as the overhead of computing a massive Softmax layer begins to offset the gains from sequence shortening. Most notably, the Qwen2.5-1.5B model exhibits a severe *efficiency reversal* at the 195,000 scale, with training time jumping from 5.58h back to 9.39h. This confirms that for smaller models, the computational cost of an oversized output layer eventually outweighs any benefits of sequence reduction.

C Pareto Frontier Distribution and Detailed Stability Analysis

C.1 BPE Path Analysis for Llama 2-7B

As a baseline grounded in the standard BPE paradigm, Llama 2-7B exhibits robust Pareto characteristics that differ fundamentally from the BBPE architecture. As illustrated in Figure 12, Figure 11 and Figure 13, its decision *knee point* consistently appears at $63,000$ tokens, a scale substantially smaller than the optimal range required by BBPE-based models. In our stability evaluation, the 140-token baseline demonstrates exceptionally high ranking consistency. Although the stability score declines at the $63,000$-token mark—owing to its position within the competitive performance-efficiency *trade-off zone*—its behavior under parameter perturbations remains relatively linear and predictable. The Jaccard similarity of the Pareto set decays gradually as perturbation

Table 8: Impact of vocabulary expansion on training duration (hours) and parameter counts (billions) across four representative LLMs. V denotes the number of newly added trilingual tokens (Mongolian, Tibetan, and Uyghur).

Vocab Size	Llama 2-7B		Qwen3-8B		Qwen2.5-7B		Qwen2.5-1.5B	
(V)	Train (h)	Params (B)	Train (h)	Params (B)	Train (h)	Params (B)	Train (h)	Params (B)
0 (Base)	82.98	6.74	71.51	8.19	57.05	7.62	19.93	1.78
140	47.00	6.74	54.81	8.19	47.37	7.62	15.82	1.78
9,000	20.17	6.81	24.35	8.26	19.69	7.68	6.90	1.80
18,000	18.57	6.89	22.45	8.34	18.86	7.74	6.38	1.83
27,000	17.84	6.96	21.60	8.41	18.32	7.81	6.13	1.86
37,689	17.12	7.05	20.91	8.50	17.48	7.88	5.92	1.89
63,000	16.26	7.25	19.93	8.70	16.74	8.06	5.69	1.97
79,500	16.04	7.39	19.61	8.84	16.27	8.18	5.64	2.02
96,000	15.96	7.52	19.35	8.97	15.69	8.30	5.59	2.07
150,000	15.58	7.97	19.15	9.42	16.19	8.69	5.58	2.24
195,000	15.46	8.33	19.20	9.79	16.61	9.01	**9.39**	2.38

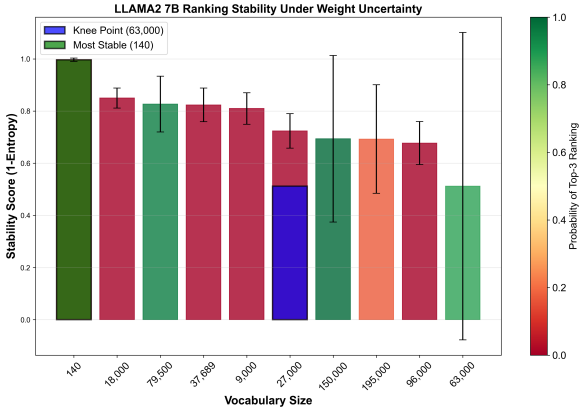


Figure 11: Rank stability of vocabulary sizes under parameter perturbations for Llama 2-7B. The 140-token baseline shows the highest consistency, while the 63,000-token knee point remains within the top two ranks despite reduced stability due to its position in the performance–efficiency trade-off zone.

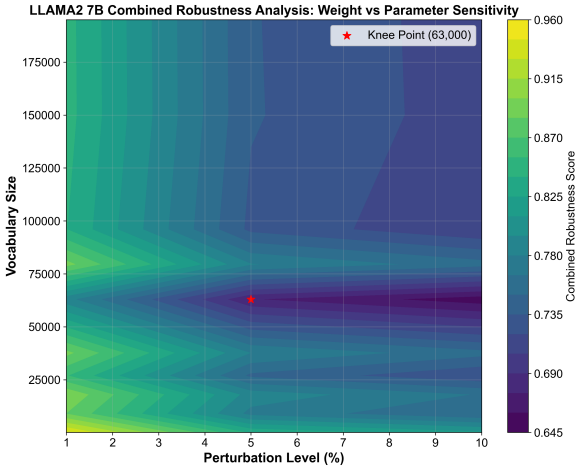


Figure 12: Pareto front of Llama 2-7B (BPE) across vocabulary sizes, showing the trade-off between downstream task performance and model efficiency. The knee point is consistently located at 63,000 tokens, indicating the optimal balance point under the standard BPE paradigm.

intensity increases, and the original knee point consistently maintains a top-two average rank under noise. This confirms that the BPE paradigm offers a more predictable decision space by monotonically mitigating the inherent over-segmentation of low-resource languages; moderate vocabulary expansion ($\approx$ 60k) can thus achieve an optimal balance without the risk of representation disruption observed in BBPE.

C.2 Robustness Analysis of Qwen2.5-7B

Representing the predecessor architecture to Qwen3-8B, Qwen2.5-7B exhibits a pronounced discrepancy in knee point identification. As shown in Figure 14,Figure 15 and Figure 16, the decision knee point for this model is identified at $195,000$ tokens. This reflects that in earlier architectures, the attention mechanism's efficiency in capturing morphological features remains heavily dependent on a larger parameter mapping space when handling ultra-large vocabularies. Regarding ranking stability, the $18,000$-token configuration (averaging 6k per language) is identified as the most stable, with a stability score ($1-\text{Entropy}$) approaching 1.0 and a high probability of maintaining a Top-3 rank across various weight distributions. The integrated robustness heatmap reveals that although $195,000$ is marked as the knee point, its stability under high-intensity parameter perturbation (10%) is slightly inferior to that of Qwen3. Specifically, the *knee point shift* exhibits a more significant upward trend as noise levels rise. This underscores that while the 7B scale provides sufficient computational capacity, the degree of architectural optimization directly

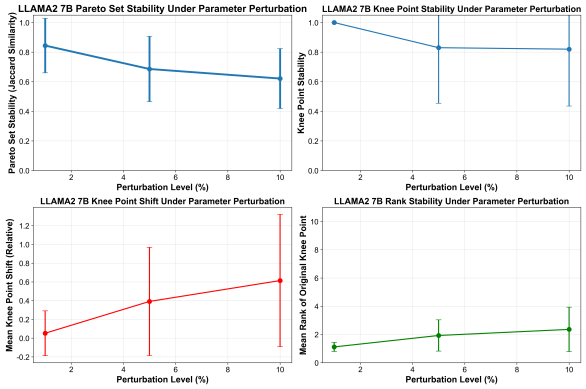


Figure 13: Jaccard similarity of the Pareto-optimal set under increasing perturbation intensity for Llama 2-7B. The gradual decay demonstrates the robustness and predictability of the BPE-based decision space, supporting the reliability of the 63,000-token knee point.

Figure 14: Pareto front for Qwen2.5-7B across vocabulary sizes, illustrating the trade-off between normalized quality/fairness and efficiency/resource costs. The knee point is located at 195,000 tokens, indicating a strong reliance on large vocabularies for performance saturation in this predecessor architecture.

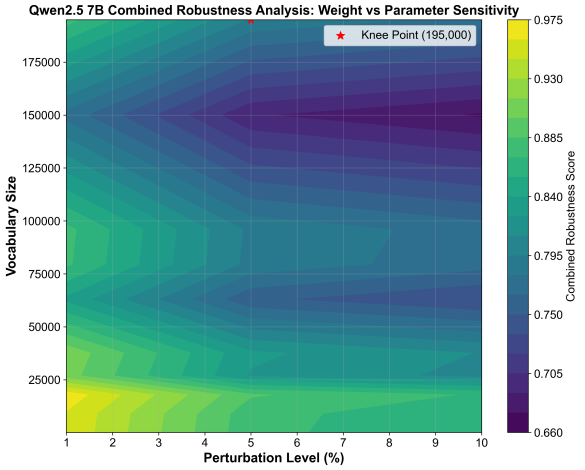


Figure 15: Ranking stability under weight uncertainty for Qwen2.5-7B. The 18,000-token configuration achieves near-perfect stability (stability score ≈ 1.0) and consistently ranks in the top three across diverse objective weightings, indicating robustness to preference variations.

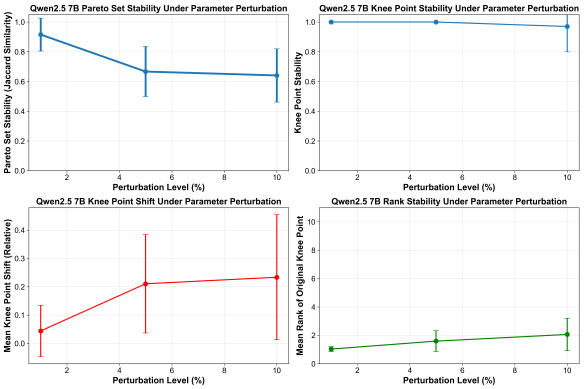


Figure 16: Parameter sensitivity and knee point robustness for Qwen2.5-7B under increasing perturbation intensity (1%–10%). The 195,000-token knee point shows greater vulnerability to noise, with a pronounced increase in knee point shift and reduced Pareto set stability compared to Qwen3-8B.

dictates the concentration and reliability of Pareto-optimal solutions.

C.3 Validation of Capacity Bottlenecks in Qwen2.5-1.5B

The experimental results for the small-scale Qwen2.5-1.5B model strongly support the "capacity constraint" hypothesis discussed in the main text. As illustrated in Figure 17,Figure 18 and Figure 19, stability scores for this model are generally low; even the 140-token baseline, identified as the "most stable" configuration, exhibits significantly higher ranking fluctuations under weight uncertainty compared to larger models. The decision knee point is situated at $150,000$ tokens. However, in our perturbation analysis, the Jaccard similarity of its Pareto optimal set plummets to below 0.4 under 10% noise, and its knee point stability score is markedly lower than that of 7B-class models. The integrated robustness heatmap further reveals that the high-robustness regions for the 1.5B model are extremely narrow and fragmented. This indicates that architectures with a disproportionately high embedding layer ratio are hypersensitive to experimental hyperparameters and weight allocations. From an engineering perspective, this instability demonstrates that forcibly adopting ultra-large vocabulary configurations in resource-constrained small models leads to a dra-

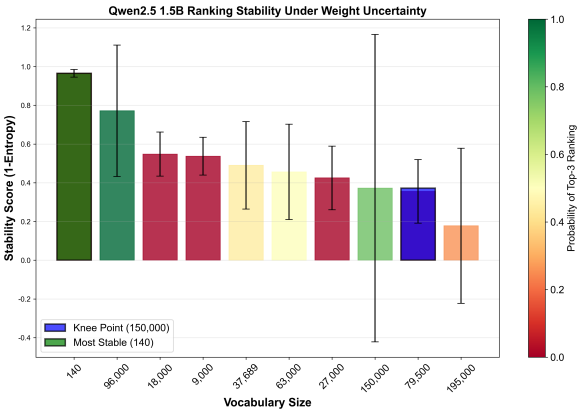


Figure 17: Pareto front of Qwen2.5-1.5B across vocabulary sizes, revealing a knee point at 150,000 tokens. Despite achieving competitive performance at large scales, the model exhibits limited capacity to stabilize trade-offs between quality, fairness, and efficiency due to its small parameter budget.

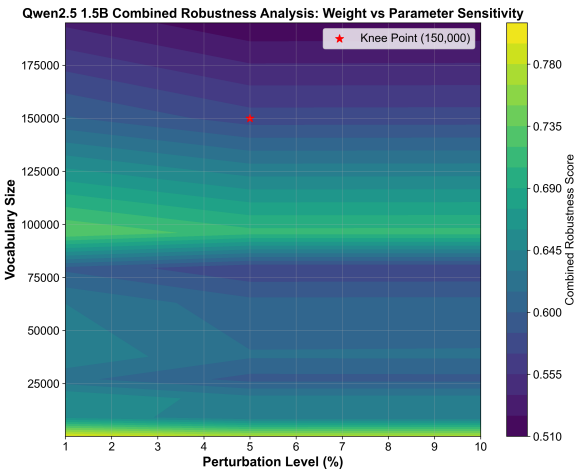


Figure 18: Ranking stability under objective weight uncertainty for Qwen2.5-1.5B. Even the 140-token baseline—the most stable configuration—shows substantial rank fluctuations, with stability scores significantly lower than those of 7B-scale models.

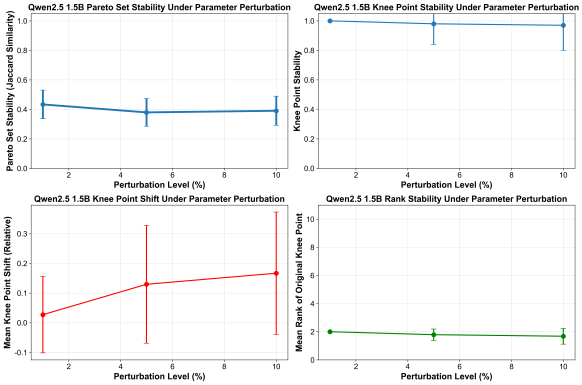


Figure 19: Pareto set robustness under parameter perturbations for Qwen2.5-1.5B. The Jaccard similarity drops below 0.4 at 10% noise intensity, and the knee point exhibits high sensitivity to hyperparameter shifts, indicating an unreliable decision space.

matic increase in decision-making risk.