

NEDOQwen: Diagnosing and Repairing a Turkish-Centric Small Language Model for Multilingual Evaluation

Ahmet Rifat Öztürk Yağız Ekrem Dalar Nedim Mutlu Sezer
Ömer Faruk Aksoy Feyzi Arda Salihoğlu

Abstract

Multilingual language modeling is often evaluated through broad aggregate scores, yet language-specific failures can remain hidden, especially for morphologically rich and under-represented languages. We present NEDOQwen, a Turkish-centric 0.824B decoder-only small language model research artifact, and use it to study a low-cost diagnosis-and-repair workflow for Turkish evaluation. Internal Turkish instruction diagnostics improve after supervised fine-tuning, but external Turkish and Turkic multiple-choice benchmarks reveal weak academic knowledge and severe answer-label bias. A compact benchmark-safe repair stage yields a small improvement on TurkishMMLU-sub, from 18.11% to 21.00%, and a larger improvement on TUMLU-mini Turkish, from 21.33% to 32.22%, while an answer-label-only ablation reaches 19.56% and 27.00%. We further analyze tokenizer coverage, internal-vs-external evaluation mismatch, prediction-label distributions, paired uncertainty, and compute cost. The contribution is not a state-of-the-art Turkish model, but an auditable case study showing how small language-specific models can expose and partially repair multilingual evaluation failures while remaining unreliable.

1 Introduction

Multilingual representation learning is commonly summarized through broad benchmark averages or multilingual leaderboards. Such reporting is useful, but it can obscure language-specific failures. A model that appears usable in aggregate may still fail on morphology, script-specific characters, answer formatting, benchmark calibration, or educational domains in a particular language. This problem is especially important for underrepresented and morphologically rich languages, where the interaction between tokenizer coverage, training data, and evaluation protocol can produce brittle behavior.

Turkish is a useful case study. It is agglutinative,

uses language-specific characters, and has extensive suffixation that can change surface forms while preserving a stable root. A Turkish-centric model therefore offers a concrete testbed for studying how language-specific tokenization, internal instruction diagnostics, and external multiple-choice benchmarks reveal different failure modes. At the same time, small language models are useful scientific artifacts: they can be trained, inspected, repaired, and rerun with modest compute budgets, making failure analysis more accessible.

We introduce NEDOQwen, a Turkish-centric 0.824B decoder-only small language model (SLM), as a research artifact for multilingual evaluation and repair. We do not claim that NEDOQwen is a strong Turkish language model, state of the art, or production-ready. Instead, we ask: what can a small Turkish-centric model reveal about multilingual evaluation, tokenizer coverage, and benchmark-safe repair?

Our central finding is that internal and external evaluations measure different behaviors. Clean supervised fine-tuning improves internal Turkish instruction diagnostics, but the same checkpoint performs weakly on public Turkish/Turkic multiple-choice benchmark slices. A small benchmark-safe repair stage improves external performance while preserving aggregate internal diagnostic behavior. However, prediction-label distributions reveal persistent C/D answer-label bias, and qualitative examples show local regressions. These results support a conservative claim: the workflow is useful for diagnosing and partially repairing Turkish-centric SLMs, but the model remains far from robust broad competence.

Research questions We study four questions:

1. What failure modes appear when a small Turkish-centric language model is evaluated beyond internal instruction diagnostics?

2. How do tokenizer coverage and answer-label behavior affect Turkish multiple-choice benchmark performance?
3. Can a benchmark-safe repair stage improve external Turkish/Turkic benchmark behavior without training on benchmark test rows?
4. What claims are justified for a low-cost Turkish-centric SLM, and what claims are not?

Contributions We make four contributions. First, we describe a reproducible Turkish-centric 0.824B SLM artifact. Second, we analyze tokenizer coverage on clean SFT text and TurkishMMLU-sub benchmark text, showing that a typed Turkish tokenizer is not automatically more compact than mature multilingual tokenizers. Third, we compare internal instruction diagnostics with external Turkish/Turkic benchmark slices and expose answer-label bias. Fourth, we run a benchmark-safe repair stage, an answer-label-only ablation, a post-repair 220-prompt regression check, and label-distribution analysis to separate genuine partial repair from superficial calibration.

2 Related Work

Multilingual representation learning Multilingual representation learning has progressed from multilingual masked language models such as multilingual BERT (Devlin et al., 2019) and XLM-R (Conneau et al., 2020) to large open multilingual generative models such as BLOOM (BigScience Workshop, 2022) and Qwen2.5 (Yang et al., 2024). These systems demonstrate that shared multilingual representations can transfer across languages, but aggregate evaluations can understate the remaining challenges of individual languages.

Turkish and Turkic evaluation TurkishMMLU measures massive multitask understanding in Turkish using school-like multiple-choice questions (Yüksel et al., 2024). TUMLU expands the multilingual evaluation frame to native Turkic-language understanding (Isbarov et al., 2025). TR-MMLU is also relevant for Turkish large language model evaluation, but we do not report TR-MMLU scores because the likely official endpoint tested during this project required authorization (Bayram et al., 2025). Our goal is not leaderboard comparison; we use public slices as diagnostic instruments.

Property	Value
Parameters	824,256,000
Layers	24
Vocabulary	65,536
Hidden size	1,536
Attention heads	16
KV heads	8
MLP dimension	4,096
Context length	1,024
Position encoding	RoPE
Norm / activation	RMSNorm / SwiGLU
Implementation	Custom PyTorch
Embedding tying	No

Table 1: Architecture of NEDOQwen.

Tokenization and morphology Subword tokenization methods such as byte-pair encoding (Sennrich et al., 2016) and SentencePiece (Kudo and Richardson, 2018) are central to multilingual NLP. For morphologically rich languages, tokenization determines how roots, suffixes, and rare surface forms are represented. A language-aware tokenizer may improve interpretability or coverage for some domains, but it is not guaranteed to outperform a mature multilingual tokenizer trained at larger scale. This paper treats tokenizer behavior as an empirical object rather than an assumed advantage.

Repair-oriented fine-tuning and contamination Post-training adaptation can improve task behavior, but it can also overfit benchmark style or leak evaluation content. We therefore distinguish row-level leakage from style-level transfer. The repair data used here intentionally teaches generic answer-label calibration and short-form behavior, but it excludes benchmark question text, answer text, answer keys, and row identifiers. This distinction is important for multilingual evaluation, where benchmark-specific improvements can otherwise be mistaken for broad language competence.

3 Model Artifact

NEDOQwen is a decoder-only transformer implemented in custom PyTorch. Table 1 summarizes the architecture. The model uses rotary positional embeddings (Su et al., 2021), RMSNorm (Zhang and Sennrich, 2019), SwiGLU feed-forward layers, grouped-query attention, a 65,536-entry typed Turkish tokenizer, and no embedding tying. The model is not currently packaged as a Transformers-native checkpoint, which limits immediate reuse; we treat it as an auditable research artifact rather than a deployable assistant.

4 Tokenizer and Representation Coverage

A central multilingual representation learning question is whether the model’s typed Turkish tokenizer provides useful coverage for Turkish morphology and evaluation text. We therefore measure token-per-word and unknown-token rates on two sources: 5,000 clean SFT examples and TurkishMMLU-sub question+choice text. We also compare against available tokenizers from Qwen2.5, Phi-3.5, and Trendyol-LLM.

Source	Tokenizer	Tok/w	UNK rate	UNK
Clean SFT	NEDO 65K	3.0008	0.8152%	5,909
TMMLU	NEDO 65K	2.2400	2.1433%	3,106
TMMLU	Qwen2.5	1.9436	0.0000%	0
TMMLU	Phi-3.5	3.0358	0.0000%	0
TMMLU	Trendyol	2.1137	0.0000%	0

Table 2: Tokenizer coverage. The typed Turkish tokenizer is language-aware, but it is not automatically more compact than mature multilingual tokenizers and has nonzero UNK rate on benchmark text. TMMLU denotes TurkishMMLU-sub.

Table 2 gives an important negative result. The NEDO tokenizer is typed and Turkish-aware, yet on TurkishMMLU-sub it produces 2.24 tokens per word and a 2.1433% UNK rate, while the Qwen2.5 tokenizer produces 1.9436 tokens per word and zero UNKs. This should not be read as a failure of Turkish-aware tokenization in general. Rather, it shows that language-aware design alone is not enough; training coverage, vocabulary construction, normalization, and fallback handling matter.

To test whether tokenizer coverage correlates with benchmark performance, we bucket TurkishMMLU-sub examples by NEDO UNK count. Table 3 shows diagnostic correlations. The relationship is not monotonic: high-UNK examples are not always the weakest bucket. This suggests that tokenizer coverage is one source of difficulty, but not the only one; answer-label bias, knowledge gaps, and protocol alignment also matter.

5 Evaluation Protocol

Internal diagnostics We use Turkish prompt diagnostics to triage checkpoints. The expanded suite contains 220 prompts across factual QA, arithmetic, reasoning, linguistic knowledge, uncertainty, identity, translation, summarization, rewriting, format following, and repetition stress. Metrics are expected-string score, repetition ratio, and bad-output count. These metrics are lightweight and

UNK bucket	n	Pre acc.	Repair acc.
0	236	15.68%	17.80%
1–2	309	17.15%	21.68%
3–5	185	17.84%	23.24%
≥ 6	170	23.53%	21.76%

Table 3: TurkishMMLU-sub accuracy by NEDO tokenizer UNK bucket. These are diagnostic correlations, not causal estimates.

reproducible, but they can under-credit correct paraphrases and over-credit partial string matches. We therefore use them as regression diagnostics, not as human-quality evaluation.

External benchmarks We evaluate two public external slices. TurkishMMLU-sub contains 900 public Turkish multiple-choice questions with five answer options. TUMLU-mini Turkish contains 900 Turkish questions from TUMLU-mini with four answer options. For both, we use a zero-shot answer-label log-probability protocol: the prompt contains the question and answer choices, and the model selects the answer letter with the highest normalized log-probability over available labels. This is a diagnostic protocol, not necessarily the official leaderboard protocol.

Baselines We compare against random answer-count baselines, Qwen2.5-0.5B-Instruct as a size-neighborhood reference, and Qwen2.5-1.5B-Instruct as a stronger multilingual reference. All reported external results use the same diagnostic answer-label protocol.

6 Diagnosis: Internal vs. External Mismatch

The first failure mode is an evaluation mismatch. Internal diagnostics improve substantially after clean SFT, but external benchmark behavior remains weak. Table 4 shows the 220-prompt internal trajectory. Moving from the Step9000 base checkpoint to clean SFT raises expected-match score from 0.187 to 0.341 and reduces bad outputs from 137 to 19. The repair checkpoint preserves aggregate internal behavior: expected score remains 0.339 and bad outputs decrease to 15.

External behavior tells a different story. The clean SFT checkpoint scores only 18.11% on TurkishMMLU-sub and 21.33% on TUMLU-mini Turkish, near or below random answer-count expectations. This indicates that internal prompt improvement does not imply benchmark knowl-

Checkpoint	Exp. $\uparrow$	Rep. $\downarrow$	Bad $\downarrow$
Base Step9000	0.187	0.216	137
Clean SFT	0.341	0.088	19
Repair-600	0.339	0.090	15
Qwen2.5-1.5B-Inst.	0.525	0.173	5

Table 4: 220-prompt internal diagnostic. Exp.: expected-string score; Rep.: repetition ratio; Bad: bad-output count. Repair preserves aggregate diagnostic behavior, but qualitative examples reveal local failures.

edge, answer-label calibration, or healthy multiple-choice behavior.

7 Benchmark-Safe Repair Fine-Tuning

External evaluation revealed weak answer-label behavior and school-level multiple-choice performance. To test whether a small targeted repair could improve these behaviors without benchmark-row leakage, we construct a synthetic repair mixture.

The full repair data contains 2,005 repair examples and 2,000 sampled clean SFT examples. Repair categories target answer-letter calibration, arithmetic, Turkish grammar, exact formatting, uncertainty calibration, identity and scope, short explanations, and stable school-level facts (Table 5). Continued SFT runs for 600 steps from the best pre-repair checkpoint with learning rate 10^{-5} , batch size 2, gradient accumulation 8, and block size 1024. The objective is not to memorize benchmark answers; it is to test whether failure modes revealed by evaluation can be partially repaired.

Category	Count	Purpose
Arithmetic	825	School-level computation
Answer-letter format	250	Label calibration
Turkish grammar	240	Language form
Exact format	200	Short-form control
Uncertainty	150	Avoid overclaiming
Short explanation	150	Concise reasoning style
Identity/scope	100	Model self-description
Stable school facts	90	General factual anchors
Clean SFT mix	2,000	Retain generation behavior

Table 5: Repair data categories. Counts for stable school facts aggregate small Turkish, science, history, geography, math, and four-choice subsets.

Contamination control The repair data excludes TurkishMMLU-sub and TUMLU-mini Turkish test rows. We do not copy benchmark question text, answer text, answer keys, or row identifiers into

the repair mixture. An exact question/choice overlap scan found zero matches between the repair files and the two evaluation slices. We distinguish row-level leakage from style-level transfer: the repair intentionally teaches generic answer-label calibration and short-form task behavior, so it is benchmark-safe with respect to test-row content, but not independent of the broad multiple-choice format.

8 Results

8.1 External Benchmark Improvements

Table 6 reports the main external results. The full repair checkpoint improves TurkishMMLU-sub by +2.89 percentage points and TUMLU-mini Turkish by +10.89 percentage points. Paired uncertainty estimates suggest that the TurkishMMLU-sub gain is small and borderline, whereas the TUMLU-mini Turkish gain is larger and more robust (Appendix A). The repaired checkpoint exceeds the Qwen2.5-0.5B reference under this diagnostic protocol, while remaining far behind Qwen2.5-1.5B-Instruct. This is consistent with our framing: the contribution is partial repair and diagnosis, not model superiority.

8.2 Repair Ablation

The answer-label-only ablation improves both benchmarks, confirming that label calibration is a real failure mode. However, full repair improves further on both datasets. This is consistent with benefits from the broader repair mixture beyond answer-label calibration. The present ablation does not isolate the contributions of arithmetic, grammar, formatting, short explanations, or stable facts. Still, the effect is limited and should not be interpreted as broad understanding.

8.3 Prediction Label Distribution

Table 7 shows prediction-label distributions. Pre-repair behavior is dominated by C predictions on both benchmarks. Full repair reduces C-only collapse, but it still predicts almost exclusively C/D and rarely A/B/E. This is one of the strongest limitations of the repaired model: the benchmark scores improve, but the label distribution remains unhealthy.

8.4 Qualitative Before/After Behavior

Qualitative examples show why automatic metrics must be interpreted carefully. Repair often reduces

Model	Params	TMMLU-sub	TUMLU-mini	Role
Random baseline	—	~20%	~25%	Reference
Qwen2.5-0.5B-Instruct	0.5B	19.33%	19.89%	Size reference
NEDO pre-repair	0.824B	18.11%	21.33%	Clean SFT
NEDO label-only	0.824B	19.56%	27.00%	Ablation
NEDO full repair	0.824B	21.00%	32.22%	Full repair
Qwen2.5-1.5B-Instruct	1.5B	28.67%	42.00%	Strong reference

Table 6: External benchmark results under the same zero-shot answer-label log-probability diagnostic protocol. TUMLU-mini results are for Turkish.

Data	Model	A	B	C	D	Acc.
TMMLU	Pre	2.3	0.1	85.9	11.7	18.11
	Label	1.9	0.1	64.6	33.3	19.56
	Full	0.0	0.0	47.6	51.6	21.00
TUMLU	Pre	2.1	0.1	84.4	13.3	21.33
	Label	2.4	0.0	63.3	34.2	27.00
	Full	0.0	0.0	45.4	54.6	32.22

Table 7: Predicted answer-label percentages and accuracy (%). Repair reduces C-collapse but leaves strong C/D bias. Only A–D percentages are reported here; the source results do not report the E percentage for TurkishMMLU-sub, which has five choices. TUMLU has four choices. Rounding and the omitted E category mean that displayed percentages need not sum to 100.

repetition, but it can remain wrong or introduce new local regressions. Table 8 gives representative samples; additional examples appear in the appendix.

9 Compute Cost and Reproducibility

All automated runs were performed with a GPU workflow. The automated benchmark and repair work cost approximately 23.25 TRY (~0.465 GPU-hours) by actual elapsed GPU time, with the repair round itself costing approximately 5.99 TRY (~0.12 GPU-hours). This low cost matters for multilingual research: language-specific diagnostic loops should be feasible for students, small labs, and local research groups. The artifact package contains model configuration, tokenizer coverage summaries, repair category counts, aggregate benchmark results, contamination-control notes, and qualitative examples. The model weights are not included in the accompanying artifact, and conversion to a Transformers-native checkpoint remains future work.

10 Discussion

Internal diagnostics are necessary but insufficient The internal 220-prompt diagnostic is useful for detecting repetition and generation re-

gressions. However, it does not predict external multiple-choice performance. This mismatch is an important multilingual evaluation lesson: a model can improve on local instruction prompts while still failing benchmark answer-label behavior, academic knowledge, and calibration.

Tokenizer coverage is a finding, not a claim of superiority The typed Turkish tokenizer has nonzero UNK rate and is less compact than Qwen2.5 on TurkishMMLU-sub. This negative result is valuable. It shows that language-aware tokenization must be evaluated empirically, especially for morphologically rich languages. Future tokenizer repair should address character normalization, rare suffix-heavy forms, and robust fallback segmentation.

Repair helps but does not solve label bias The answer-label-only ablation shows that calibration accounts for part of the benchmark gain. Full repair improves further, but prediction distributions remain heavily skewed toward C/D. Therefore, the correct interpretation is partial repair of benchmark behavior, not robust multiple-choice reasoning.

Small models are useful diagnostic artifacts Although NEDOQwen is weak compared with stronger baselines, its small scale makes the full loop auditable: train, diagnose, evaluate externally, repair, ablate, rerun, and report limitations. This is a useful research pattern for underrepresented languages, where access to large-scale compute and fully mature evaluation infrastructure may be limited.

11 Conclusion

We presented NEDOQwen as a Turkish-centric small-language-model artifact for studying multilingual evaluation and repair. The experiments show that internal Turkish diagnostics, tokenizer coverage, external benchmarks, repair ablations, and prediction-label distributions reveal different

Prompt	Pre-repair	Repair-600
“12 + 15 + 3?”	Repeats arithmetic fragments	No repetition, but wrong
Avoid “evet” repetition	Repeats “Evet” many times	Single “Evet”
Ali > Ayşe > Mehmet; longest?	Repeats the question	Mentions Ali, adds extra names
Summarize library hours	Compact correct summary	Repeats output marker

Table 8: Qualitative examples from the 220-prompt rerun. Repair can reduce repetition without solving correctness.

failure modes. A 600-step benchmark-safe repair improves TurkishMMLU-sub and TUMLU-mini Turkish without aggregate internal diagnostic regression, but the model remains weak and label-biased. The appropriate contribution is a low-cost, auditable workflow for Turkish-centric model diagnosis and partial repair, not broad Turkish LM superiority.

Limitations

This work has several limitations. First, NEDO-Qwen is not production-ready and should not be used for high-stakes factual answering, educational grading, legal, medical, or safety-critical decisions. Second, our external evaluation uses a zero-shot answer-label log-probability diagnostic protocol and is not necessarily leaderboard-comparable with official benchmark reporting. Third, full TurkishMMLU access was not used, and TR-MMLU was not run because dataset access was unavailable during this project. Fourth, human evaluation is not included. Fifth, the repair data is synthetic and small; even without row-level benchmark leakage, it can teach broad benchmark style. Sixth, the repaired model still has strong C/D label bias and local qualitative regressions. Seventh, the tokenizer has nonzero unknown-token rate on benchmark text. Finally, the model is custom PyTorch and not yet packaged as a standard Transformers-native checkpoint, limiting reuse.

Ethics Statement

This work supports more inclusive language technology by focusing on Turkish and Turkic-language evaluation rather than only English-centric capability. At the same time, the model is experimental and unreliable. We explicitly avoid production or superiority claims. Synthetic repair data can overfit shallow formats, and benchmark improvements may not transfer to real users. We therefore present NEDOQwen as a research artifact for diagnosis and repair, not as an assistant for deployment.

References

- M. Ali Bayram, Ali Arda Fincan, Ahmet Semih Gümüş, Banu Diri, Savaş Yıldırım, and Öner Aytaş. 2025. [Büyük Dil Modelleri için TR-MMLU Benchmarkı: Performans Değerlendirmesi, Zorluklar ve İyileştirme Fırsatları](#). *arXiv preprint arXiv:2508.13044*.
- BigScience Workshop. 2022. [BLOOM: A 176b-parameter open-access multilingual language model](#). *arXiv preprint arXiv:2211.05100*.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. [Unsupervised cross-lingual representation learning at scale](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Jafar Isbarov, Arofat Akhundjanova, Mammad Hajili, Kavsar Huseynova, Dmitry Gaynullin, Anar Rzayev, Osman Tursun, Aizirek Turdubaeva, Ilshat Saetov, Rinat Kharisov, Saule Belginova, Ariana Kenbayeva, Amina Alisheva, Abdullatif Köksal, Samir Rustamov, and Duygu Ataman. 2025. [TUMLU: A unified and native language understanding benchmark for Turkic languages](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 22816–22838, Vienna, Austria. Association for Computational Linguistics.
- 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.
- 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, Berlin, Germany. Association for Computational Linguistics.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. 2021. *RoFormer: Enhanced Transformer with Rotary Position Embedding*. *arXiv preprint arXiv:2104.09864*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 23 others. 2024. *Qwen2.5 Technical Report*. *arXiv preprint arXiv:2412.15115*.

Arda Yüksel, Abdullatif Köksal, Lütfi Kerem Senel, Anna Korhonen, and Hinrich Schuetze. 2024. *TurkishMMLU: Measuring massive multitask language understanding in Turkish*. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7035–7055, Miami, Florida, USA. Association for Computational Linguistics.

Biao Zhang and Rico Sennrich. 2019. *Root mean square layer normalization*. In *Advances in Neural Information Processing Systems*, volume 32.

A Paired Uncertainty Estimates

Table 9 reports paired bootstrap intervals and approximate sign-flip checks computed from existing row-level outputs. These estimates are diagnostic rather than definitive, but they guide interpretation: the TurkishMMLU-sub gain is small and borderline, while the TUMLU-mini Turkish gain is larger under the same diagnostic protocol.

Contrast	Δpp	95% CI	p
T: full – pre	+2.889	[0.222, 5.556]	0.05
U: full – pre	+10.889	[7.778, 14.000]	<0.001
T: label – pre	+1.444	[-0.333, 3.333]	0.173
U: label – pre	+5.667	[3.444, 8.111]	<0.001
T: full – label	+1.444	[-0.444, 3.333]	0.160
U: full – label	+5.222	[2.889, 7.556]	<0.001

Table 9: Paired uncertainty estimates from row-level outputs (pp: percentage points). T: TurkishMMLU-sub; U: TUMLU-mini Turkish; label: label-only repair. TurkishMMLU-sub improvements should be interpreted cautiously; TUMLU-mini Turkish gains are larger under this diagnostic protocol.

B Subject-Level Accuracy

Table 10 reports subject-level accuracy for the repair-600 checkpoint.

Subject	TMMLU	TUMLU
Biology	25%	33%
Chemistry	24%	29%
Geography	18%	31%
History	22%	41%
Mathematics	23%	30%
Philosophy	18%	33%
Physics	19%	25%
Religion & Ethics	19%	36%
Turkish Lang. & Lit.	21%	32%

Table 10: Subject-level accuracy for NEDOQwen repair-600 on both benchmarks. Performance varies across subjects.

C Difficulty-Level Accuracy

Table 11 reports accuracy by difficulty for the repair-600 checkpoint on TurkishMMLU-sub.

Difficulty	Accuracy
Easy	16.14%
Medium	23.68%
Hard	21.80%

Table 11: Difficulty-level accuracy on TurkishMMLU-sub. The non-monotonic pattern (easy < medium) suggests that the difficulty labels do not fully align with this model’s failure modes.

D Prompt and Choice Sanity Check

A 200-example TurkishMMLU-sub sanity check with the repair-600 checkpoint suggests that results remain protocol-sensitive and label-biased. Original prompt with original choices scored 23.0%; terse prompt with original choices scored 18.0%; original prompt with shuffled choices scored 23.5%; and terse prompt with shuffled choices scored 21.5%. Prediction distributions remained heavily concentrated on C/D or C/E/D depending on prompt wording. We therefore treat benchmark scores as diagnostic rather than leaderboard-style capability claims.

E Additional Qualitative Examples

Additional before/after examples from the 220-prompt diagnostic are included in the supplementary artifact. They intentionally include improvements, unchanged failures, and regressions. Representative patterns are: (i) format prompts improve from empty output to containing the requested fields, but not always valid JSON; (ii) arithmetic prompts often show less repetition after repair, but correctness is still unreliable; (iii) repetition-stress

prompts improve strongly, for example repeated “Evet” collapses to a single answer; (iv) some summarization and rewrite prompts regress locally, showing that repair improves aggregate behavior without guaranteeing open-ended quality.

F Contamination and Leakage Control

The repair data was constructed without using TurkishMMLU-sub or TUMLU-mini Turkish test rows. We did not copy benchmark question text, choice text, answer keys, or row identifiers into the repair mixture. An exact question/choice overlap scan found zero matches between the repair files and the two evaluation slices. The repair set targets category-level behaviors revealed by evaluation: answer-letter formatting, arithmetic, Turkish grammar, exact formatting, uncertainty calibration, identity/scope, short explanations, and stable school-level facts. We distinguish row-level leakage from style-level transfer: the repair intentionally teaches generic answer-label calibration and short-form task behavior, so it is benchmark-safe with respect to test-row content but not independent of the broad multiple-choice format. To make this auditable, the artifact reports repair category counts and construction logs.

G Responsible NLP Notes

NEDOQwen is an experimental research artifact and should not be deployed for high-stakes decisions. The limitations and ethics statement describe its reliability constraints, synthetic repair data, and the scope of the evaluation.