CAMBRIDGE
UNIVERSITY PRESS

TabiBERT:A Large-Scale ModernBERT Foundation Model and A Unified Benchmark for Turkish

Melikşah Türker[1,2], Asude Ebrar Kızıloğlu[1]*, Onur Güngör[1], and Susan Üsküdarlı[1]

[1]Department of Computer Engineering, Boğaziçi University, Türkiye
[2]VNGRS, Istanbul, Türkiye
meliksah.turker@std.bogazici.edu.tr, ebrar.kiziloglu@tum.de, onurgu@pt.bogazici.edu.tr, suzan.uskudarli@bogazici.edu.tr

Abstract

The introduction of BERT established encoder-only Transformer models as a foundational paradigm in natural language processing, motivating extensive research on architectural design and pretraining strategies. Subsequent work addressed limitations of early encoder models through improvements in computational efficiency, training stability, context length, and positional encoding. ModernBERT consolidated these improvements by integrating rotary positional embeddings, FlashAttention-based attention implementations, and refined normalization schemes to support longer contexts and more stable training. Unfortunately, the Turkish NLP landscape lacks a monolingual encoder trained from scratch that systematically incorporates these modern architectural advances.

This work introduces TabiBERT, a monolingual Turkish encoder based on the ModernBERT architecture that is trained from scratch on a large, carefully curated corpus. TabiBERT is pretrained on one trillion tokens sampled from an 84.88B token multi-domain corpus comprising web text (73%), scientific publications (20%), source code (6%), and mathematical content (0.3%). The model supports a context length of 8,192 tokens, which is 16x of the original BERT models. It achieves an inference speedup of up to 2.65x while reducing GPU memory consumption, allowing for larger batch sizes during training and inference. To support rigorous and reproducible evaluation, we introduce the TabiBench benchmark with 28 datasets across eight task categories with standardized splits and evaluation protocols. Performance is summarized using a GLUE-style macro-average score. On TabiBench, TabiBERT attains a score of 77.58, outperforming BERTurk by 1.62 points and establishing new state-of-the-art results on five of eight task categories, with particularly strong gains on question answering (+9.55 points), code retrieval (+2.41 points), and academic understanding (+0.66 points). When compared with task-specific prior best results, including specialized models such as TurkishBERTweet, TabiBERT achieves an average improvement of 1.47 points, indicating robust cross-domain generalization. We release model weights, training configurations, and evaluation code to provide a transparent and reproducible foundation for future Turkish encoder research.

1. Introduction

The emergence of pretrained transformer encoder models has become central to contemporary natural language processing, providing general-purpose representations that support a wide range of downstream tasks. The architectural framework introduced by Bidirectional Encoder

---

*The work was conducted while the author was affiliated with the Department of Computer Engineering, Boğaziçi University, Istanbul, Türkiye. Current address: Technical University of Munich (TUM), Munich, Germany.

Representations from Transformers (BERT) (Devlin et al. 2019) has been a foundational component of natural language understanding, enabling the induction of deeply contextualized linguistic features through self-supervised learning on large-scale corpora. BERT was first trained on large English text corpora and achieved strong baseline performance on a wide range of natural language processing tasks, serving as the basis for many later pretrained language models.

The success of BERT motivated, and thus was followed by extensive research, leading to resource-efficient variants (Liu et al. 2019; Sanh et al. 2019; Jiang et al. 2020; Lan et al. 2020; Clark et al. 2020), domain-specialized models (Beltagy et al. 2019; Huang et al. 2019b; Lee et al. 2020; Chalkidis et al. 2020), and multilingual architectures designed to support cross-lingual transfer (Pires et al. 2019; Vulić et al. 2020; Conneau et al. 2020). Initial refinements focused on pretraining procedures that improved performance without changing the core architecture. For example, RoBERTa (Liu et al. 2019) achieved improved performance via larger and more diverse pretraining corpora, dynamic masking strategies, optimized training regimes, and by eliminating the next sentence prediction objective without altering the original transformer architecture. Several works concurrently focused on efficiency through knowledge distillation (Sanh et al. 2019), parameter sharing strategies (Lan et al. 2020), and alternative pretraining objectives (Clark et al. 2020), enabling computational gains while maintaining or improving performance.

Subsequent architectural innovations sought to address BERT's quadratic complexity, which stems from the self-attention mechanism and limits sequence length and computational efficiency. Beltagy et al. (2020) introduced sparse attention to enable longer sequences, while IO-aware implementations significantly reduced memory overhead (Dao et al. 2022). Positional encoding improvements, such as Rotary Position Embeddings (RoPE) (Su et al. 2024) resulted in improved length extrapolation compared to BERT's original absolute positional embeddings. Normalization strategies were introduced to improve training stability for deeper architectures (Xiong et al. 2020; Zhang and Sennrich 2019). These accumulated improvements were synthesized into ModernBERT, an updated encoder-only architecture that incorporates rotary positional embeddings, FlashAttention technology, alternating attention patterns, and refined pretraining procedures (Warner et al. 2024). Its architecture supports context windows up to 8,192 tokens, sixteen times BERT's original context length of 512, with improved computational efficiency that addresses limitations stemming from fine-tuning instability and restricted sequence length (Xiong et al. 2020; Liu et al. 2019).

While these architectural advances have been widely adopted for high-resource languages, Turkish NLP remains primarily centered on earlier architectural paradigms. BERTurk (Schweter 2020) laid the foundation for transformer-based Turkish NLP as the first monolingual BERT model pretrained from scratch and publicly released. Its widespread adoption across diverse downstream tasks has made it the de facto standard for Turkish language understanding. Following, subsequent models such as ConvBERTurk and various ELECTRA adaptations have emerged on community platforms, although they frequently prioritized immediate availability over technical documentation. Critical specifications regarding corpus composition, preprocessing pipelines, and systematic evaluation protocols are often underspecified, which impedes reproducibility and limits the ability of the research community to build systematically on prior work.

Recent large-scale multilingual efforts have incorporated Turkish alongside other languages. Multilingual BERT (mBERT) (Devlin et al. 2019) and XLM-R (Conneau et al. 2020) share parameters and vocabulary across typologically diverse languages, which dilutes representational capacity for individual languages (Conneau et al. 2020; Vulić et al. 2020). For morphologically rich languages, multilingual tokenizers produce inefficient segmentations, leading to increased token fertility, longer effective sequences, and reduced vocabulary utilization (Rust et al. 2021; Hofmann et al. 2022).

This work identifies and addresses two distinct gaps in Turkish NLP research. First, an *architectural gap* exists: no monolingual Turkish encoder has been trained from scratch to incorporate the efficiency, stability, and representational improvements synthesized in ModernBERT. Second, an

Table 1. : Turkish BERT-based models across scale, training data, architectural features, and benchmark performance.

Model	# Params (M)	Data Size (GB)	Context Length	Code & Math	Flash Attention	TabiBench Score
TurkishBERTweet	163	110	512	✗	✗	67.48
YTU-Cosmos-BERT	111	75	512	✗	✗	72.26
BERTurk	110	35	512	✗	✗	75.96
TabiBERT	149	426	**8,192**	✓	✓	**77.58**

evaluation gap persists due to the lack of a standardized, reproducible benchmarking framework, hindering systematic comparison across model generations and architectural paradigms.

We address these gaps by introducing TabiBERT, a monolingual Turkish encoder, based on the ModernBERT architecture, and TabiBench, a unified evaluation benchmark with standardized protocols. TabiBERT is a monolingual Turkish encoder based on the ModernBERT architecture, trained from scratch on a large, carefully curated corpus. Across TabiBench, TabiBERT achieves an average score of 77.58 compared to the next best model BERTurk's 75.96, representing an improvement of 1.62 absolute points (on a 0-100 scale, following GLUE-style macro-averaging) and establishing state-of-the-art results among Turkish-specific encoders on five out of eight task categories. TabiBERT's technical specifications and results compared to prior Turkish BERT models are summarized in Table 1. Performance gains are particularly substantial on four tasks, with improvements of 9.55 points on question answering, 2.41 points on code retrieval, 0.66 points on academic understanding, as well as 0.60 points on information retrieval. When compared against the previous best result for each task individually, TabiBERT achieves an average improvement of 1.47 points, demonstrating strong cross-domain generalization with a single general-purpose architecture. Beyond accuracy improvements, TabiBERT provides 2.65× faster inference and supports a context length of 8,192 tokens, sixteen times that of prior Turkish BERT models, while reducing GPU memory consumption to enable larger batch sizes during training and inference.

For completeness, we also evaluate against mmBERT (Marone et al. 2025), a recent large-scale multilingual encoder with 307M parameters and a 256,000-token vocabulary that achieves 79.26 on TabiBench. mmBERT represents a different model class, trained on multilingual data at substantially larger scale with correspondingly higher computational requirements. We include it as a reference point to situate TabiBERT within the broader landscape of models applicable to Turkish. Notably, mmBERT's multilingual tokenizer has 41 per cent higher token fertility for Turkish texts compared to TabiBERT's monolingual tokenizer, resulting in longer effective sequences and reduced computational efficiency for Turkish-specific applications (detailed in Section 5). Our work focuses on developing an efficient, monolingual encoder that incorporates modern architectural advances at a scale suitable for widespread research and practical deployment.

The main contributions of this work are as follows:

– TabiBERT (Türker et al. 2025c), a state-of-the-art monolingual Turkish encoder based on the ModernBERT architecture. The model was pretrained for 1T tokens on a corpus spanning diverse domains and achieves state-of-the-art results among Turkish-specific encoders on five of eight benchmark categories. TabiBERT provides substantial efficiency improvements, including 2.65× faster inference, 16× longer context length (8,192 tokens), and reduced memory overhead compared to prior Turkish BERT models.

– TabiBench (Türker et al. 2025b), a unified evaluation framework comprising 28 datasets across 8 task categories with standardized protocols, data splits, and preprocessing pipelines. The benchmark includes specialized domains such as code retrieval and academic text understanding that extend beyond prior Turkish NLP evaluation frameworks.

– Open-source release of model weights and evaluation code (Türker et al. 2025d) to provide a reproducible foundation for Turkish NLP research.

The remainder of this paper is organized as follows: Section 2 reviews prior work on encoder architectures, Turkish language models and evaluation benchmarks, Section 3 describes the TabiBERT model and the pretraining procedure, Section 4 specifies the TabiBench evaluation framework, Section 5 reports the evaluation results of TabiBERT on the TabiBench benchmark, Section 6 discusses the results, Section 7 presents some limitations of our work, and Section 8 concludes the paper and outlines directions for future work.

2. Related Work

The introduction of Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al. 2019) was a significant advancement in encoder-only transformer models. These models were trained on large English corpora and established strong performance baselines across diverse natural language processing tasks. They provided a pretrained model that could be fine-tuned for various downstream tasks.

2.1 Encoder-only Transformer Models & Architectural Evolution

Following BERT, encoder-only transformer models were improved by various architectural innovations or training techniques, including changes to pretraining objectives, optimization strategies, and large-scale data and model scaling, leading to consistent performance gains across a wide range of downstream NLP tasks. RoBERTa (Liu et al. 2019) was trained with the same masked language modeling objective but with a larger training corpus and bigger model sizes, achieving state-of-the-art performance on various NLP tasks. Following, ALBERT (Lan et al. 2020) introduced a number of mechanisms such as parameter sharing and a factorized embedding parameterization, resulting in a significant reduction in model size and memory consumption. This enabled training deeper, wider encoder models while still performing competitively on downstream benchmarks. ELECTRA (Clark et al. 2020) proposed a more sample-efficient pretraining approach based on replaced token detection, where a discriminator model learns to distinguish real input tokens from plausible alternatives generated by a small generator network, resulting in strong performance with substantially reduced pretraining compute. DeBERTa (He et al. 2021) introduced disentangled attention to decouple content and positional representations, improving modeling of relative position information in encoder-only transformers. The success of DeBERTa was followed by DeBERTaV2 (Microsoft 2020), which refined this approach by using improved mask decoding and pretraining strategies. Moreover, DeBERTaV3 (He et al. 2023) further enhanced training efficiency by integrating a replaced-token detection objective inspired by ELECTRA.

Subsequent encoder-only transformer works after DeBERTaV3 focused on improving pretraining efficiency by revisiting training objectives and modernizing the BERT architecture rather than attempting to introduce novel attention mechanisms. MosaicBERT (Portes et al. 2023) worked on engineering a training recipe by combining architectural and systems optimizations such as Flash Attention (Dao et al. 2022), ALiBi positional biases (Press et al. 2021), Gated Linear Units (Shazeer 2020), dynamic unpadding (Zeng et al. 2022), and low-precision normalization. This engineering effort demonstrated that BERT pretraining can run significantly faster and more cost-effectively, while maintaining competitive downstream performance. In parallel, task-oriented pretraining strategies were explored, exemplified by RetroMAE (Xiao et al. 2022), which redesigned masked autoencoding to better support dense retrieval tasks using a BERT-style encoder. FlexiBERT (Tuli et al. 2023) used neural architecture search to identify performant encoder configurations under different computational constraints, aiming to investigate architectural flexibility and efficiency trade-offs. Renewed interest in encoder-only models has led to modernized successors to BERT.

ModernBERT (Warner et al. 2024) revisited the original BERT model, extending context length and updating architectural elements and design choices. NeoBERT (Breton et al. 2025) and OptiBERT (Dervishi et al. 2025) emphasized compute-optimal scaling and practical deployment. Together, these studies show a trend toward improving the efficiency and practicality of BERT-style encoder models, while maintaining strong performance on language understanding tasks.

2.2 Monolingual Turkish BERT Models

The success of transformer-encoder models motivated a series of BERT models known as BERTurk to be trained on Turkish data (Schweter 2020). Several variations of BERT, such as DistilBERT (Sanh et al. 2019), ConvBERTurk (Schweter 2025), and ELECTRA (Clark et al. 2020) were also trained on Turkish data by the same authors (Schweter 2020). These models were generally trained on a 35GB corpus consisting of 4.4B tokens drawn from the Turkish OSCAR corpus (Abadji et al. 2022), a Wikipedia dump[a], and various OPUS corpora (Tiedemann 2012). Some models, like ConvBERTurk and ELECTRA, were also trained on the Turkish portion of the mC4 corpus (Xue, Linting et al. 2021; Schweter 2020), which is a cleaned version of the public web crawl data of Common Crawl. These models have been evaluated on various downstream tasks (such as part-of-speech tagging, named entity recognition, and question answering) where they generally outperform their multilingual counterparts (Schweter 2020; Wu and Dredze 2019; Conneau et al. 2020). Despite their popularity, these models were limited in transparency as they were not products of academic publications.

Following, two Turkish BERT models were introduced with academic publications. YTU-Cosmos-BERT (Kesgin et al. 2023) utilized an uncased Turkish tokenizer and was trained on a corpus of an unknown number of tokens reported to be around 75GB in size, that consists of text collected from the web, Turkish Wikipedia and novels. TurkishBERTweet (Najafi and Varol 2023) was tailored for social media. It also employed an uncased tokenizer, specifically designed for social media texts, by the addition of special tokens that can encode special information in tweets like hashtags, mentions, and URLs. It was pretrained with a corpus of around 110GB in uncompressed form, containing about 900 million tweets of 10 tokens on average, which totals to a sum of 9B tokens when extrapolated. Recently, a RoBERTa-based model for Turkish was published (Scheible-Schmitt and Schweter 2025), which has been an improvement over the previous RoBERTa-based models for Turkish (Tas 2024) as it was trained on a larger corpus, using a larger batch size. However, it did not achieve a significant improvement over the previous models.

2.3 Multilingual BERT Models

The development of multilingual pretrained language models began with Multilingual BERT (mBERT) (Devlin et al. 2019), which extended the original BERT architecture to 104 languages by pretraining on a corpus obtained by concatenation of Wikipedia corpora. Despite lacking explicit cross-lingual objectives, mBERT demonstrated surprising zero-shot cross-lingual transfer capabilities (Pires et al. 2019). XLM (Conneau and Lample 2019) worked on enhancing cross-lingual alignment and introduced Translation Language Modeling (TLM) that leverages parallel corpora. XLM-RoBERTa (Conneau et al. 2020) further scaled multilingual pretraining to 2.5TB of CommonCrawl data across 100 languages, achieving state-of-the-art performance using only masked language modeling without parallel data. Later studies proposed different pretraining approaches. InfoXLM (Chi et al. 2021) applied cross-lingual contrastive learning, ERNIE-M (Ouyang et al. 2021) used back-translation masked language modeling to create pseudo-parallel data, and Unicoder (Huang et al. 2019a) combined several cross-lingual pretraining objectives. Architectural improvements have also been explored in multilingual encoder models.

---

[a]An unspecified Wikipedia dump, presumed to date from approximately 2020.

RemBERT (Chung et al. 2020) decoupled input and output embeddings in order to better use the model capacity. mDeBERTa (He et al. 2023) extended DeBERTa's disentangled attention to the multilingual setting and combined it with ELECTRA-style pretraining, resulting in consistent improvements over XLM-RoBERTa on cross-lingual benchmarks. More recently, XLM-V (Liang et al. 2023) addressed the vocabulary bottleneck by improving token allocation across languages with varying lexical overlap.

More recent work has further expanded the landscape of multilingual encoders. EuroBERT (Boizard et al. 2025) introduced multilingual encoders covering European and several widely spoken global languages. Similarly, mmBERT (Marone et al. 2025), building on ModernBERT, incorporated long-context attention and efficient training strategies. It demonstrated strong performance on classification, embedding, and retrieval tasks, particularly for low-resource languages, while largely preserving the inference efficiency of ModernBERT despite its large vocabulary of 256,000 tokens.

However, these broad multilingual approaches often struggle with the linguistic specificity required for morphologically complex languages, motivating the development of language-family-specific or language-specific models.

2.3.1 Curse of Multilinguality

A key challenge in multilingual modeling is the *curse of multilinguality*, a phenomenon where adding more languages to a fixed-capacity model eventually degrades per-language performance (Conneau et al. 2020). It is demonstrated (Wang et al. 2020) that this stems from negative interference between languages competing for shared model parameters, affecting both high-resource and low-resource languages. In a comprehensive study conducted over 10,000 models and across 250 languages (Chang et al. 2024), it is observed that while moderate multilinguality can benefit low-resource languages, particularly when linguistically similar languages are added, performance degrades for all languages as the number of languages increases, suggesting that massively multilingual pretraining may not be optimal for any language involved.

Several architectural approaches have been proposed to address this issue, such as modular designs that use language-specific adapters (Pfeiffer et al. 2022) and expert models trained on subsets of languages (Blevins et al. 2024). These findings motivate the development of more targeted models that focus on either monolingual models or models that focus on related language families rather than attempting to cover all languages in a single model.

2.4 ModernBERT

ModernBERT is one of the recent and most significant bidirectional encoder-only Transformer models. It upgrades the original BERT architecture by utilizing contemporary enhancements to support longer context lengths and improve efficiency and performance in natural language understanding tasks. It was pretrained on a massive corpus of 2 trillion tokens of English text and code, enabling stronger generalization and suitability for both language and code understanding. ModernBERT introduces several architectural techniques aimed at improving efficiency, including Rotary Positional Embeddings (RoPE) Su et al. (2024), Flash Attention (Dao et al. 2022), unpadding (Zeng et al. 2022), GeGLU activations (Shazeer 2020), and alternating attention (Team et al. 2024). Together, these design choices reduce memory usage and inference time compared to conventional encoder models, while preserving strong performance on downstream tasks. These features make ModernBERT effective for classification, retrieval, semantic search, and other NLU tasks, and it serves as an efficient alternative to older BERT variants in research and production settings.

2.4.1 Language-Specific Variants

The success of ModernBERT has inspired several language-specific adaptations, demonstrating the effectiveness of specialized training over multilingual approaches. For example, ModernGBERT is a family of German encoder models (134M, 1B parameters) trained from scratch, incorporating ModernBERT's architectural innovations (Ehrmanntraut et al. 2025). Their work showed that dedicated encoders outperform multilingual models adapted via LLM2Vec (BehnamGhader et al. 2024) in terms of both performance and parameter-efficiency, validating the approach of language-specific model development.

LLM-JP-MODERNBERT is a similar ModernBERT-based model trained on a large-scale Japanese corpus with 8,192 token context length (Sugiura et al. 2025). While achieving competitive results on fill-mask evaluations, the model highlighted the importance of high-quality, language-specific pretraining data for optimal performance in downstream tasks.

2.4.2 Domain-Specific Variants

The ModernBERT architecture has proven particularly effective for specialized domains that require robust long-context understanding and efficient representation learning. Clinical ModernBERT was developed as a transformer-based encoder pretrained on biomedical literature, clinical notes, and medical ontologies, incorporating large-scale resources such as PubMed abstracts and MIMIC-IV clinical data (Lee et al. 2025). Building on ModernBERT's architectural upgrades, most notably long-context support and efficient attention mechanisms, this domain-adapted model demonstrated strong performance in producing semantically rich representations for long-document biomedical tasks, including clinical classification, retrieval, and entity recognition.

BioClinical ModernBERT further extended this line of work through continual pretraining on the largest biomedical and clinical corpus to date, comprising more than 53.5 billion tokens (Sounack et al. 2025). Unlike earlier clinical encoders trained on narrow or single-institution datasets, this model uses data from multiple institutions, domains, and regions. This broader training data improves generalization and enables the model to achieve state-of-the-art performance across many biomedical NLP benchmarks.

Beyond English biomedical text, ModernBERT has also been shown to be effective in non-English and highly technical clinical settings. In Japanese radiology report classification, ModernBERT achieved a 39 per cent reduction in training time and up to a 1.65× speed-up during training and inference, while maintaining comparable classification performance (Yamagishi et al. 2025).

These findings show that ModernBERT is well-suited for domain-specific applications that involve processing long documents efficiently, and specialized vocabularies. They act as motivation to use ModernBERT as a strong base model for further domain-adapted encoder development.

2.4.3 Architectural Analysis and Comparisons

In a controlled study designed to isolate architectural effects from differences in pretraining data, ModernBERT was compared with DeBERTaV3 when both models were pretrained on the same dataset. The results indicate that while ModernBERT offers faster training and inference speeds attributable to its architectural optimizations, the earlier DeBERTaV3 architecture maintained superior sample efficiency and overall benchmark performance under identical pretraining conditions (Antoun et al. 2025). This finding underscores the importance of disentangling architectural improvements from data effects when evaluating advances in encoder design and suggests that modern architectural enhancements may need to be paired with training methodology innovations to fully realize performance gains.

Another line of architectural comparison came from NeoBERT (Breton et al. 2025), a next-generation bidirectional encoder that incorporates state-of-the-art architectural refinements, modern

pretraining regimes, and optimized depth-to-width design. NeoBERT achieves state-of-the-art results on large evaluation suites such as GLUE and the Massive Text Embedding Benchmark (MTEB), regardless of its relatively small model size of 250M parameters and extended context length. It outperforms standard baselines, including BERT Large, RoBERTa Large, NomicBERT, and ModernBERT under identical fine-tuning conditions (Breton et al. 2025). This comparison highlights how architectural choices such as optimal layer configurations, positional encoding mechanisms, and more balanced model scaling can yield efficiency and performance advantages even compared to models with greater nominal capacity.

Together, these comparative analyses illustrate active research interest in the architectural evolution of encoder-only transformers, where improvements are evaluated not only relative to legacy models like BERT and RoBERTa, but also against contemporary designs such as DeBERTaV3 and NeoBERT. Such work highlights trade-offs between efficiency, sample efficiency, and task performance. Moreover, it motivates ongoing investigation into how design choices impact downstream effectiveness across varied NLP benchmarks.

2.5 Turkish NLP Benchmarks

Early evaluation of Turkish NLP models has predominantly relied on individual, task-specific datasets, developed independently for problems such as named entity recognition (NER), sentiment analysis, dependency parsing, or question answering. Examples include Turkish NER datasets derived from Wikipedia (Rahimi et al. 2019) and news (Tür et al. 2003), sentiment datasets collected from the web (Amasyali et al. 2018), and syntactic resources such as Universal Dependencies Turkish treebanks (Zeman et al. 2019; Türk et al. 2023; Marşan et al. 2023). Even though these datasets have been useful in enabling a steady and consistent progress on task-specific problems, they often differ substantially in annotation quality, domain coverage, and evaluation protocols, and many lack widely adopted, fixed train/validation/test splits, limiting the comparability of reported results across studies.

As transformer-based models became dominant, evaluation practices for Turkish largely continued to follow this fragmented paradigm. Recent works frequently assembled their own evaluation datasets or introduced task-specific splits tailored to a single study. For instance, YTU-Cosmos-BERT evaluated models on tasks such as sentiment analysis, news classification, mask prediction, and zero-shot classification using custom-constructed datasets and internally defined splits, rather than established benchmark splits (Kesgin et al. 2023). TurkishBERTweet focused on sentiment analysis and hate speech detection in social media, similarly relying on task-specific tweet datasets and cross-validation-based evaluation protocols rather than standardized benchmark test sets (Najafi and Varol 2023). While the stand-alone evaluations conducted within such works demonstrate practical effectiveness, they make it difficult to draw robust, reproducible comparisons between models trained under different conditions.

In response to these limitations, several Turkish NLP benchmark efforts have emerged that aim to consolidate multiple tasks under unified evaluation frameworks. Mukayese aggregated diverse Turkish NLP tasks, including language modeling, machine translation, NER, summarization, and text classification into a single benchmark suite, providing common baselines (Safaya et al. 2022). TrGLUE adapted the GLUE paradigm to Turkish natural language understanding tasks, though its scope remained largely limited to sentence-level NLU (Altinok 2025). Cetvel, a large-scale LLM-oriented evaluation benchmark, covered a broader set of discriminative and generative tasks while emphasizing cultural and linguistic relevance for Turkish (Kesen et al. 2024). More recently, TR-MTEB introduced a Turkish adaptation of the Massive Text Embedding Benchmark (MTEB), focusing on embedding and retrieval tasks (Baysan and Gungor 2025) by utilizing machine translation.

Despite recent advances, a systematic quality assessment of widely used Turkish benchmark datasets showed that a substantial portion of these datasets fail to meet basic criteria for linguistic

correctness, coherence, and cultural appropriateness, underscoring the need for stricter dataset construction and validation standards (Umutlu et al. 2025). Last but not least, most existing benchmarks focus on traditional NLP tasks, with limited coverage of modern NLP tasks such as academic text understanding, retrieval, and code-related tasks, which have become central to contemporary transformer and LLM evaluation.

Taken together, these observations motivate the need for a standardized Turkish NLP benchmark tailored to transformer-encoder models, characterized by (i) high-quality and well-documented datasets, (ii) fixed and community-accepted evaluation splits, (iii) diverse task coverage, including modern tasks such as coding, academic understanding, and retrieval. Such a benchmark would address the fragmentation evident in recent Turkish BERT studies (Kesgin et al. 2023; Najafi and Varol 2023) and enable more reliable, reproducible, and comprehensive assessment of Turkish language models.

3. TabiBERT: Model Architecture and Pretraining

In this section, we present a comprehensive overview of the pretraining procedure of the TabiBERT model, in terms of the pretraining dataset, tokenizer details, the pretraining objectives, and the model architecture.

3.1 Pretraining Data

We compiled a comprehensive dataset primarily consisting of Turkish monolingual text, supplemented with a small portion of English, coding and math data, to pretrain our model. To address significant size disparities and differences in quality across corpora and ensure balanced representation of different text types, we employed strategic oversampling of smaller but high-quality datasets. After applying the oversampling strategy, our final pretraining corpus consisted of 86.58B tokens, of which 13 per cent were non-Turkish sources. In the rest of the paper, all references to the size of the pretraining corpus will include the effects of oversampling. The details of each corpus are explained in the following subsections, and the training corpora statistics, as well as the sampling rates, are summarized in Table 2.

Web Corpora. We extracted the tur_Latn subset of the FineWeb-2 (Penedo et al. 2025) as the main Turkish web corpus, with a knowledge cutoff date of April 2024. We also included a sample of 5 million English web pages from the FineWeb-2 English dataset to provide English language context during pretraining. Moreover, we included a Turkish Wikipedia dataset comprising articles scraped from Wikipedia's Turkish-language pages in August 2024. This dataset provides encyclopedic knowledge covering a wide range of topics written in a formal and informative style.

Scientific Corpora. We used the Dergipark and Yöktez datasets from TURNA (Uludoğan et al. 2024). Dergipark is a corpus of Turkish academic articles, and Yöktez is a corpus of Turkish graduate theses.

Book Corpus. We used the Book Corpus dataset from TURNA (Uludoğan et al. 2024), a compilation of Turkish fiction and non-fiction books.

Bilkent Creative Writings. The *Bilkent Creative Writings* (Yilmaz 2025) corpus comprises short stories written by Bilkent University students while taking creative writing courses in Turkish.

ParlaMintTR. The *ParlaMintTR* corpus originates from the CLARIN Flagship project[b], and contains transcripts of Turkish parliamentary debates within the broader European ParlaMint initiative.

---

[b] clarin.eu/parlamint

Code Corpus.   The *Code* corpus consists of English code repositories, documentation and instructions obtained by randomly selecting a subset of 2.5M examples from several coding corpora (Replete-AI 2024; Parrot 2022; Bisht 2023; Pham 2023).

Math Corpus.   The *Math* corpus comprises mathematical questions and solutions in English, collected from well-known academic and public datasets (Amini et al. 2019; Hendrycks et al. 2021b; Gao et al. 2024; Yue et al. 2023; Yu et al. 2023; Gallifant 2024).

Corpus	Type	# Docs	# Tokens (B)	Sampling Rate
FineWeb-2 Turkish	Web	88,769,907	56.041	1
FineWeb-2 English	Web	5,000,000	5.676	1
Wikipedia-TR	Web	604,716	0.169	2
Dergipark	Scientific	334,429	1.738	1
Yöktez	Scientific	475,817	14.887	1
Books	Literary	5,078	0.621	3
Bilkent Creative Writings	Creative Text	8,457	0.007	3
ParlaMint-TR	Dialogue	1,333	0.068	5
Code	Code	2,500,000	5.415	1
Math	Math	817,392	0.261	1

Table 2. : Training Corpora Statistics: The complete pretraining corpus consists of 86.58B tokens.

3.2 Tokenizer

In this work, we use the tokenizer developed for *Kumru* LLM (Türker et al. 2025a).

Specifications.   Kumru tokenizer is a modern LLM tokenizer with a vocabulary of 50,176 tokens, trained with the byte-pair encoding (BPE) (Sennrich et al. 2016) algorithm and uses a modern pre-tokenization regular expression (RegEx) to split the text into contiguous character spans that correspond to natural token boundaries, such as words, numeric expressions, punctuation marks, and formatting symbols, before tokenization. As a result of this pre-tokenization RegEx, symbols such as digits, punctuations, and special characters, including line breaks (\n) and tab characters (\t), are represented as single tokens. This helps the model better understand and process code, mathematical equations, and page structure.

   Its design decisions were based on two recent works. Vocabulary size is selected to be computationally optimal for monolingual model training (Ali et al. 2024). Training algorithm and pre-tokenization RegEx are chosen based on their demonstrated effectiveness for LLM tokenization (Dagan et al. 2024).

Training Dataset.   The tokenizer was trained on a dataset mixture that consists of 95 per cent Turkish and 5 per cent English, code, and math, a similar mixture to our model's pretraining data. The dataset mixture was curated to support a wide range of language capabilities, including proficiency in Turkish and English, as well as the ability to handle code snippets and mathematical expressions.

Adaptation for TabiBERT.   To use a tokenizer developed for a decoder-only LLM with TabiBERT, we added the special tokens [CLS] and [SEP] and modified the input processing template to make it compatible with BERT tasks.

Fertility.   Fertility (Rust et al. 2021) measures the average number of tokens required to represent text and serves as an indicator of tokenizer representation efficiency. While lower fertility is generally desirable in language modeling, vocabulary size and fertility involve clear trade-offs with model size and computational requirements (Ali et al. 2024).
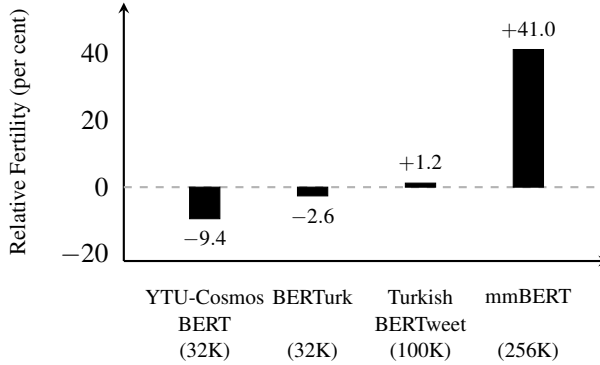
Figure 1: Relative tokenizer efficiency, in per cents, on Turkish text across established Turkish BERT models and mmBERT, normalized to TabiBERT (dashed baseline). X-axis labels indicate models with their vocabulary sizes, while the Y-axis reports relative token counts required for text representation. Although TabiBERT's conscious allocation of tokens to structural elements yields slightly higher fertility than BERTurk (comparable despite a larger vocabulary), this design choice enables superior handling of code, mathematical content, and document structure. The multilingual mmBERT, despite its 256K-token vocabulary, exhibits 41.0 per cent higher fertility, reflecting greater subword fragmentation on Turkish text.

We measure fertility values by tokenizing 1M randomly selected web pages from our Turkish web corpus and report the relative fertility values compared to TabiBERT in Figure 1. We observe that:

– TabiBERT and BERTurk tokenizers, both being cased tokenizers, achieve comparable fertility values, despite TabiBERT having a significantly larger vocabulary size. This reflects TabiBERT tokenizer's pre-tokenization RegEx design, which allocates dedicated tokens to digits, punctuation, and structural characters. This represents a conscious design choice that trades modest fertility increases for improved representation of structured content.

– YTU-Cosmos-BERT achieves the lowest fertility as a result of using an uncased tokenization strategy.

– As a result of being optimized for social media text, TurkishBERTweet has a similar fertility value to TabiBERT despite being uncased and having twice the vocabulary size.

– Despite its large vocabulary of 256,000 tokens, the mmBERT tokenizer exhibits significantly higher fertility than Turkish-specific tokenizers. This reflects the limited representation of Turkish in large-scale multilingual training corpora such as FineWeb-2 (Penedo et al. 2025), where the Turkish portion is only 0.67 per cent.[c]

3.3 Pretraining Objectives

For pretraining, we adopt the Masked Language Modeling (MLM) objective as in ModernBERT, with a 30 per cent token masking rate. We utilize the official ModernBERT pretraining codebase and recipe to ensure methodological consistency and reproducibility. The pretraining process is organized into 3 progressive phases, each exposing the model to 850B, 125B, and 25B tokens,

---

[c] github.com/huggingface/fineweb-2/blob/main/fineweb2-language-distribution.csv

respectively. Hence, we scale the ModernBERT pretraining recipe by 50 per cent with the aim of pretraining for 1T tokens in total.

To best utilize the dataset and maximize training throughput, we pre-tokenize the entire pre-training corpus and chunk it into pieces that fit within the context length. We perform optimization using the StableAdamW algorithm(Wortsman et al. 2023), paired with a trapezoidal learning rate schedule (Xing et al. 2018) and 1-sqrt decay (Hägele et al. 2024) for improved stability during large-scale training. We pretrain our model for 117 hours on a cluster of 8 x NVIDIA H100 GPUs, resulting in 1T tokens being exposed to the model, that is, 11.83 epochs over the entire dataset.

3.4 Model Architecture

Our model is built on the ModernBERT (Warner et al. 2024) architecture. We use the ModernBERT-base setup, with 22 hidden layers, 12 attention heads, an intermediate size of 1,152, a hidden size of 768, and GLU (Shazeer 2020) expansion to 2,304. We modify the vocabulary size config to 50,176 to match with the tokenizer's vocabulary size.

Consequently, our model incorporates several recent improvements, including Rotary Positional Embeddings (RoPE) (Su et al. 2024) to support longer contexts, Local-Global Alternating Attention (Team et al. 2024) for efficient processing of long texts, as well as unpadding (Portes et al. 2023; Zhang et al. 2024) and Flash Attention (Dao et al. 2022) mechanisms to enable faster inference and training.

Its inference speed is comparable to other models for a standard context length of 512; however, it is 2.65 times faster than others when processing long contexts of 8,192 tokens (Warner et al. 2024). Moreover, for both short and long context lengths, it can process inputs with twice the batch size thanks to its lower GPU memory footprint.

To facilitate replication and reproducibility, we release TabiBERT model weights on HuggingFace (Türker et al. 2025c), with the hopes that it serves as a well-documented baseline incorporating ModernBERT's architectural advances.

4. TabiBench: A Standardized Turkish NLP Benchmark

During the evaluation of TabiBERT, we encountered significant challenges in identifying appropriate fine-tuning datasets for establishing reliable baseline comparisons. A systematic review of commonly used Turkish NLP datasets revealed substantial fragmentation, which hampers the consistency and comparability of reported results. This fragmentation is also reflected in recent Turkish BERT studies, which frequently rely on ad hoc evaluation datasets or custom data splits tailored to individual works (Kesgin et al. 2023; Najafi and Varol 2023).

Our analysis identified several recurring issues:

– Quality inconsistencies: Several widely cited datasets suffer from inconsistent annotations, ambiguous labeling schemes, unclear annotation guidelines, or insufficient scale for robust evaluation.

– Fragmented evaluation practices: Turkish NLP research lacks standardized train/validation/test splits. As a result, different studies adopt varying data subsets, preprocessing pipelines, and evaluation metrics for the same tasks, undermining the reliability of cross-study comparisons.

– Limited task coverage: Existing benchmarks do not provide comprehensive coverage of the diverse task spectrum required to evaluate modern encoder-based models.

Such challenges are common in lower-resource language NLP but are rarely addressed systematically. Rather than compromise evaluation quality by using problematic datasets, we developed

TabiBench as a standardized evaluation framework for Turkish NLP. TabiBench is designed around three core principles:

– Quality over quantity: Systematic review of datasets for annotation consistency, label reliability, and task validity.

– Standardization: Fixed train-validation-test splits with explicit methodology for reproducibility and fair comparison.

– Task diversity: Inclusion of modern NLP tasks such as retrieval, code understanding, and domain specialization, as well as traditional NLU tasks.

By addressing the fragmentation and quality concerns evident in prior Turkish NLP benchmarks, TabiBench aims to enable more reliable, comprehensive, and future-facing evaluation of Turkish encoder models. The remainder of this section describes the task categories and dataset coverage in detail.

4.1 Dataset Selection and Quality Assessment

We systematically review existing Turkish fine-tuning datasets, evaluating both dataset quality and task relevance. To ensure our evaluation benchmark reflects realistic and meaningful downstream applications, datasets are selected based on three criteria:

– Clear pattern-label correspondence: The relationship between input text and output labels must be unambiguous and learnable

– Mutually exclusive labels: Each example must belong to exactly one class, ensuring well-defined classification boundaries

– Documented annotation guidelines: Clear, consistently applied annotation protocols must be available to guarantee label reliability

4.2 Standardized Split Generation

Another issue encountered in Turkish NLP evaluation is the absence of standardized dataset splits. Many existing datasets lack explicit train-validation-test divisions, leading researchers to create inconsistent splits that prevent fair cross-study comparisons. We address this through a systematic split generation methodology applied uniformly across all datasets:

– If a train-validation-test split is available, we use the existing divisions as provided.

– For datasets with a train-test split only, we create a validation split from the training set, sized to match the test set.

– In cases with a train-validation split, we reassign the validation set as the test split, then generate a new validation split from the training data following the approach above.

– In cases with a validation-test split, we split validation into train and validation sets in 80 and 20 per cent proportions, respectively.

– When only a single combined split is present, we partition the data into train, validation, and test sets in 70, 15, and 15 per cent proportions, respectively.

All splits use stratified sampling to preserve class distributions and ensure statistical validity.

4.3 Task Categories and Dataset Coverage

TabiBench encompasses 28 datasets spanning 8 major task categories: text classification (4 datasets), token classification (4 datasets), semantic textual similarity (2 datasets), natural language inference (2 datasets), question answering (2 datasets), information retrieval (6 datasets), code retrieval (4 datasets), and academic understanding tasks (4 datasets). Our benchmark combines 21 carefully curated Turkish datasets with 7 newly created machine-translated datasets for code retrieval and academic understanding evaluation, ensuring both depth and breadth in evaluation. The following sections detail each task category and its corresponding datasets:

– Text classification includes: News Cat (Guney 2024), which contains news articles categorized into five classes; Product Reviews (Barmanbay 2023), a binary sentiment classification dataset; Bil Tweet News (Toraman 2017 2021), a four-class news classification dataset based on social media posts; and Gender Hate Speech Turkish (Toraman et al. 2022; Şahinuç et al. 2023), a three-class dataset for detecting gender-based hate speech.

– Token classification includes: WikiNER (Altinok 2023) and WikiANN-TR (Rahimi et al. 2019), which provide Turkish named entity recognition annotations; and POS-UD BOUN (Marşan et al. 2023) and POS-UD IMST (Türk et al. 2023), which provide Turkish part-of-speech annotations based on Universal Dependencies.

– Semantic textual similarity tasks include: SICK-TR (Dehghan et al. 2025) and STSb-TR (Beken Fikri et al. 2021), which provide Turkish sentence pairs annotated with semantic similarity scores.

– Natural language inference tasks include: SNLI-TR and MultiNLI-TR, which are derived from the Natural Language Inference in Turkish (NLI-TR) (Budur et al. 2020) dataset.

– Question answering tasks include: TQuAD (Metin 2024) and XQuAD (Artetxe et al. 2019), for reading comprehension and cross-lingual question answering.

– Information retrieval tasks include TR-MTEB (Baysan and Gungor 2025), obtained by translating the original MTEB (Muennighoff et al. 2023): BiText, MsMarco-TR, Scifact-TR, Fiqa-TR, NFCorpus-TR, and Quora-TR, covering diverse domains from general web search to scientific and financial documents.

– Code retrieval tasks are translated from English using GPT-4.1 while preserving code integrity:

  – AppsRetrieval (Hendrycks et al. 2021a) consists of programming problem descriptions as queries and corresponding code solutions as documents. Its Turkish variant, Apps-TR, is created by translating only the problem descriptions.

  – CodeSearchNet (Husain et al. 2019) contains code snippets paired with docstrings across six programming languages. The Turkish variant, CodeSearchNet-21K-TR, is constructed by selecting 2,500, 500, and 500 examples per programming language for training, validation, and testing, respectively, and translating the docstrings.

  – CosQA (Huang et al. 2021) contains programming questions paired with corresponding code solutions. The Turkish version, CosQA-TR, is created by translating the questions while keeping the code unchanged.

  – StackOverflow-QA (Li et al. 2025) consists of programming questions paired with their highest-voted answers. The Turkish version, StackOverflowQA-TR, is obtained by translating the natural language components of both questions and answers.

– Academic understanding tasks assess model performance on specialized scholarly text. Three datasets are translated from English using GPT-4.1, while one dataset is natively constructed in Turkish:

  – MedNLI (Romanov and Shivade 2018) is a medical-domain natural language inference dataset consisting of sentence pairs annotated with entailment labels.

  – PubMedRCT-20K (Dernoncourt and Lee 2017) is a sentence-level classification dataset where sentences from scientific abstracts are labeled with rhetorical roles. A balanced subset of 10,000 examples is selected using stratified sampling before translation.

  – SciCite (Cohan et al. 2019) is a three-class citation intent classification dataset, labeling citations as background, method, or result.

  – Thesis-Abstract-Classification-11K is a natively constructed Turkish dataset derived from thesis abstracts collected from the National Thesis Center of Turkey (Ertugrul 2025). The task is to predict the field of study of a thesis abstract among 187 categories. A balanced subset with 60 samples per category is constructed, making this a challenging multi-class classification task due to the large label space, small number of samples per category, and specialized academic language.

4.4 Impact and Availability

TabiBench addresses evaluation infrastructure gaps common to many lower-resource languages lacking standardized benchmarks comparable to GLUE (Wang et al. 2018) and SuperGLUE (Wang et al. 2019) for English. We believe that this approach may serve as a valuable template to be applied to other low/medium resource languages facing similar challenges.

To support reproducibility and community adoption, we release TabiBench on HuggingFace (Türker et al. 2025b) to: (1) establish evaluation best practices for Turkish NLP, (2) enable direct cross-model comparisons for future research, and (3) provide reproducible infrastructure for measuring progress. Beyond supporting this work, TabiBench represents an independent contribution that can facilitate more rigorous Turkish NLP research.

5. Evaluation

This section presents a comprehensive evaluation of TabiBERT across diverse Turkish NLP tasks. We first introduce the baseline models used for comparison, then describe our evaluation methodology, and present results demonstrating TabiBERT's strengths and limitations. Prior to getting into the details, we can note the key findings:

– TabiBERT achieves state-of-the-art performance among monolingual Turkish models with an average score of 77.58, surpassing the previous best by 1.62 points.

– It leads in 5 of 8 task categories, with the largest gains in question answering (+9.55 points, 16 per cent relative improvement), code retrieval (+2.41 points, 4.4 per cent relative improvement).

– Performance on text understanding tasks (classification, NER, STS, NLI) is competitive with established models, demonstrating that long-context optimization does not compromise short-text understanding.

Table 3. : Hyperparameter Search Space

Parameter	Values
Learning Rate	5×10^{-6}, 1×10^{-5}, 2×10^{-5}, 3×10^{-5}
Weight Decay	1×10^{-5}, 1×10^{-6}
Batch Size	16, 32
Epoch	10, with early stopping

5.1 Models Used for Comparison

We compare TabiBERT against three monolingual Turkish encoder-only models based on BERT-Base configuration: BERTurk (Schweter 2020), YTU-Cosmos-BERT (Kesgin et al. 2023), and TurkishBERTweet (Najafi and Varol 2023). BERTurk, though not peer-reviewed, is long-standing and widely adopted in Turkish NLP; YTU-Cosmos-BERT and TurkishBERTweet are peer-reviewed and well-documented. All three are pretrained on large, diverse Turkish corpora, constituting strong representative baselines.

Additionally, we include the multilingual mmBERT (Marone et al. 2025), a ModernBERT-based encoder with 307M parameters covering approximately 1800 languages. mmBERT differs significantly from TabiBERT in scale and training scope ($3\times$ training tokens) and is not directly comparable. We include it as an informative reference point contextualizing Turkish-focused models within large-scale multilingual settings.

5.2 Evaluation Methodology

Hyperparameter tuning.   We perform a systematic hyperparameter search for all model-task pairs to ensure fair comparisons. The search space, informed by prior work (He et al. 2023; Warner et al. 2024; Breton et al. 2025) is presented in Table 3. All models train for up to 10 epochs with early stopping (patience=2) based on validation loss. Best configurations are selected via validation performance for the final test evaluation.

Fine-tuning procedure.   We use optimal fine-tuning parameters identified during hyperparameter tuning, with a linear warmup-decay scheduler (6 per cent warmup, decay to $0.02\times$ learning rate), mixed precision training, and random seed 25 for reproducibility. Configurations follow ModernBERT's fine-tuning examples.[d]

Fair comparison protocol.   Note that we perform hyperparameter tuning followed by fine-tuning for *all* models in our evaluation using TabiBench, not just TabiBERT. This ensures all models are evaluated at their best performance on these tasks, making comparisons attributable to architectural differences and explicit design decisions rather than hyperparameter selection discrepancies.

Context length.   We compute the 95th percentile sequence length for each dataset and set the maximum context length to min(model context limit, dataset 95th percentile). If the 95th percentile exceeds the model's maximum context length, we use the model limit. This design choice enables efficient use of available computer resources while maximizing context utilization. Previous Turkish BERT models are limited to 512 tokens, whereas TabiBERT and mmBERT support input lengths of up to 8,192 tokens, enabling the use of longer contexts when available.

Training objectives.   Task-specific objectives are used across the benchmark: cross-entropy is utilized for text classification, NLI, and token classification; Mean Squared Error (MSE) loss is

---

[d]github.com/AnswerDotAI/ModernBERT/yamls/finetuning/

Table 4. : Comparison of downstream task performance across all evaluated models. For each column, the highest score among the Turkish models (excluding the multilingual mmBERT) is shown in bold. The evaluation metric used for each task type is also displayed in the column headers. TabiBERT achieves the highest score in five out of eight task categories among Turkish models.

Model	# of params	Text Clf	Token Clf	STS	NLI	QA	Academic Understanding	Information Retrieval	Code Retrieval	Total Avg
	M	F1	F1	Pearson	F1	F1	F1	NDCG@10	NDCG@10	(TabiBench)
TurkishBERTweet	163	79.71	92.02	75.86	79.10	38.13	63.12	68.40	43.49	67.48
YTU-Cosmos-BERT	111	**84.25**	93.60	84.68	84.16	31.50	71.78	74.29	53.80	72.26
BERTurk	110	83.42	**93.67**	**85.33**	84.33	60.16	71.40	74.84	54.54	75.96
TabiBERT	149	83.44	93.42	84.74	**84.51**	**69.71**	**72.44**	**75.44**	**56.95**	**77.58**
mmBERT	307	82.54	93.81	87.05	84.38	71.47	72.65	76.20	66.02	79.26

applied for semantic similarity regression; span prediction via cross-entropy is implemented for question answering; and contrastive learning is used for retrieval tasks.

Uncased models. For uncased models (YTU-Cosmos-BERT, TurkishBERTweet), we apply Turkish-aware lowercasing (`text.replace("I", "ı").lower()`).

Evaluation metrics. Task-appropriate metrics are adopted for performance assessment: macro-averaged F1 is used for text classification and NLI; micro-averaged F1 at the word level is used for token classification; the Pearson correlation coefficient is used for semantic textual similarity (STS); the F1 score is used for question answering; and Normalized Discounted Cumulative Gain at rank 10 (NDCG@10) is implemented for retrieval tasks.

Aggregation of results. For each task category, we report a single score via weighted average across datasets, with weights proportional to test set size (ranging 150 to 35K examples). This ensures that the aggregate score reflects performance across all test instances rather than giving equal weight to small and large datasets, which would overrepresent performance on smaller datasets. The detailed results per dataset for each task category are presented in Appendix A.
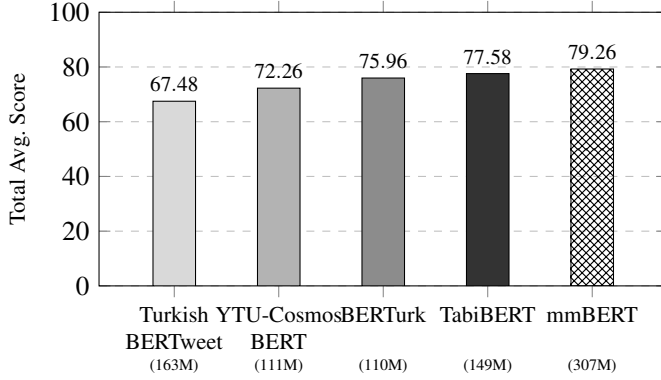
5.3 Results

Table 4 and Figure 2 present comprehensive results across all models and tasks. TabiBERT achieves an overall average of 77.58, surpassing the previous best Turkish model (BERTurk, 75.96) by 1.62 points. Performance is consistent across categories, with TabiBERT leading in 5 of 8 tasks and remaining competitive in others.
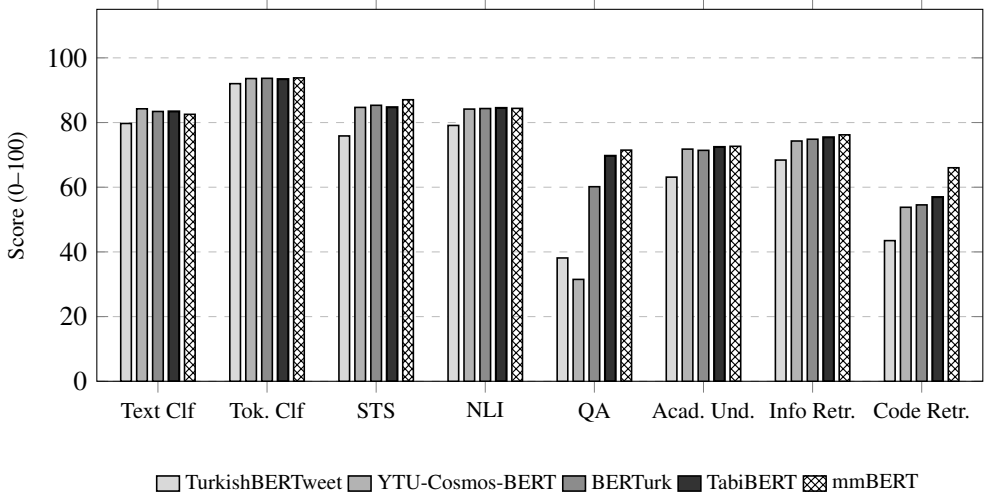
5.3.1 Overall Performance

TabiBERT establishes the highest average among monolingual Turkish models (77.58), followed by BERTurk (75.96) and YTU-Cosmos-BERT (75.83). The multilingual mmBERT achieves 79.26, reflecting its substantially greater resources: $2.1\times$ parameters (307M vs. 149M) and $3\times$ training tokens (3T vs. 1T).

Table 5 compares TabiBERT against the top-performing baseline for each task. TabiBERT achieves a mean score of 77.58, surpassing the aggregate best-performer average of 76.11. The model establishes new state-of-the-art results on five of the eight benchmark tasks. In these instances, TabiBERT outperforms the closest competitor by an average of 2.63 points, with significant gains in QA (+9.55 points) and Code Retrieval (+2.41 points) over BERTurk. Conversely, in cases where TabiBERT trails, the performance gap remains marginal, with an average deficit of 0.55 points and a maximum of 0.81 points.

These results suggest that TabiBERT's architectural advantages are task-specific rather than uniform. Specifically, the model's dominance in QA and retrieval tasks likely reflects the benefits

(a) Overall average performances across all tasks on models. Values in parentheses denote the number of parameters if model in millions.

(b) Average performances by tasks: Text&NLI&Academic (macro-F1), Token Clf (micro-F1), STS (Pearson), QA (F1), and Information Retrieval&Code Retrieval (NDCG@10). TabiBERT consistently performs strongly across all tasks.

Figure 2: Overall and task-specific performance of models on TabiBENCH.

of the ModernBERT extended context window, which facilitates reasoning over longer text spans. While further evaluation across broader long-context benchmarks is required to confirm this mechanism, the current findings indicate that the architectural advancements of ModernBERT translate effectively to the Turkish language domain.

5.3.2 Performance by Task Category

Text understanding tasks. Text classification: Results converge among top models (except TurkishBERTweet). YTU-Cosmos-BERT leads (84.25), followed closely by TabiBERT (83.44) and BERTurk (83.18). Margins under 1 point suggest approaching saturation.

Table 5. : Performance difference (Δ) between TabiBERT and previous best Turkish model across benchmark tasks. Tasks are ordered by relative model performance differences, ranging from the largest negative (trailing) to the largest positive (leading) gap. Positive values indicate tasks where TabiBERT achieves state-of-the-art results; negative values indicate it trails. TabiBERT leads on 5/8 tasks with an average improvement of +2.63 points and trails on 3/8 tasks by an average of 0.55 points.

Task	TabiBERT	Prev. Best	Model	Δ
Text Clf	83.44	**84.25**	YTU-Cosmos-BERT	−0.81
STS	84.74	**85.33**	BERTurk	−0.59
Token Clf	93.42	**93.67**	BERTurk	−0.25
NLI	**84.51**	84.33	BERTurk	0.18
Information Retrieval	**75.44**	74.84	BERTurk	0.60
Academic Understanding	**72.44**	71.78	YTU-Cosmos-BERT	0.66
Code Retrieval	**56.95**	54.54	BERTurk	2.41
QA	**69.71**	60.16	BERTurk	9.55
Average	**77.58**	76.11		**1.47**

Token classification: All Turkish models exceed 92. BERTurk (93.67), YTU-Cosmos-BERT (93.60), and TabiBERT (93.42) cluster within 0.25 points, indicating strong performance across models with limited improvement headroom.

Semantic textual similarity: Top models (excluding TurkishBERTweet) score 84.59–85.33. BERTurk leads (85.33), TabiBERT follows (84.74), then YTU-Cosmos-BERT (84.59).

Natural language inference: TabiBERT achieves the highest Turkish model score (84.51), marginally exceeding BERTurk (84.33) and YTU-Cosmos-BERT (84.16).

These results demonstrate that TabiBERT's long-context optimization preserves sentence-level semantic precision, narrow margins across tasks indicate no performance sacrifice for extended context capability.

Question answering.   TabiBERT shows largest improvement over Turkish baselines: 69.71 F1 versus BERTurk's 60.16 (+9.55 points, 16 per cent relative gain). Two factors explain this: (1) TabiBERT's 8,192-token window (16× that of baselines' 512) processes TQuAD's longer passages without truncation, preserving answer-critical information; (2) stronger capability on complex reasoning requiring extended context integration.

Academic understanding tasks.   TabiBERT leads Turkish models (77.02), exceeding BERTurk (76.36) and YTU-Cosmos-BERT (76.12). Particularly strong on Thesis-Abstract-Classification-11K (187-class task with specialized language and limited per-class data), where pretraining on scientific articles and theses provides a domain advantage, in addition to the impact of its long context support.

Information retrieval.   TabiBERT achieves 75.44 NDCG@10 on TR-MTEB benchmarks, surpassing BERTurk (74.84). Modest gain reflects the benefits of an extended context for processing longer documents without truncation.

Code retrieval.   TabiBERT substantially outperforms Turkish models: 56.95 NDCG@10 versus BERTurk (54.54, +2.41) and YTU-Cosmos-BERT (53.80, +3.15). Improvements stem from the extended context for longer code snippets, exposure to code during pretraining, and the tokenizer's explicit code syntax support. For retrieval tasks requiring document-level coherence and cross-modal understanding, these design choices provide measurable advantages.

5.3.3 Comparison with Multilingual mmBERT

mmBERT achieves 79.26, a 1.68-point advantage over TabiBERT. This reflects fundamental resource differences rather than architectural superiority, as both share ModernBERT architecture with different specialization strategies.

Resource differences.   Model capacity: mmBERT (307M parameters) versus TabiBERT (149M), 2.1× of the latter. Both follow BERT-base configuration (119M non-embedding parameters), but mmBERT's larger embedding matrix (accommodating 1800 languages) provides greater learning capacity at the cost of higher memory requirements.

Tokenization: mmBERT's tokenizer shows 41 per cent higher fertility on Turkish text, requiring 41 per cent more tokens for identical content. This increases computational cost and reduces effective context by 29 per cent (equivalent to a reduction from 8,192 to 5,800 tokens of Turkish text). Finer tokenization may capture morphological nuances via character n-gram fragmentation but sacrifices efficiency and effective context length.

Training scale: mmBERT pretrained on 3T tokens versus TabiBERT's 1T, threefold of the latter. Massive multilingual corpus (1800 languages) enables cross-lingual transfer, particularly from high-resource languages like English. Different philosophy: mmBERT leverages cross-lingual knowledge; TabiBERT focuses computational resources on Turkish-specific phenomena.

Practical trade-offs.   Despite mmBERT's higher scores, TabiBERT offers advantages for Turkish-focused applications: faster inference (due to lower tokenizer fertility), 51 per cent smaller memory footprint, and 29 per cent longer effective context for Turkish text. These efficiency benefits suit resource-constrained deployments or low-latency applications.

Our results demonstrate that targeted, language-specific development with careful architectural choices can achieve competitive performance with substantially lower resource requirements. Model choice depends on application needs: Turkish-focused tasks may benefit from TabiBERT's efficiency and specialized coverage; multilingual applications or those requiring cross-lingual transfer may prefer mmBERT.

6. Discussion

Figure 2 visualizes performance patterns across TabiBench. This section interprets these results, discusses architectural trade-offs, and addresses methodological considerations that informed our evaluation design.

6.1 Interpreting Performance Patterns

Task saturation versus architectural advantages.   Results reveal a clear pattern: narrow margins (~1 point) on text understanding tasks (classification, token classification, STS, NLI) suggest these benchmarks approach saturation for base-sized encoders. All top Turkish models converge around similar scores, indicating limited remaining headroom. Critically, TabiBERT's competitive performance on these tasks demonstrates that long-context optimization does not compromise short-text understanding, a key validation of ModernBERT's architectural design.

Conversely, substantial gains on question answering (+9.55), code retrieval (+2.41), and academic understanding (+0.66) reveal where extended context and specialized pretraining provide measurable advantages. These tasks stress capabilities that saturated benchmarks cannot assess: processing long passages, maintaining coherence across extended contexts, and cross-modal understanding (natural language + code).

Architectural innovations successfully transfer to Turkish.   TabiBERT's consistent performance across diverse tasks validates that ModernBERT's innovations, RoPE positional embeddings, Flash Attention, and optimized layer configurations transfer effectively to Turkish. The model achieves competitive or superior results while providing efficiency benefits (faster inference, lower memory)

over earlier BERT-based Turkish models. This demonstrates that recent encoder architecture advances are not English-specific but generalize to morphologically rich, agglutinative languages.

Monolingual specialization versus multilingual scale.   The TabiBERT versus mmBERT comparison illuminates resource allocation trade-offs. mmBERT's 1.68-point advantage reflects $2.1\times$ parameters and $3\times$ training compute, not fundamental architectural superiority. For practitioners, this suggests a Pareto frontier: monolingual models offer efficiency and effective context length; multilingual models provide cross-lingual transfer and broader coverage. Neither universally dominates; optimal choice depends on application requirements and resource constraints.

Curse of multilinguality.   Despite the now well-known phenomenon of the "curse of multilinguality" (Conneau et al. 2020), it is observed that mmBERT, a multilingual model, outperforms TabiBERT on several tasks. This can be attributed to the gains obtained by the factors, explained in Section 5, that contribute to this phenomenon, outweighing the negative effects of multilinguality in this case.

6.2 Methodological Considerations and Design Decisions

Tokenization-agnostic evaluation for token classification.   For named entity recognition and part-of-speech tagging, we adopted word-level rather than subword-level evaluation. This addresses a fundamental comparability issue: different tokenizers fragment words into varying numbers of subword tokens. A higher-fertility tokenizer would achieve artificially inflated subword-level scores by correctly predicting more fragmented tokens per word, even with identical word-level accuracy.

Word-level evaluation aggregates predictions at word boundaries before computing metrics, ensuring fair comparison regardless of tokenization strategy. While subword-level metrics may be appropriate for analyzing tokenizer-specific behavior, cross-model comparisons require tokenization-agnostic metrics. This principle is broadly relevant in multilingual NLP, where tokenization strategies vary substantially.

Pretraining infrastructure and reproducibility documentation.   During initial pretraining, we observed training throughput 30 per cent of ModernBERT paper's reported speeds on $8\times$H100 SXM GPUs. Experiments with $8\times$A100 SXM showed only marginal decreases, suggesting data loading bottlenecks rather than GPU compute limitations. After consulting ModernBERT authors,[e] we learned reported speeds relied on pre-tokenization, a critical implementation detail not prominently documented. Implementing pre-tokenization resolved the throughput issue.

This experience underscores the importance of comprehensive documentation for reproducible large-scale training. Critical but "obvious" implementation decisions must be explicitly stated; otherwise, researchers attempting replication may encounter substantial inefficiencies or failed reproductions. We have documented our procedures in detail (Section 3) to facilitate future work.

7. Limitations

We acknowledge limitations constraining this work's scope and indicating areas for future research.

7.1 Computational Resource Constraints

Model scale.   We pretrained only base-sized models (149M parameters) due to budget limitations. ModernBERT includes a large variant (395M parameters). Scaling experiments with BERT-large configuration could yield stronger performance, particularly on tasks requiring greater capacity (complex reasoning, low-resource specialized domains).

---

[e]github.com/AnswerDotAI/ModernBERT/issues/236

Corpus size and training duration.   Our pretraining corpus (86.58B tokens) is modest compared to large-scale corpora like FineWeb (18T tokens). TabiBERT was pretrained for 1T tokens versus ModernBERT (2T) and mmBERT (3T). Limited Turkish text availability necessitated 11.83 epochs over the same data, raising potential overfitting concerns. However, we hypothesize that base model capacity (149M parameters) substantially mitigates memorization risks; that is, the model likely lacks sufficient capacity to memorize the corpus. Larger, more diverse Turkish corpora would directly address this limitation.

7.2 Evaluation Constraints

Statistical significance.   Due to computational constraints, we did not conduct significance testing or multiple fine-tuning runs with different random seeds. Reported scores represent single runs. Performance margins <1 point (common on saturated tasks) may or may not reach statistical significance. Formal hypothesis testing with multiple runs would be required for definitive relative performance claims on close-call tasks. We present results as indicative evidence rather than statistically confirmed superiority.

7.3 Translation-Based Datasets: Rationale and Limitations

Thirteen of the twenty-eight TabiBench datasets consist of machine-translated content (all information and code retrieval tasks plus three academic understanding tasks). Seven of them were translated by us using GPT-4.1; six (TR-MTEB) were translated by prior work (Baysan and Gungor 2025). We opted to rely on translations from a pragmatic stance. High-quality Turkish datasets for retrieval and specialized domains are scarce or non-existent, yet these tasks are critical for comprehensive encoder evaluation. For code retrieval, only natural language components are translated while code remains unchanged. As a result of using a state-of-the-art LLM for translation, we present a high quality translation that preserves original meaning and task relevance.

Machine translated content may have potential limitations, such as native linguistic nuances, idiomatic expressions or cultural context not fully captured. However, in our cases, all of the content is technical or formal in nature, minimizing such risks. Moreover, omitting such task categories with the aim of avoiding machine-translated content would severely limit the evaluation scope.

Long-context evaluation gap.   TabiBERT's 8,192-token window represents 16× of earlier models, yet TabiBench provides limited opportunity to demonstrate this. Most tasks involve short texts: sentences or paragraphs (text and token classification, NLI, STS), query-document pairs (retrieval). Tasks requiring long-context understanding, document-level QA, multi-document reasoning, and legislative analysis are absent from the Turkish NLP evaluation infrastructure. This represents a broader Turkish NLP infrastructure gap rather than a work-specific limitation, but constrains comprehensive architectural validation.

7.4 Data Availability

Due to licensing restrictions on portions of our pretraining corpus (academic publications, books from institutional repositories), we cannot release complete training data. This limits exact reproducibility. However, we provide detailed corpus statistics, composition breakdowns, and preprocessing methodology (Section 3), enabling replication with alternative corpora. Researchers can assemble comparable Turkish corpora using publicly available web data, parliamentary records, and permissively licensed scientific literature following our documented procedures.

8. Conclusions and Future Work

This work introduces two resources for Turkish NLP: TabiBERT, a ModernBERT-based encoder demonstrating that recent architectural advances transfer effectively to Turkish; and TabiBench, a standardized evaluation benchmark addressing reproducibility challenges in Turkish NLP research.

8.1 Key Contributions

TabiBERT.   We developed a Turkish encoder incorporating state-of-the-art architectural innovations: RoPE positional embeddings, Flash Attention, optimized layers, and 8,192-token context ($16\times$ that of earlier Turkish models). Pretrained on diverse Turkish text (86.58B tokens, 1T training tokens); web, scientific, parliamentary, academic, plus supplementary bilingual/code/math with a Turkish-specialized tokenizer providing efficient representation (41 per cent lower fertility than multilingual alternatives) while supporting structured content including code and math.

TabiBERT achieves 77.58 average on TabiBench, establishing state-of-the-art among monolingual Turkish models with particular advantages in question answering (+9.55), code retrieval (+2.41), and academic understanding tasks (+0.66). Competitive sentence-level performance that demonstrates long-context optimization does not sacrifice short-text understanding. Substantial efficiency benefits: faster inference, lower memory (51 per cent smaller than mmBERT), 29 per cent longer effective context than higher-fertility multilingual alternatives.

While mmBERT scores higher (79.26), this 1.68-point gap reflects $2.1\times$ parameters and $3\times$ training tokens, not architectural differences. Results demonstrate that careful language-specific design can achieve competitive performance with substantially lower resource requirements.

T*abi*B*ench.*   We developed a standardized benchmark addressing Turkish NLP evaluation infrastructure gaps: 28 curated datasets across 8 task categories with fixed splits and task-appropriate metrics. Manual quality review ensured annotation consistency and adequate size. Fair comparison protocol fine-tunes all models on TabiBench, attributing performance differences to architecture rather than dataset-specific optimization.

Released as open resource to establish evaluation best practices, enable reproducible cross-model comparisons, and provide a template for similar efforts in other lower-resource languages. Addresses long-standing reproducibility issues in Turkish NLP.

Methodological insights.   Documentation of pretraining infrastructure challenges highlights the importance of comprehensive implementation detail reporting for reproducibility. Adoption of tokenization-agnostic metrics (word-level for token classification) addresses fundamental comparability issues relevant across multilingual NLP.

8.2 Broader Implications

For Turkish NLP: Provides both updated model architecture and evaluation infrastructure as foundations for future research.

For lower-resource languages: Methodologies, systematic quality review, language-specific optimization, transparent constraint reporting, and generalize to other languages lacking standardized evaluation frameworks.

For model development philosophy: Supports viability of targeted, resource-efficient development as an alternative/complement to large-scale multilingual approaches. Optimal strategy depends on application requirements (multilingual coverage vs. specialized efficiency) rather than universal superiority.

8.3 Future Directions

Long-context evaluation datasets.   Most significant gap: absence of Turkish datasets specifically evaluating long-context understanding (long and multi-document QA, legislative analysis). Developing such benchmarks would enable comprehensive, extended-context assessment and drive progress on applications requiring long-range understanding.

Statistical validation.   Comprehensive significance testing across benchmarks with multiple seeds would formalize performance comparisons. Robustness analysis under input perturbations (typos, morphological variations, domain shifts) would deepen understanding of capabilities and failure modes.

Scaling experiments.   Pretraining a larger variant (BERT-large, 395M parameters) would reveal whether advantages scale proportionally with capacity. Extended training beyond 1T tokens on expanded corpora would determine whether additional training yields proportional gains or diminishing returns.

Domain-specific optimization.   Fine-tuning or continued training for specialized domains (legal, medical, scientific) could reveal differential benefits for applications requiring domain expertise.

Embedding model development.   Continued training from TabiBERT optimized for retrieval could provide Turkish NLP with a specialized embedding model potentially superior to multilingual alternatives.

Cross-lingual comparative analysis.   Systematic comparison of Turkish-specific versus multilingual ModernBERT variants, controlling for training compute and corpus quality, would quantify monolingual specialization benefits versus cross-lingual transfer.

TabiBench evolution.   Benchmark can expand to additional task types, such as mathematical understanding, and maintain versioned updates as new datasets become available. Living benchmark with periodic releases would support sustained progress measurement.

TabiBERT and TabiBench contribute to Turkish NLP by providing both an efficient, modern encoder model and a standardized evaluation benchmark. These resources address infrastructure gaps that have hindered progress, and we believe they will facilitate more rigorous and reproducible research in Turkish language technology.

Acknowledgments

AI Assistance

We acknowledge the use of AI assistants for grammar checking and language refinement during the preparation of this paper.

Declaration of Competing Interests

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Melikşah Türker is a PhD student at Boğaziçi University and is employed by VNGRS, which provided the cloud GPU infrastructure for model training. Dr. Onur Güngör is a part-time faculty member at Boğaziçi University and is also employed by Udemy.

---

[f]https://vngrs.com/
[g]https://lambda.ai/

While these entities may promote this research for institutional branding or professional visibility, the authors received no direct financial benefits from these activities. All training, evaluation, and manuscript preparation were conducted independently by the research lab to ensure transparency and reproducibility. The companies had no role in the study design or the reporting of results. All other authors declare no competing interests.

References

Abadji, J., **Ortiz Suarez, P.**, **Romary, L.**, and **Sagot, B.** (2022). Towards a cleaner document-oriented multilingual crawled corpus. arXiv:2201.06642.

Ali, M., **Fromm, M.**, **Thellmann, K.**, **Rutmann, R.**, **Lübbering, M.**, **Leveling, J.**, **Klug, K.**, **Ebert, J.**, **Doll, N.**, **Buschhoff, J.**, **and others** (2024). Tokenizer choice for LLM training: Negligible or crucial? In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 3907–3924.

Altinok, D. (2023). A diverse set of freely available linguistic resources for Turkish. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13739–13750, Toronto, Canada. Association for Computational Linguistics.

Altinok, D. (2025). TrGLUE - A natural language understanding benchmark for Turkish. Github repository. https://github.com/turkish-nlp-suite/TrGLUE. Accessed: 2025-12-24.

Amasyali, M. F., **Tasköprü, H.**, and **Çalışkan, K.** (2018). Words, meanings, characters in sentiment analysis. In *2018 Innovations in Intelligent Systems and Applications Conference (ASYU)*, pp. 1–6. IEEE.

Amini, A., **Gabriel, S.**, **Lin, S.**, **Koncel-Kedziorski, R.**, **Choi, Y.**, and **Hajishirzi, H.** (2019). MathQA: Towards interpretable math word problem solving with operation-based formalisms. 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)*, pp. 2357–2367, Minneapolis, Minnesota. Association for Computational Linguistics.

Antoun, W., **Sagot, B.**, and **Seddah, D.** (2025). ModernBERT or DeBERTaV3? Examining architecture and data influence on transformer encoder models performance. arXiv:2504.08716.

Artetxe, M., **Ruder, S.**, and **Yogatama, D.** (2019). On the cross-lingual transferability of monolingual representations. arXiv:1910.11856.

Barmanbay, F. (2023). Turkish Product Reviews Dataset. Hugging Face model repository. https://huggingface.co/datasets/fthbrmnby/turkish_product_reviews. Accessed: 2025-12-24.

Baysan, M. S. and Gungor, T. (2025). TR-MTEB: A comprehensive benchmark and embedding model suite for Turkish sentence representations. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 8867–8887, Suzhou, China. Association for Computational Linguistics.

BehnamGhader, P., **Adlakha, V.**, **Mosbach, M.**, **Bahdanau, D.**, **Chapados, N.**, and **Reddy, S.** (2024). LLM2Vec: Large Language Models are secretly powerful text encoders. arXiv:2404.05961.

Beken Fikri, F., **Oflazer, K.**, and **Yanikoglu, B.** (2021). Semantic similarity based evaluation for abstractive news summarization. In *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)*, pp. 24–33, Online. Association for Computational Linguistics.

Beltagy, I., **Lo, K.**, and **Cohan, A.** (2019). SciBERT: A pretrained language model for scientific text. In *Proceedings of EMNLP-IJCNLP*.

Beltagy, I., **Peters, M. E.**, and **Cohan, A.** (2020). Longformer: The long-document Transformer. arXiv:2004.05150.

Bisht, T. (2023). Python code Alpaca 18K. Hugging Face dataset. https://huggingface.co/datasets/iamtarun/python_code_instructions_18k_alpaca. Accessed: 2025-12-24.

Blevins, T., **Limisiewicz, T.**, **Gururangan, S.**, **Li, M.**, **Gonen, H.**, **Smith, N. A.**, and **Zettlemoyer, L.** (2024). Breaking the curse of multilinguality with cross-lingual expert language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 10822–10837.

Boizard, N., **Gisserot-Boukhlef, H.**, **Alves, D. M.**, **Martins, A.**, **Hammal, A.**, **Corro, C.**, **Hudelot, C.**, **Malherbe, E.**, **Malaboeuf, E.**, **Jourdan, F.**, **Hautreux, G.**, **Alves, J.**, **El-Haddad, K.**, **Faysse, M.**, **Peyrard, M.**, **Guerreiro, N. M.**, **Fernandes, P.**, **Rei, R.**, and **Colombo, P.** (2025). EuroBERT: Scaling multilingual encoders for european languages. arXiv:2503.05500.

Breton, L. L., **Fournier, Q.**, **Mezouar, M. E.**, **Morris, J. X.**, and **Chandar, S.** (2025). NeoBERT: A next-generation BERT. *Transactions on Machine Learning Research*.

Budur, E., **Özçelik, R.**, and **Güngör, T.** (2020). Data and representation for Turkish natural language inference. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, Online. Association for Computational Linguistics.

Chalkidis, I., **Fergadiotis, M.**, **Malakasiotis, P.**, **Aletras, N.**, and **Androutsopoulos, I.** (2020). Legal-BERT: The muppets straight out of law school. In *Findings of EMNLP*.

Chang, T. A., **Arnett, C.**, **Tu, Z.**, and **Bergen, B.** (2024). When is multilinguality a curse? Language modeling for 250 high-

and low-resource languages. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 4074–4096.

Chi, Z., Dong, L., Wei, F., Yang, N., Singhal, S., Wang, W., Song, X., Mao, X.-L., Huang, H.-Y., and Zhou, M. (2021). InfoXLM: An information-theoretic framework for cross-lingual language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3576–3588.

Chung, H. W., Fevry, T., Tsai, H., Johnson, M., and Ruder, S. (2020). Rethinking embedding coupling in pre-trained language models. arXiv:2010.12821.

Clark, K., Luong, M.-T., Le, Q. V., and Manning, C. D. (2020). ELECTRA: Pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020*, Addis Ababa, Ethiopia. OpenReview.net.

Cohan, A., Ammar, W., van Zuylen, M., and Cady, F. (2019). Structural scaffolds for citation intent classification in scientific publications. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pp. 3586–3596, Minneapolis, Minnesota. Association for Computational Linguistics.

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., and Stoyanov, V. (2020). Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8440–8451.

Conneau, A. and Lample, G. (2019). Cross-lingual language model pretraining. *Advances in neural information processing systems*, 32.

Dagan, G., Synnaeve, G., and Roziere, B. (2024). Getting the most out of your tokenizer for pre-training and domain adaptation. arXiv:2402.01035.

Dao, T., Fu, D., Ermon, S., Rudra, A., and Ré, C. (2022). FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems*, volume 35, pp. 16344–16359. Curran Associates, Inc.

Dehghan, S., Amasyali, M. F., and Toraman, C. (2025). A Turkish dataset and BERTurk-Contrastive model for semantic textual similarity. *Journal of Information Systems and Telecommunication (JIST)*, 1(49):24.

Dernoncourt, F. and Lee, J. Y. (2017). PubMed 200k RCT: A dataset for sequential sentence classification in medical abstracts. arXiv:1710.06071.

Dervishi, M., Allauzen, A., Synnaeve, G., and Lecun, Y. (2025). Training compute-optimal transformer encoder models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 35602–35617.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (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*, pp. 4171–4186. Association for Computational Linguistics.

Ehrmanntraut, A., Wunderle, J., Pfister, J., Jannidis, F., and Hotho, A. (2025). ModernGBERT: German-only 1B encoder model trained from scratch. arXiv:2505.13136.

Ertugrul, U. (2025). Turkish academic theses dataset. Hugging Face dataset. https://huggingface.co/datasets/umutertugrul/turkish-academic-theses-dataset. Accessed: 2025-12-24.

Gallifant, J. (2024). Evaluation dataset for Hendrycks Competition Math. Hugging Face dataset. https://huggingface.co/datasets/gallifantjack/hendrycks_competition_math_N_A. Accessed: 2025-12-24.

Gao, B., Song, F., Yang, Z., Cai, Z., Miao, Y., Dong, Q., Li, L., Ma, C., Chen, L., Xu, R., Tang, Z., Wang, B., Zan, D., Quan, S., Zhang, G., Sha, L., Zhang, Y., Ren, X., Liu, T., and Chang, B. (2024). Omni-MATH: A universal olympiad level mathematic benchmark for large language models. arXiv:2410.07985.

Guney, C. (2024). News Cat. Hugging Face model repository. https://huggingface.co/datasets/mcemilg/news-cat. Accessed: 2025-12-24.

Hägele, A., Bakouch, E., Kosson, A., Von Werra, L., Jaggi, M., and others (2024). Scaling laws and compute-optimal training beyond fixed training durations. *Advances in Neural Information Processing Systems*, 37:76232–76264.

He, P., Gao, J., and Chen, W. (2023). DeBERTaV3: Improving DeBERTA using ELECTRA-style pre-training with gradient-disentangled embedding sharing. arXiv:2111.09543.

He, P., Liu, X., Gao, J., and Chen, W. (2021). DeBERTa: Decoding-enhanced BERT with disentangled attention. In *International Conference on Learning Representations*.

Hendrycks, D., Basart, S., Kadavath, S., Mazeika, M., Arora, A., Guo, E., Burns, C., Puranik, S., He, H., Song, D., and Steinhardt, J. (2021)a. Measuring coding challenge competence with APPS. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Hendrycks, D., Burns, C., Kadavath, S., Arora, A., Basart, S., Tang, E., Song, D., and Steinhardt, J. (2021)b. Measuring mathematical problem solving with the MATH dataset. arXiv:2103.03874.

Hofmann, V., Schütze, H., and Pierrehumbert, J. (2022). An embarrassingly simple method to mitigate undesirable properties of pretrained language model tokenizers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 385–393, Dublin, Ireland. Association for Computational Linguistics.

Huang, H., **Liang, Y.**, **Duan, N.**, **Gong, M.**, **Shou, L.**, **Jiang, D.**, **and Zhou, M.** (2019)a. Unicoder: A universal language encoder by pre-training with multiple cross-lingual tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2485–2494.

Huang, J., **Tang, D.**, **Shou, L.**, **Gong, M.**, **Xu, K.**, **Jiang, D.**, **Zhou, M.**, **and Duan, N.** (2021). CoSQA: 20,000+ web queries for code search and question answering. 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)*, pp. 5690–5700.

Huang, K., **Altosaar, J.**, **and Ranganath, R.** (2019)b. ClinicalBERT: Modeling clinical notes and predicting hospital readmission. arXiv:1904.05342.

Husain, H., **Wu, H.-H.**, **Gazit, T.**, **Allamanis, M.**, **and Brockschmidt, M.** (2019). CodeSearchNet challenge: Evaluating the state of semantic code search. arXiv:1909.09436.

Jiang, Z.-H., **Yu, W.**, **Zhou, D.**, **Chen, Y.**, **Feng, J.**, **and Yan, S.** (2020). ConvBERT: Improving BERT with span-based dynamic convolution. *Advances in Neural Information Processing Systems*, 33:12837–12848.

Kesen, I., **Guney, M. C.**, **Erdem, A.**, **and Sahin, G. G.** (2024). CETVEL: A unified benchmark for evaluating Turkish LLMs. GitHub repository. `https://github.com/KUIS-AI/cetvel`. Accessed: 2025-12-24.

Kesgin, H. T., **Yuce, M. K.**, **and Amasyali, M. F.** (2023). Developing and evaluating tiny to medium-sized Turkish BERT models. arXiv:2307.14134.

Lan, Z., **Chen, M.**, **Goodman, S.**, **Gimpel, K.**, **Sharma, P.**, **and Soricut, R.** (2020). ALBERT: A lite BERT for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Lee, J., **Yoon, W.**, **Kim, S.**, **Kim, D.**, **Kim, S.**, **So, C. H.**, **and Kang, J.** (2020). BioBERT: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*.

Lee, S. A., **Wu, A.**, **and Chiang, J. N.** (2025). Clinical ModernBERT: An efficient and long context encoder for biomedical text. arXiv:2504.03964.

Li, X., **Dong, K.**, **Lee, Y. Q.**, **Xia, W.**, **Zhang, H.**, **Dai, X.**, **Wang, Y.**, **and Tang, R.** (2025). CoIR: A comprehensive benchmark for code information retrieval models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 22074–22091.

Liang, D., **Gonen, H.**, **Mao, Y.**, **Hou, R.**, **Goyal, N.**, **Ghazvininejad, M.**, **Zettlemoyer, L.**, **and Khabsa, M.** (2023). XLM-V: Overcoming the vocabulary bottleneck in multilingual masked language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 13142–13152.

Liu, Y., **Ott, M.**, **Goyal, N.**, **Du, J.**, **Joshi, M.**, **Chen, D.**, **Levy, O.**, **Lewis, M.**, **Zettlemoyer, L.**, **and Stoyanov, V.** (2019). RoBERTa: A robustly optimized BERT pretraining approach. arXiv:1907.11692.

Marone, M., **Weller, O.**, **Fleshman, W.**, **Yang, E.**, **Lawrie, D.**, **and Van Durme, B.** (2025). mmBERT: A modern multilingual encoder with annealed language learning. arXiv:2509.06888.

Marşan, B., **Akkurt, S. F.**, **Türk, U.**, **Atmaca, F.**, **Özateş, Ş. B.**, **Berk, G.**, **Bedir, S. T.**, **Köksal, A.**, **Öztürk Başaran, B.**, **Güngör, T.**, **and Özgür, A.** (2023). UD Turkish BOUN. Universal Dependencies Treebank. `https://universaldependencies.org/treebanks/tr_boun/index.html`. Accessed: 2025-12-24.

Metin, E. (2024). TQuAD2: A comprehensive turkish question answering dataset. Hugging Face dataset. `https://huggingface.co/datasets/Metin/Tquad2`. Accessed: 2025-12-24.

Microsoft (2020). Official implementation of DeBERTa and extensions. GitHub repository. `https://github.com/microsoft/DeBERTa`. Accessed: 2025-12-24.

Muennighoff, N., **Tazi, N.**, **Magne, L.**, **and Reimers, N.** (2023). MTEB: Massive text embedding benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 2014–2037.

Najafi, A. and Varol, O. (2023). TurkishBERTweet: Fast and reliable large language model for social media analysis. arXiv:2311.18063.

Ouyang, X., **Wang, S.**, **Pang, C.**, **Sun, Y.**, **Tian, H.**, **Wu, H.**, **and Wang, H.** (2021). ERNIE-M: Enhanced multilingual representation by aligning cross-lingual semantics with monolingual corpora. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 27–38.

Parrot, C. (2022). Code Bagel. Hugging Face dataset. `https://huggingface.co/datasets/codeparrot/github-code`. Accessed: 2025-12-24.

Penedo, G., **Kydlíček, H.**, **Sabolčec, V.**, **Messmer, B.**, **Foroutan, N.**, **Kargaran, A. H.**, **Raffel, C.**, **Jaggi, M.**, **Von Werra, L.**, **and Wolf, T.** (2025). FineWeb2: One pipeline to scale them all – Adapting pre-training data processing to every language. arXiv:2506.20920.

Pfeiffer, J., **Goyal, N.**, **Lin, X.**, **Li, X.**, **Cross, J.**, **Riedel, S.**, **and Artetxe, M.** (2022). Lifting the curse of multilinguality by pre-training modular transformers. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3479–3495.

Pham, N. (2023). Tiny codes. Hugging Face dataset. `https://huggingface.co/datasets/nampdn-ai/tiny-codes`. Accessed: 2025-12-24.

Pires, T., **Schlinger, E.**, **and Garrette, D.** (2019). How multilingual is multilingual BERT? In *Proceedings of the 57th Annual*

Meeting of the Association for Computational Linguistics, pp. 4996–5001, Florence, Italy. Association for Computational Linguistics.

Portes, J., **Trott, A.**, **Havens, S.**, **King, D.**, **Venigalla, A.**, **Nadeem, M.**, **Sardana, N.**, **Khudia, D.**, **and Frankle, J.** (2023). MosaicBERT: A bidirectional encoder optimized for fast pretraining. *Advances in Neural Information Processing Systems*, 36:3106–3130.

Press, O., **Smith, N. A.**, **and Lewis, M.** (2021). Train short, test long: Attention with linear biases enables input length extrapolation. arXiv:2108.12409.

Rahimi, A., **Li, Y.**, **and Cohn, T.** (2019). Massively multilingual transfer for NER. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 151–164, Florence, Italy. Association for Computational Linguistics.

Replete-AI (2024). Code Bagel. Hugging Face dataset. https://huggingface.co/datasets/Replete-AI/code_bagel. Accessed: 2025-12-24.

Romanov, A. and Shivade, C. (2018). Lessons from natural language inference in the clinical domain. arXiv:1808.06752.

Rust, P., **Pfeiffer, J.**, **Vulić, I.**, **Ruder, S.**, **and Gurevych, I.** (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*, pp. 3118–3135. Association for Computational Linguistics.

Safaya, A., **Kurtuluş, E.**, **Goktogan, A.**, **and Yuret, D.** (2022). Mukayese: Turkish NLP strikes back. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 846–863.

Şahinuç, F., **Yilmaz, E. H.**, **Toraman, C.**, **and Koç, A.** (2023). The effect of gender bias on hate speech detection. *Signal, Image and Video Processing*, 17(4):1591–1597.

Sanh, V., **Debut, L.**, **Chaumond, J.**, **and Wolf, T.** (2019). DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. arXiv:1910.01108.

Scheible-Schmitt, R. and Schweter, S. (2025). SindBERT, the sailor: Charting the seas of Turkish NLP. arXiv:2510.21364.

Schweter, S. (2020). BERTurk – BERT models for Turkish. Zenodo: 3770924. Accessed: 2025-12-24.

Schweter, S. (2025). BERTurk v2. Zenodo: 14963493. Accessed: 2025-12-24.

Sennrich, R., **Haddow, B.**, **and Birch, A.** (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)*, pp. 1715–1725.

Shazeer, N. (2020). GLU variants improve Transformer. arXiv:2002.05202.

Sounack, T., **Davis, J.**, **Durieux, B. N.**, **Chaffin, A.**, **Pollard, T. J.**, **Lehman, E.**, **Johnson, A. E. W.**, **McDermott, M.**, **Naumann, T.**, **and Lindvall, C.** (2025). BioClinical ModernBERT: A state-of-the-art long-context encoder for biomedical and clinical NLP. arXiv:2506.10896.

Su, J., **Ahmed, M.**, **Lu, Y.**, **Pan, S.**, **Bo, W.**, **and Liu, Y.** (2024). Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Sugiura, I., **Nakayama, K.**, **and Oda, Y.** (2025). llm-jp-modernbert: A ModernBERT model trained on a large-scale Japanese corpus with long context length. arXiv:2504.15544.

TabiLab, B. (2025)a. TabiBERT academic domain benchmark. Hugging Face collection. https://huggingface.co/collections/boun-tabilab/tabibench-academic. Accessed: 2025-12-24.

TabiLab, B. (2025)b. TabiBERT code retrieval benchmark. Hugging Face collection. https://huggingface.co/collections/boun-tabilab/tabibench-code-retrieval. Accessed: 2025-12-24.

Tas, N. (2024). Roberturk: Adjusting roberta for Turkish. arXiv:2401.03515.

Team, G., **Riviere, M.**, **Pathak, S.**, **Sessa, P. G.**, **Hardin, C.**, **Bhupatiraju, S.**, **Hussenot, L.**, **Mesnard, T.**, **Shahriari, B.**, **Ramé, A.**, **and others** (2024). Gemma 2: Improving open language models at a practical size. arXiv:2408.00118.

Tiedemann, J. (2012). Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Toraman, C. (2017). Early prediction of public reactions to news events using microblogs. In *Seventh BCS-IRSG Symposium on Future Directions in Information Access (FDIA 2017)*, Barcelona, Spain.

Toraman, C. (2021). Event-related microblog retrieval in Turkish. *Turkish Journal of Electrical Engineering and Computer Sciences*.

Toraman, C., **Şahinuç, F.**, **and Yilmaz, E.** (2022). Large-scale hate speech detection with cross-domain transfer. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference (LREC 2022)*, pp. 2215–2225.

Tuli, S., **Dedhia, B.**, **Tuli, S.**, **and Jha, N. K.** (2023). FlexiBERT: Are current transformer architectures too homogeneous and rigid? *Journal of Artificial Intelligence Research*, 77:39–70.

Tür, G., **Hakkani-Tür, D.**, **and Oflazer, K.** (2003). A statistical information extraction system for Turkish. *Natural Language Engineering*, 9(2):181–210.

Türk, U., **Özateş, Ş. B.**, **Marşan, B.**, **Akkurt, S. F.**, **Çöltekin, Ç.**, **Cebiroğlu Eryiğit, G.**, **Gökırmak, M.**, **Kaşıkara, H.**, **Sulubacak, U.**, **and Tyers, F.** (2023). UD Turkish IMST. Universal Dependencies Treebank. https://universaldependencies.org/treebanks/tr_imst/index.html. Accessed: 2025-12-24.

Türker, M., **Ari, E.**, **and Han, A.** (2025)a. Kumru. Hugging Face model repository. https://huggingface.co/vngrs-ai/Kumru-2B. Accessed: 2025-12-24.

Türker, M., **Kızıloğlu, A. E.**, **Güngör, O.**, **and Üsküdarlı, S.** (2025)b. TabiBench: A comprehensive benchmarking framework for Turkish Natural Language Processing. Hugging Face collection, TABILab, CMPE, Boğaziçi University. https://huggingface.co/collections/boun-tabilab/tabibench.

Türker, M., **Kızıloğlu, A. E.**, **Güngör, O.**, **and Üsküdarlı, S.** (2025)c. TabiBERT: A Turkish ModernBERT Model. Hugging Face Model Card, TABILab, CMPE, Boğaziçi University. https://huggingface.co/boun-tabilab/TabiBERT.

Türker, M., **Kızıloğlu, A. E.**, **Güngör, O.**, **and Üsküdarlı, S.** (2025)d. TabiBERT: A Turkish ModernBERT Model. GitHub repository, TABILab, CMPE, Boğaziçi University. https://github.com/boun-tabi-LMG/TabiBERT.

Uludoğan, G., **Yirmibeşoğlu Balal, Z.**, **Akkurt, F.**, **Türker, M.**, **Güngör, O.**, **and Üsküdarlı, S.** (2024). TURNA: A Turkish encoder-decoder language model for enhanced understanding and generation. arXiv:2401.14373.

Umutlu, E. E., **Cengiz, A. A.**, **Sever, A. K.**, **Erdem, S.**, **Aytan, B.**, **Tufan, B.**, **Topraksoy, A.**, **Darıcı, E.**, **and Toraman, C.** (2025). Evaluating the quality of benchmark datasets for low-resource languages: A case study on Turkish. In *Proceedings of the Fourth Workshop on Generation, Evaluation and Metrics (GEM^2)*, pp. 471–487.

Vulić, I., **Litschko, S.**, **Škrlj, B.**, **Šebesta, D.**, **Lample, G.**, **and Daza, J.** (2020). Multilingual BERT needs to be reset: Training a continual cross-lingual language model. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6607–6617, Online. Association for Computational Linguistics.

Wang, A., **Pruksachatkun, Y.**, **Nangia, N.**, **Singh, A.**, **Michael, J.**, **Hill, F.**, **Levy, O.**, **and Bowman, S.** (2019). SuperGLUE: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32.

Wang, A., **Singh, A.**, **Michael, J.**, **Hill, F.**, **Levy, O.**, **and Bowman, S.** (2018). GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP workshop BlackboxNLP: Analyzing and interpreting neural networks for NLP*, pp. 353–355.

Wang, Z., **Lipton, Z. C.**, **and Tsvetkov, Y.** (2020). On negative interference in multilingual models: Findings and a meta-learning treatment. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4438–4450.

Warner, B., **Chaffin, A.**, **Clavié, B.**, **Weller, O.**, **Hallström, O.**, **Taghadouini, S.**, **Gallagher, A.**, **Biswas, R.**, **Ladhak, F.**, **Aarsen, T.**, **Cooper, N.**, **Adams, G.**, **Howard, J.**, **and Poli, I.** (2024). Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. arXiv:2412.13663.

Wortsman, M., **Dettmers, T.**, **Zettlemoyer, L.**, **Morcos, A.**, **Farhadi, A.**, **and Schmidt, L.** (2023). Stable and low-precision training for large-scale vision-language models. arXiv:2304.13013.

Wu, S. and Dredze, M. (2019). Beto, Bentz, Becas: The surprising cross-lingual effectiveness of BERT. arXiv:1904.09077.

Xiao, S., **Liu, Z.**, **Shao, Y.**, **and Cao, Z.** (2022). RetroMAE: Pre-training retrieval-oriented language models via masked auto-encoder. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 538–548.

Xing, C., **Arpit, D.**, **Tsirigotis, C.**, **and Bengio, Y.** (2018). A walk with SGD. arXiv:1802.08770.

Xiong, R., **Yang, Y.**, **He, Y.**, **Zheng, S.**, **Zheng, C.**, **Xing, C.**, **Zhang, H.**, **He, R.**, **Hu, Y.**, **Wang, W.**, **and others** (2020). On layer normalization in the Transformer architecture. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 10524–10533. PMLR.

Xue, Linting, **Constant, Noah**, **Roberts, Adam**, **Kale, Mihir**, **Al-Rfou, Rami**, **Siddhant, Aditya**, **Barua, Aditya**, **and Raffel, Colin** (2021). mT5: A massively multilingual pre-trained text-to-text transformer.

Yamagishi, Y., **Kikuchi, T.**, **Hanaoka, S.**, **Yoshikawa, T.**, **and Abe, O.** (2025). ModernBERT is more efficient than conventional BERT for chest CT findings classification in Japanese radiology reports. arXiv:2503.05060.

Yilmaz, S. F. (2025). Compilation of Bilkent Turkish writings dataset. Zenodo. Zenodo: 15498155. Accessed: 2025-12-24.

Yu, L., **Jiang, W.**, **Shi, H.**, **Yu, J.**, **Liu, Z.**, **Zhang, Y.**, **Kwok, J. T.**, **Li, Z.**, **Weller, A.**, **and Liu, W.** (2023). MetaMath: Bootstrap your own mathematical questions for large language models. arXiv:2309.12284.

Yue, X., **Qu, X.**, **Zhang, G.**, **Fu, Y.**, **Huang, W.**, **Sun, H.**, **Su, Y.**, **and Chen, W.** (2023). MAmmoTH: Building math generalist models through hybrid instruction tuning. arXiv:2309.05653.

Zeman, D., **Nivre, J.**, **and others** (2019). Universal dependencies 2.5. http://hdl.handle.net/11234/1-3105. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL). Accessed: 2025-12-24.

Zeng, J., **Li, M.**, **Wu, Z.**, **Liu, J.**, **Liu, Y.**, **Yu, D.**, **and Ma, Y.** (2022). Boosting distributed training performance of the unpadded bert model. arXiv:2208.08124.

Zhang, B. and Sennrich, R. (2019). Root mean square layer normalization. arXiv:1910.07467.

Zhang, X., **Zhang, Y.**, **Long, D.**, **Xie, W.**, **Dai, Z.**, **Tang, J.**, **Lin, H.**, **Yang, B.**, **Xie, P.**, **Huang, F.**, **and others** (2024). mGTE: Generalized long-context text representation and reranking models for multilingual text retrieval. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 1393–1412.

Appendix A. Fine-Tuning Results

A.1 Fine-tuning Results

The appendix presents detailed performance comparisons across all evaluation tasks. Each table compares TabiBERT against established monolingual Turkish models, with the best-performing monolingual model highlighted in each task. We also include mmBERT as a reference point. While the very large-scale multilingual model mmBERT often achieves competitive or superior performance, direct comparison is complicated by substantial differences in model scale, training data volume, and cross-lingual training objectives.

A.1.1 Text Classification

Model performance is evaluated across four text classification datasets: NewsCat (Guney 2024), BilTweetNews (Toraman 2017 2021), GenderHateSpeech (Toraman et al. 2022; Şahinuç et al. 2023), and ProductReviews (Barmanbay 2023). Table 6 presents these results using the macro F1 score. With the exception of TurkishBERTweet, the models achieve comparable results across the selected tasks. The YTU-Cosmos-BERT model yields the highest performance with a macro F1 score of 84.25, followed by TabiBERT at 83.44. Although TabiBERT does not attain the highest score on individual benchmarks, it is comparable with the top-performing models. Notably, mmBERT underperforms relative to the Turkish models with an average macro F1 score of 82.54.

Table 6. : Downstream performance of models on Text classification (eval metric: macro F1). The number of test samples per task is also given. For each column, the highest score among the Turkish models (excluding mmBERT) is shown in bold.

Model	NewsCat	BilTweetNews	GenderHateSpeech	ProductReviews	Weighted Avg
Test Size	250	150	2,000	35,275	
TurkishBERTweet	91.98	53.16	68.58	80.37	79.71
YTU-Cosmos-BERT	**97.21**	53.39	**71.02**	**85.04**	**84.25**
BERTurk	95.60	**57.87**	68.25	84.30	83.42
TabiBERT	95.20	50.11	69.01	84.32	83.44
mmBERT	94.80	49.06	66.45	83.51	82.54

A.1.2 Token Classification

Token classification performance is evaluated using four datasets: WikiNER (Altinok 2023), WikiANN TR (Rahimi et al. 2019), PosUDBOUN (Marşan et al. 2023), PosUDIMST (Türk et al. 2023). Table 7 presents the micro F1 scores. All Turkish models demonstrate high performance, with micro F1 scores exceeding 92. The BERTurk model achieves the highest overall score of 93.67, taking the leading position across all datasets. Consistent with the results observed in text classification tasks, TabiBERT maintains competitive performance levels.

A.1.3 Semantic Textual Similarity

The performance in semantic textual similarity is evaluated with two datasets: SICK-TR (Dehghan et al. 2025) and STSb-TR (Beken Fikri et al. 2021). Table 8 reports the results using the Pearson correlation score. All models, with the exception of TurkishBERTweet, achieve similar results with BERTurk performing the best with 85.33. TabiBERT follows closely at 84.74.

A.1.4 Natural Language Inference

Natural language inference capabilities are evaluated using two Turkish datasets: SNLI-TR and MultiNLI-TR (Budur et al. 2020). Table 9 reports model performance based on the macro F1 score. With the exception of TurkishBERTweet, the models exhibit similar performance levels with macro F1 scores near 84. These results suggest that the leading models, including the multilingual mmBERT, are comparable for Turkish NLI tasks. Further performance improvements may necessitate more complex datasets or architectural innovations.

Table 7. : Downstream performance of models on Token classification (eval metric: micro F1). The number of test samples per task is also given. For each column, the highest score among the Turkish models (excluding mmBERT) is shown in bold.

Model	WikiNER	WikiANN TR	PosUDBOUN	PosUDIMST	Weighted Avg
Test Size	1,000	10,000	979	1,100	
TurkishBERTweet	72.6	94.05	89.74	93.21	92.02
YTU-Cosmos-BERT	78.96	95.41	89.19	94.30	93.60
BERTurk	**79.44**	**95.37**	**89.82**	**94.60**	**93.67**
TabiBERT	76.41	95.34	90.40	94.07	93.42
mmBERT	75.97	95.85	90.72	94.26	93.81

Table 8. : Downstream performance of models on Semantic Textual Similarity (eval metric: Pearson correlation). The number of test samples per task is also given. For each column, the highest score among the Turkish models (excluding mmBERT) is shown in bold.

Model	SICK-TR	STSb-TR	Weighted Avg
Test Size	4,927	1,379	
TurkishBERTweet	78.35	66.96	75.86
YTU-Cosmos-BERT	85.27	82.55	84.68
BERTurk	**85.95**	83.12	**85.33**
TabiBERT	85.00	**83.84**	84.74
mmBERT	87.81	84.31	87.05

Table 9. : Downstream performance of models on Natural Language Inference (eval metric: macro F1). The number of test samples per task is also given. For each column, the highest score among the Turkish models (excluding mmBERT) is shown in bold.

Model	SNLI-TR	MultiNLI-TR	Weighted Avg
Test Size	9,824	4,923	
TurkishBERTweet	83.05	71.21	79.10
YTU-Cosmos-BERT	87.18	78.15	84.16
BERTurk	**87.21**	78.57	84.33
TabiBERT	86.47	**80.60**	**84.51**
mmBERT	86.44	80.28	84.38

A.1.5 Question Answering

Question answering performance is evaluated using two Turkish datasets: TQuAD (Metin 2024) and XQuAD (Artetxe et al. 2019). Table 10 presents the F1 scores obtained by each model. Among all evaluated tasks, this benchmark reveals the most substantial performance gain achieved by TabiBERT over other Turkish models with an F1 score of 69.71 in comparison to the next best result of 60.16 recorded by BERTurk.

Two primary factors likely contribute to this marked improvement. First, the architectural enhancements of TabiBERT, specifically its deeper encoder and refined attention mechanism, appear effective for the span detection and contextual reasoning needed for question answering. Second, all models were finetuned using context lengths based on the 95th percentile of each dataset, specifically 1,024 tokens for TQuAD and 833 for XQuAD. Since the 95th percentile of TQuAD reaches 1,024 tokens exceeding the 512 maximum supported by the other Turkish models, only TabiBERT with its 8,192 token capacity could be finetuned without truncation. Consequently, only TabiBERT could process the TQuAD examples without information loss during both training and inference.

As for XQuAD, the limited volume of training data (833 training and only 179 test examples) likely constrains model learning, resulting in lower scores across all models irrespective of their context handling capabilities.

Table 10. : Downstream performance of models on Question Answering (eval metric: F1). The number of test samples per task is also given. For each column, the highest score among the Turkish models (excluding mmBERT) is shown in bold.

Model	Tquad	XQuAD	Weighted Avg
Test Size	2,520	179	
TurkishBERTweet	38.04	**39.40**	38.13
YTU-Cosmos-BERT	32.01	24.25	31.50
BERTurk	63.30	15.96	60.16
TabiBERT	**72.34**	32.61	**69.71**
mmBERT	71.55	70.40	71.47

A.1.6 Information Retrieval

We evaluate information retrieval performance on six datasets: BiText, MsMarco-TR, Scifact-TR, Fiqa-TR, NFCorpus-TR, and Quora-TR, which are translated into Turkish by the TR-MTEB group (Baysan and Gungor 2025). Table 11 reports the NDCG@10 scores achieved by each model. Across the TRMTEB retrieval benchmarks, TabiBERT consistently outperforms existing Turkish encoder models, achieving an overall NDCG@10 score of 75.44 and surpassing prior monolingual models such as BERTurk. We attribute these gains in part to TabiBERT's exposure to more diverse pretraining data, including bilingual sources, which appear to be particularly beneficial for retrieval-oriented tasks. TabiBERT attains the highest NDCG@10 scores on the BiText and MsMarco-TR datasets. On Scifact-TR, TabiBERT trails the best-performing model, YTU-Cosmos-BERT, by approximately 6 points (best score: 79.88). However, Scifact-TR is the smallest dataset in the TR-MTEB suite, containing only 339 test instances, and is therefore the least representative benchmark in the retrieval evaluation. For Fiqa-TR, TabiBERT achieves the second-highest NDCG@10 score of 50.67, exceeding BERTurk by 1.21 points. Similarly, on NFCorpus-TR, TabiBERT ranks second with an NDCG@10 score of 31.19, only 1.28 points below the best-performing model, BERTurk. On Quora-TR, all models except TurkishBERTweet, including mmBERT, exhibit nearly identical performance, with scores clustered around 92.50, and TabiBERT performing slightly below this level.

Table 11. : Downstream performance of models on Retrieval tasks. The number of test samples per task is also given. For each column, the highest score among the Turkish models (excluding mmBERT) is shown in bold.

Model	BiText	MsMarco-TR	Scifact-TR	Fiqa-TR	NFCorpus-TR	Quora-TR	Weighted Avg
Test Size	3,000	31,692	339	1,706	12,334	15,675	
TurkishBERTweet	95.98	74.20	68.25	36.23	27.79	86.84	68.40
YTU-Cosmos-BERT	97.25	81.99	**79.88**	**51.10**	28.36	**92.87**	74.29
BERTurk	96.77	81.73	78.47	49.46	**32.47**	92.72	74.84
TabiBERT	**99.42**	**83.31**	74.22	50.67	31.19	92.47	**75.44**
mmBERT	99.49	83.68	80.00	52.29	33.44	92.78	76.20

A.1.7 Code Retrieval

We evaluated code retrieval performance using four datasets: Apps-TR, CosQA-TR, StackOverflowQA-TR, CodeSearchNet-21K-subset-TR (TabiLab 2025b). Table 12 presents the NDCG@10 scores obtained by each model. In code retrieval tasks,

TabiBERT outperforms all other Turkish models, achieving an average NDCG@10 score of 56.95, considerably higher than the next best result of 54.54 by BERTurk. Thanks to its support for longer contexts and the incorporation of code in pretraining data, the model demonstrates clear strengths in code retrieval tasks. Across the individual tasks, TabiBERT secures the top NDCG@10 score in three out of four datasets, with being the second highest in the Apps-TR, lagging behind only by 0.69 points.

Table 12. : Downstream performance of models on Code Retrieval tasks. The number of test samples per task is also given. For each column, the highest score among the Turkish models (excluding mmBERT) is shown in bold.

Model	Apps-TR	CosQA-TR	StackoverflowQA-TR	CodeSearchNet-21K-TR	Weighted Avg
Test Size	3,770	500	1,994	3,000	
TurkishBERTweet	4.48	82.32	67.01	70.41	43.49
YTU-Cosmos-BERT	14.47	82.04	82.65	79.35	53.80
BERTurk	**18.18**	85.72	79.93	78.15	54.54
TabiBERT	17.49	**88.11**	**82.85**	**84.12**	**56.95**
mmBERT	32.24	89.31	90.43	88.36	66.02

A.1.8 Academic Understanding Tasks

We evaluated academic understanding tasks using four datasets: MedNLI-TR, PubmedRCT-10K-TR, SciCite-TR, ThesisAbstractClassification-11K (TabiLab 2025a). The MedNLI-TR dataset is an NLI dataset; the other three datasets are text classification datasets. Table 13 presents the results obtained by each model, reporting the macro F1 scores. We observe substantial improvements in academic understanding tasks, such as NLI and multi-class text classification. For PubmedRCT-10K-TR task, TabiBERT shares the second place with YTU-Cosmos-BERT with a score of 75.32, only 0.29 behind the top performer BERTurk. On SciCite-TR, TabiBERT achieves a score of 83.29, closely trailing YTU-Cosmos-BERT's leading score of 84.09 with a difference of 0.80. On the demanding task of Thesis-Abstract-Classification-11K, characterized by long input contexts (with a 95th percentile context length of 1,152 tokens) and sparse number of samples for each class (a sum of 60 examples per class throughout all splits), TabiBERT attains the highest score among Turkish models with 50.77, outperforming the next best model BERTurk by 1.57 points. Lastly, on the task of MedNLI-TR, TabiBERT achieves the highest score of 80.85, slightly surpassing the second-best YTU-Cosmos-BERT by 0.23 points. By leveraging a pretraining dataset rich in Turkish scientific articles and graduate theses, TabiBERT sets a new benchmark among Turkish models. These results highlight how TabiBERT's exposure to academic understanding texts and ability to process longer inputs underpin its strong performance in this area.

Table 13. : Downstream performance of models on Academic understanding tasks. The number of test samples per task is also given. For each column, the highest score among the Turkish models (excluding mmBERT) is shown in bold.

Model	PubmedRCT-20K-TR	SciCite-TR	Thesis-Abstract-Classification-11K	MedNLI-TR	Weighted Avg
Task Type	*Text Clf*	*Text Clf*	*Text Clf*	*NLI*	
Test Size	1,500	1,859	1,683	1,422	
TurkishBERTweet	70.07	79.59	31.18	72.05	63.12
YTU-Cosmos-BERT	75.32	**84.09**	47.56	80.62	71.78
BERTurk	**75.61**	81.60	49.20	79.90	71.40
TabiBERT	75.32	83.29	**50.77**	**80.85**	**72.44**
mmBERT	74.37	83.27	52.47	80.83	72.65