MICROSOFT RESEARCH

Thinking Augmented Pre-Training

A Simple and Scalable Approach to Improve Data Efficiency in Large Language Model Training

Liang Wang, Nan Yang, Shaohan Huang,
Li Dong, Furu Wei

Research Paper
Work in Progress

December 2025

PRESENTATION OVERVIEW

Research Overview

01

Introduction & Motivation

The Challenge of Data Efficiency in Large Language Model Training

Understanding the fundamental limitations of current LLM training paradigms

THE CORE PROBLEM

The Data Efficiency Challenge

The Scaling Dilemma

Large language models have achieved remarkable success, but their development is constrained by an unprecedented challenge: **training compute is growing exponentially while high-quality data remains limited**.

Modern LLMs are trained on **over 10 trillion tokens**, but the pool of human-authored, organically generated data on the web is finite and has been largely exhausted by existing frontier models.

The Learning Impediment

A primary impediment is that **certain high-quality tokens are exceptionally difficult to learn** given a fixed model capacity. The underlying rationale for a single token can be extremely complex and deep.

When a model's capacity is limited, it may struggle to learn such tokens beyond pure memorization, which will not generalize well to new contexts.

10T+	Exhausted	Limited
Training Tokens (Modern LLMs)	High-Quality Web Data	Model Capacity

Illustrative Example

Problem: "The largest positive integer n for which $n^3 + 100$ is divisible by $n + 10$ is 890."

Target Token (Difficult to Learn)

890

Required Understanding:

- ✓ Polynomial division
- ✓ Remainder Theorem
- ✓ Properties of divisors
- ✓ Multi-step reasoning

The token **"890"** represents the output of intricate, multi-step human reasoning processes that are exceptionally difficult to learn in a single next-token prediction step.

THE PROPOSED SOLUTION

TPT Solution Overview

Thinking Augmented Pre-Training

TPT is a **universal methodology** that augments existing text data with automatically generated thinking traj ectories. These trajectories simulate an expert's in-depth thought process as they analyze the given text.

The method effectively **increases training data volume** and makes high-quality tokens more learnable thr ough step-by-step reasoning and decomposition.

Core Mechanism

1. **Generate** thinking trajectories using open-s ource LLMs
2. **Augment** original documents with thinking content
3. **Train** on augmented data with next-token prediction
4. **Achieve** superior performance with less dat a

Key Advantages

- **No human annotation** required
- **Universal applicability** to any text
- **Highly scalable** document-level processing
- **Dynamic compute allocation** to difficult samp les

Key Achievements

3×
Data Efficiency
Reduction in training tokens for same performance

10%+
Performance Gain
On challenging reasoning benchmarks (3B model)

100B
Training Tokens
Evaluated across diverse configurations

Performance vs. LLaMA-3.1

TPT-8B trained on only **100B tokens** achieves performance comparable to LLaM A-3.1-8B trained on **15T tokens** — a **150× data efficiency improvement**.

02

Core Methodology

Understanding the TPT Framework and Its Properties

A deep dive into thinking trajectory generation and training mechanics

METHODOLOGY

TPT Framework Architecture

Thinking Trajectory Generation Process

Given a document **d** from the pre-training dataset, a thinking trajectory **t** is generated using an o ff-the-shelf model with a specialized prompt, where the placeholder **{{CONTEXT}}** is replaced b y the document text.

```
# Prompt Template:
{{CONTEXT}}
## End of context
Simulate an expert's in-depth thought process...
```

Data Formation

The original document and generated thinki ng trajectory are concatenated:

$$x = [d; t]$$

Augmented Training Sample

Training Objective

Minimize standard next-token prediction los s:

$$L = -1/N \; \Sigma \log p(x_i \mid x_{<i})$$

Cross-Entropy Loss

Applicability Across Training Stages

This approach is applicable across different LM training stages, including **pre-training from scratch** and **mid-tr**

Generation Parameters

Input Document Length

≤ 2,048 tokens

Max Thinking Tokens

8,192 tokens

Generation Temperature

0.6

Top-p (Nucleus Sampling)

0.9

Generation Models Used

Mid-training: DeepSeek-R1-Distill-Qwen-7B

Pre-training: Qwen3-8B

Generation Cost: ~20k A100 GPU hours for 100B training tokens

ADVANTAGES

Key Properties of TPT

Scalability

The process of thinking augmentation is **extremely simple and universally applicable** to any text data. Compared to RPT (Reinforcement Pre-Training), our method:

- ✓ **No online rollouts** required
- ✓ Operates at the **document level**
- ✓ **No human annotation** needed
- ✓ **Highly scalable** infrastructure

Dynamic Compute Allocation

Valuable tokens can be difficult to learn in a generalizable manner by training on them directly. Thinking augmentation:

- ✓ **Breaks down complex tokens** into smaller steps
- ✓ **Allocates more training compute** to challenging samples
- ✓ Analogous to **test-time scaling** but applied during training
- ✓ **Natural up-sampling** of high-value domains

LLM-Friendly Data Format

Web-crawled data are often noisy and of varying quality, necessitating extensive filtering and rewriting. TPT provides:

- ✓ **Complementary method** to existing pipelines
- ✓ Transforms raw text into **LLM-friendly format**
- ✓ **Facilitates more efficient learning**
- ✓ **No constraints** on document structure

Comparison with RPT

TPT (Ours)	RPT
✓ Document-level	✗ Token-level
✓ Offline generation	✗ Online rollouts
✓ Simple & scalable	✗ Compute intensive

TPT applies test-time scaling principles **during training**

03

Experimental Results

Comprehensive Evaluation Across Multiple Training Configurations

Validating TPT effectiveness across model sizes and training scenarios

EXPERIMENT 1

Pre-Training With Abundant Data (100B Tokens)

Experimental Setup

Two 8B parameter models trained from scratch following LLaMA-3-8B architecture wit h a total training budget of **100B tokens**. Training consists of 25k steps with a batch size of 4M tokens.

Vanilla Model	TPT Model
Trained on original dataset	Trained on thinking-augmented dataset

Training Loss

Thinking-augmented model achieves **substantially lower training loss**, suggesting augmented data is less noisy and more learnable.

Loss Reduction
Significant

5-Task Evaluation

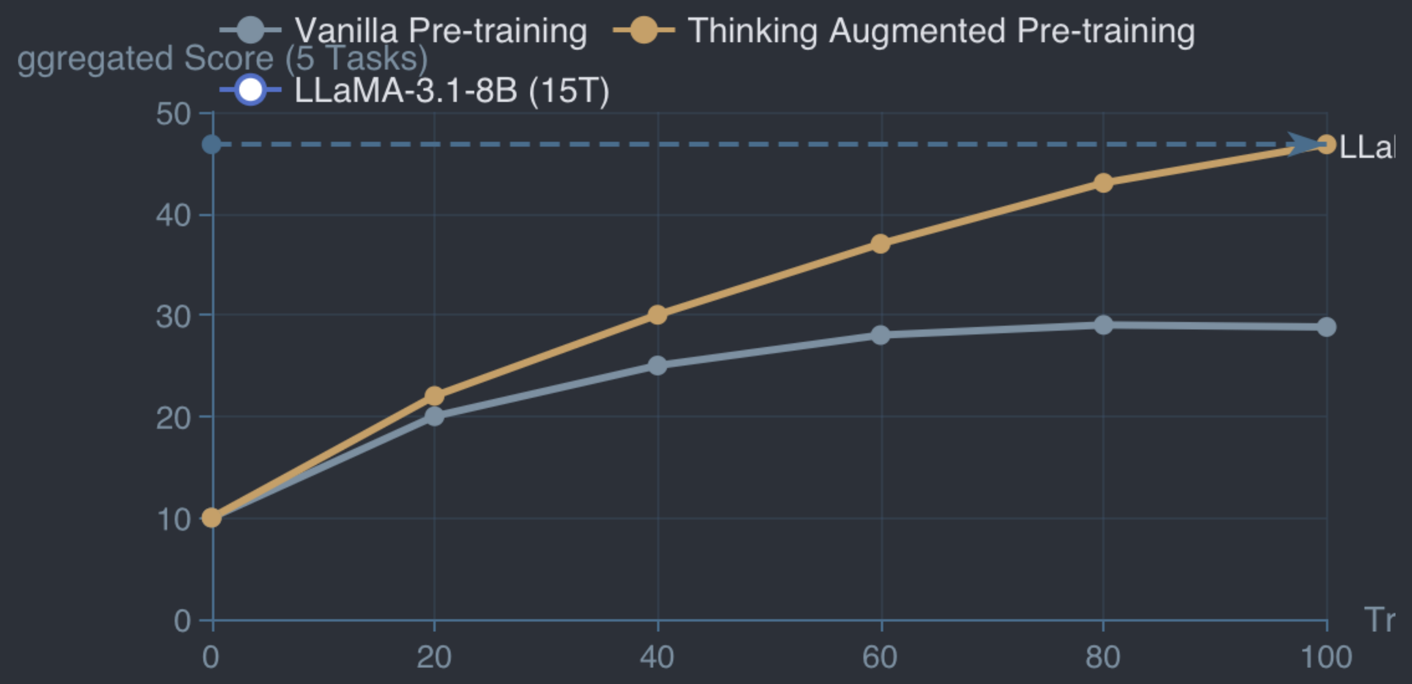
Aggregated score across: **GSM8k, MATH, BoolQ, MMLU, MMLUPro**

Gap Widening
After 20B

Key Achievement

At 100B training tokens, TPT-8B achieves performance **comparable to LLaMA-3.1-8B** trained on 15T tokens — a **150× data efficiency improvement**.

Performance Comparison



Final Results (100B Tokens)

Model	GSM8k	MATH	Avg
Vanilla-8B	26.2	9.1	28.8
TPT-8B	**50.1**	**21.8**	**46.8**
LLaMA-3.1	47.0	14.1	46.8

All scores on 5-task average. TPT-8B matches LLaMA-3.1-8B with 150× less data.

EXPERIMENT 2

Pre-Training Under Constrained Data (10B Document Tokens)

Motivation & Setup

Frontier LLM training is approaching the exhaustion of high-quality web data. This scenari o simulates data scarcity by **limiting total training tokens from raw documents to 10B** with a training budget of 40B tokens.

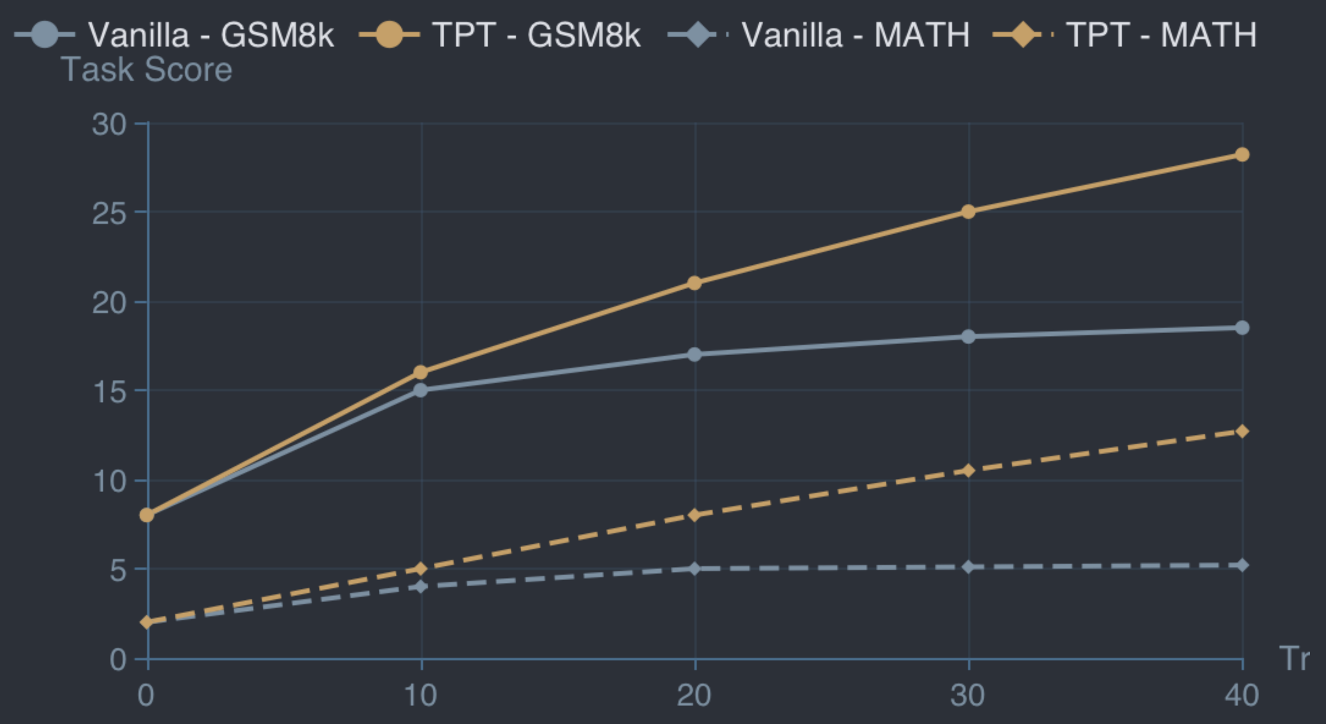
Vanilla Model	TPT Model
Sees entire dataset **4 epochs**	Sees data **once** (augmented)

Key Findings

- ✓ **Initial Phase:** Both models exhibit similar performance trajectories across all benchma rks
- ✓ **Divergence:** Vanilla performance plateaus as unique tokens are exhausted
- ✓ **TPT Sustained Growth:** Continues improving steadily, especially on mathematical reas oning
- ✓ **Data Extraction:** Enables models to extract more value from same underlying data

Critical Insight: TPT's sustained improvement suggests thinking trajectories enable models to extract more value from the same underlying data, making it particularly valuable in data-constrained scena

Performance Trajectories (Constrained Data)



Mathematical Reasoning Performance

Task	Vanilla	TPT	Improvement
GSM8k	18.5	28.2	+52%
MATH	5.2	12.7	+144%
MMLU Pro	12.1	16.8	+39%

Scores at 40B training tokens. TPT shows substantial gains in reasoning tasks

EXPERIMENT 3

Thinking Augmented Mid-Training Results

Mid-Training Methodology

Mid-training (continual pre-training) enhances existing LLMs by further training on curated datasets, circumventing the need to train from scratch.

Models Tested	Model Families	Mid-Training
3	2	100B
1.5B to 7B	Qwen2.5, LLaMA-3	tokens

Training Pipeline

1. Mid-Training
100B thinking-augmented tokens

2. SFT
Mixture-of-Thoughts (350k samples)

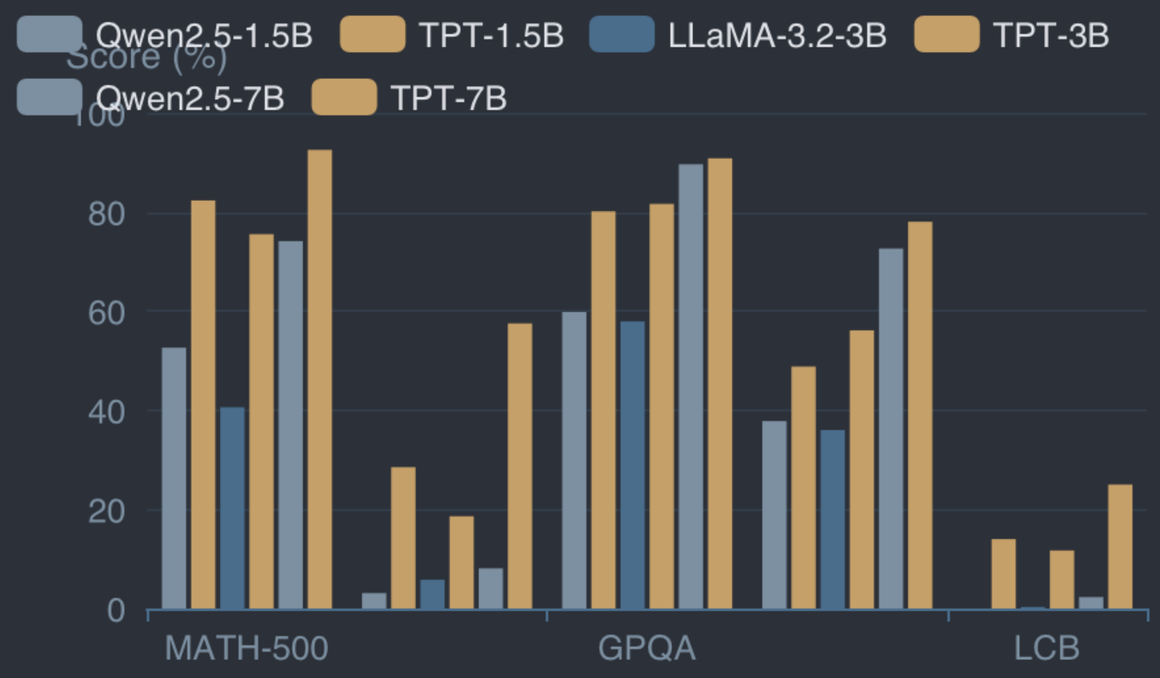
Evaluation

- ✓ **10** challenging benchmarks
- ✓ Math, code, general reasoning
- ✓ Max 32k generation length

★ Notable Achievement

Particularly pronounced improvements for **LLaMA models**, likely because their pre-training corpora contai n less reasoning-intensive data compared to Qwen2.5.

Mid-Training Performance Gains



LLaMA-3B Improvements

Benchmark	Before	After
AIME24	5.8	18.6
MATH-500	40.6	75.5
GPQA	27.7	45.2

↑ 3× improvement on AIME24 demonstrates TPT's effectiveness

POST-TRAINING EVALUATION

Supervised Fine-Tuning Performance

SFT Evaluation Setup

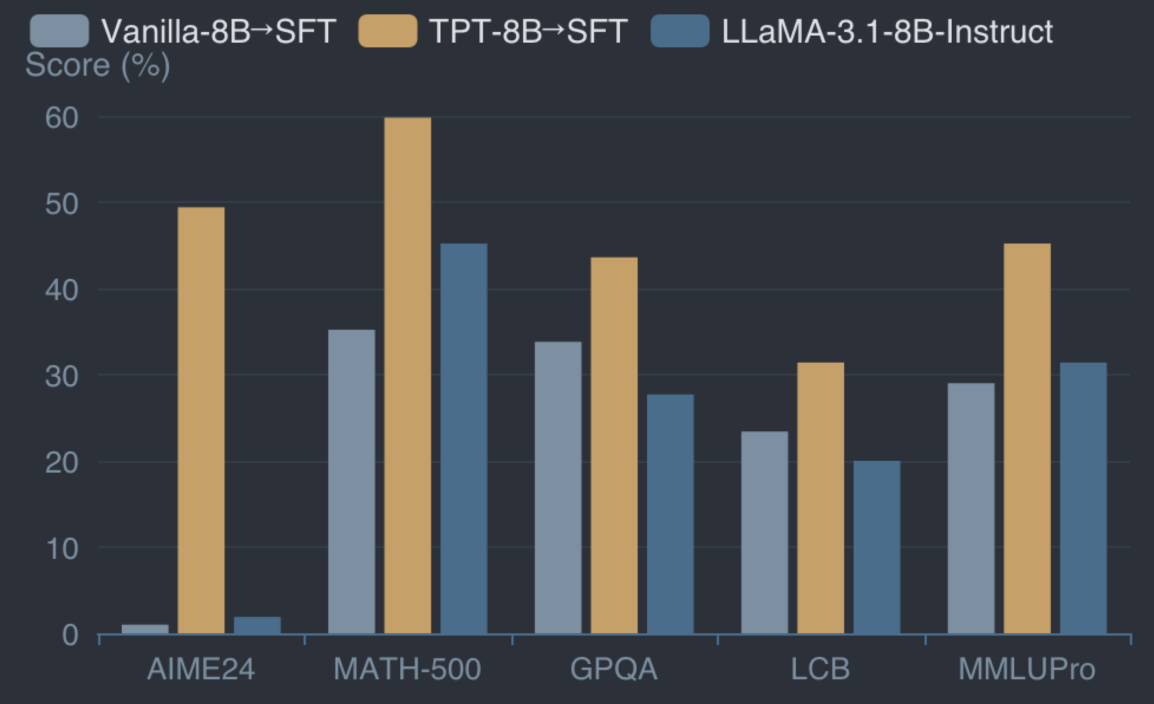
Models assessed on challenging benchmarks after SFT on the **2B-token Mixture-of-Thoughts dataset** (350k samples distilled from DeepSeek-R1).

Vanilla-8B→SFT	TPT-8B→SFT
Fails to develop strong reasoning	Substantial performance uplift

Key Findings

- **Vanilla Failure:** Very low scores on AIME24 and LiveCodeBench (LCB)
- **TPT Excellence:** Substantial performance uplift across all evaluated tasks
- **Superior Performance:** Outperforms LLaMA-3.1-8B-Instruct on every benchmark
- **Data Efficiency:** Superior reasoning with fraction of training data

Post-SFT Performance Comparison



Performance Gains (TPT vs. Vanilla)

Benchmark	Vanilla	TPT
AIME24	1.0	49.4
MATH-500	35.2	59.8
GPQA	33.8	43.6

Critical Insight

04

Analysis & Findings

Understanding Thinking Patterns and Ablation Studies

Deep dive into the characteristics and behaviors of thinking-augmented training

ANALYSIS

Thinking Pattern Analysis

Analysis Methodology

Analysis of **20k documents** from essential-web-v1.0 dataset, stratified across **three metadata groups**: domain, reasoning intensity, and target audience. Thinking trajectories generated using DeepSeek-R1-Distill-Qwen-7B.

Domains	Reasoning	Audience
Various subjects	4 intensity levels	Expertise levels

√×Domain Analysis

Domains such as **Mathematics and Physics** exhibit notably longer thinking trajectories, aligning with the expectation that these fields necessitate deep reasoning.

Math & Physics Thinking Length

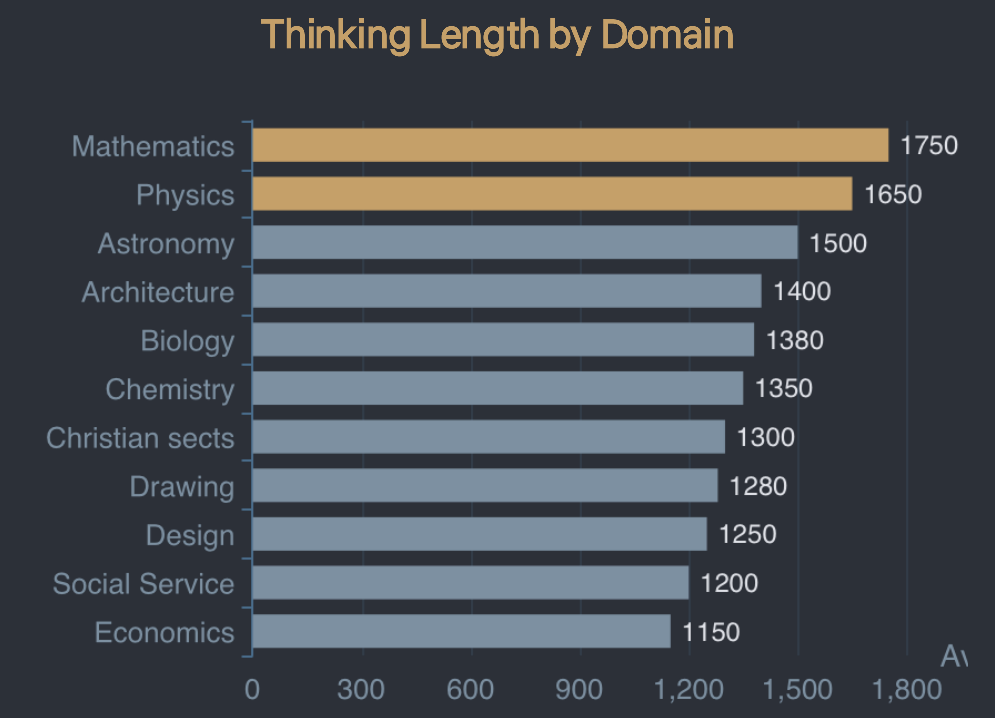
~1,600 tokens

~60% above average

Reasoning Intensity

Positive correlation between reasoning intensity and thinking length. The "Advanced Reasoning" group possesses approximately **50% more tokens** than the "No Reasoning" group.

No Reasoning	Advanced



Target Audience Analysis

Somewhat counterintuitively, for the target audience tag, the **"Expert"** group exhibits **shorter thinking trajectories** compared to the "Undergraduate" group.

Explanation: Expert-level documents often contain more specialized concepts, but do not necessarily require a greater number of reasoning steps for comprehension.

Undergraduate	Expert

ABLATION STUDY 1

Thinking Trajectory Generation Strategies

Alternative Generation Strategies

Exploring alternative strategies for generating thinking trajectories to compare against our default methodology.

1 Customized Back-thinking Model

Fine-tune DeepSeek-R1-Distill-Qwen-7B to generate thinking content within tags with final response and o riginal question as input.

2 Prompt with Random Focus Point

Modify prompt by instructing model to focus on a random point within document to generate more diverse outputs.

Results Summary

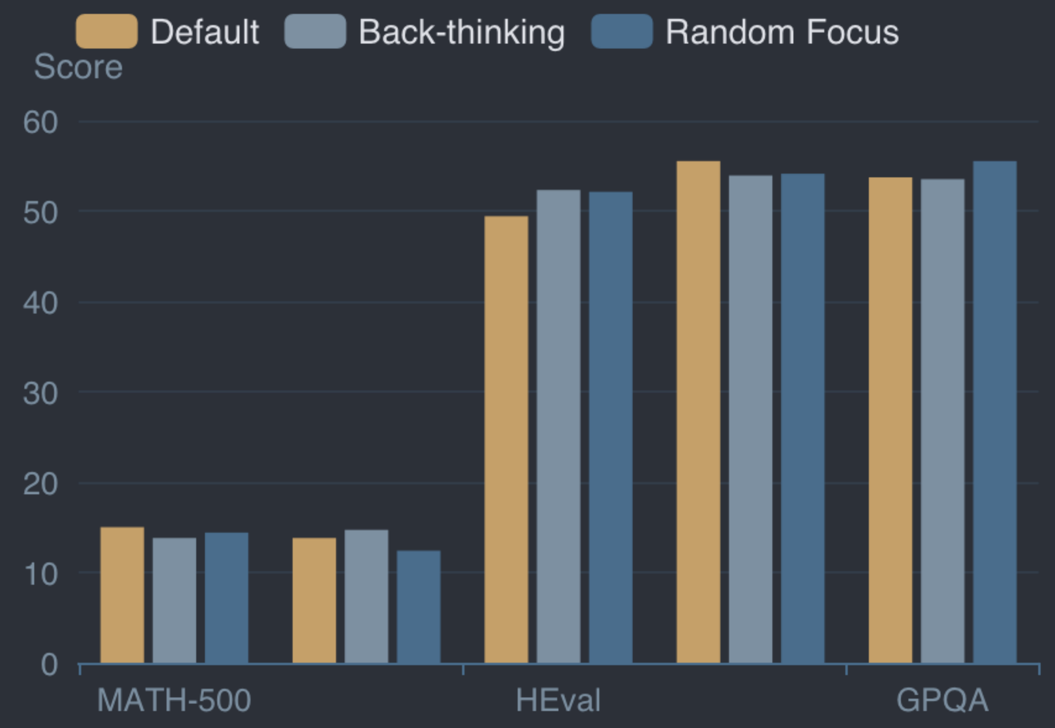
- ✓ Only **marginal gains** across most benchm arks
- ✗ **Extra implementation complexity**
- ✗ Requires **custom fine-tuning**

Conclusion

We stick to the **default strategy** for main expe riments to ensure:

- ✓ Simplicity
- ✓ Reproducibility
- ✓ Scalability

Generation Strategy Comparison



Performance Comparison

Method	Math	Code	General
Default	15.0	18.7	36.0
Back-thinking	13.8	18.4	41.7
Random Focus	14.4	18.9	36.7

All models: 40B tokens mid-training + SFT

ABLATION STUDY 2

Scaling Thinking Generation Model & Budget Impact

Scaling Thinking Generation Model

Perhaps surprisingly, using a **smaller model for thinking generation outperforms t he larger model**.

Default Model	Smaller Model
7B	1.5B
DS-Distill-Qwen-7B	DS-Distill-Qwen-1.5B

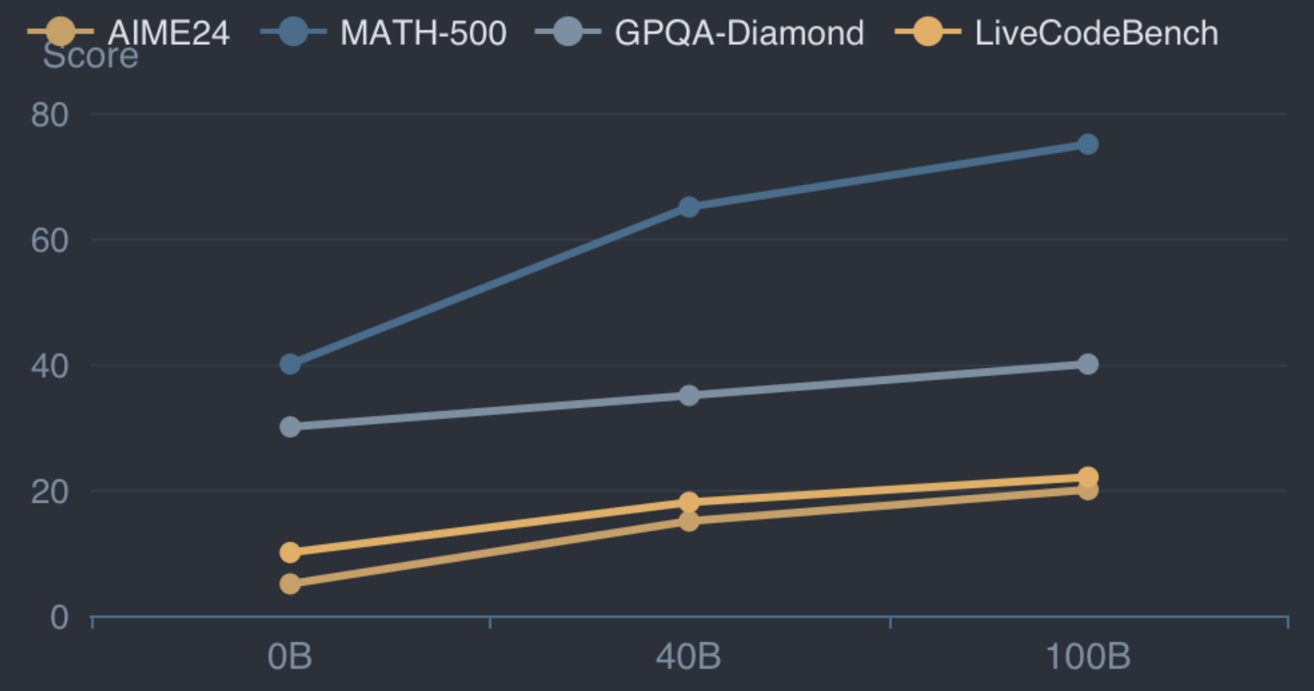
Finding: Smaller model may generate trajectories better suited for downstream mod el learning. The relationship warrants further investigation.

Impact of Mid-Training Token Budget

SFT with 350k samples proves **insufficient for developing strong reasoning capab ilities**.

- ✗ **0B (Direct SFT):** Barely solves any AIME24 problems
- ✓ **100B tokens:** ~15-point performance increase on AIME24
- ↑ **Continual gains:** Scaling beyond 100B likely yields further improvements

Mid-Training Budget Impact



Impact of SFT Data Size

- ✓ Increasing SFT epochs **improves performance** across most benchmarks
- ✓ **No serious overfitting** observed even at 5 epochs
- ★ Mid-trained checkpoints demonstrate **superior starting points** that persist through SFT

05

Related Work

Positioning TPT Within the Broader Research Landscape

Contextualizing our contributions within existing literature

RELATED WORK 1

Data Engineering & Synthetic Data Generation

Evolution of Data Engineering

A core contributing factor to the success of large foundation models is the curation of large-scale, high-quality training data. The scaling laws suggest model performance can be significantly improved by increasing dataset size alongside model size.

GPT-3 (2020)		Modern LLMs
300B tokens	→	**10T+** tokens

Modern Data Curation Pipeline

Complex, multi-stage process transforming raw data into high-quality corpus:

- ✓ Text extraction
- ✓ Deduplication
- ✓ Heuristic filtering
- ✓ Model-based filtering
- ✓ Domain balancing
- ✓ Rewriting/Paraphrasing

Exhaustion of Human-Authored Data

The community is moving towards **exhausting high-quality human-authored data** on the web. As a result:

- → **Synthetic data generation** has emerged as promising approach
- → Critical for both **pre-training and post-training**
- → Phi series heavily relies on **textbook-like synthetic data**

TPT vs. Related Approaches

Reasoning CPT

Mines hidden thoughts using non-thinking LLM. Limited to **~150M tokens** with evaluation limited to base models.

BoLT

Bootstraps latent thoughts using EM algorithm. Limited scope compared to TPT.

TPT (Ours)

Scales to **100B tokens** for both pre-training and mid-training with significant improvements on wide range of benchmarks.

Key Distinction: TPT augments existing datasets with detailed thinking trajectories, orthogonal to rewriting-based approaches

RELATED WORK 2

Chain-of-Thought Reasoning & Test-Time Scaling

Chain-of-Thought (CoT) Reasoning

CoT enables LLMs to generate intermediate steps for solving complex problems, thereby eliciting their reasoning capabilities at the cost of increased inference time.

Initial Studies (Wei et al., 2022)

Simply encouraging a step-by-step process dramatically improves performance on reasoning tasks.

Advanced Structures

Research moved beyond linear chains to sophisticated tree structures (Yao et al., 2023) for exploration, backtracking, and self-correction.

Test-Time Scaling

Instead of solely relying on prompting, recent approaches fine-tune LLMs with reinforcement learning to explicitly encourage generation of long thinking trajectories:

OpenAI o1
RL fine-tuning for long thinking

DeepSeek-R1
Incentivizing reasoning capability

Test-Time Scaling Phenomenon

These methods demonstrate substantial performance improvements on Olympiad-level math and coding problems, observing a **positive correlation** between generated token length and task performance.

Key Observation

More tokens → Better performance

Longer thinking during inference leads to improved accuracy

TPT's Key Distinction

We leverage open-source LLMs to generate thinking trajectories for **augmenting training data**. Our key innovation:

Training-Time Scaling

Apply test-time scaling principles **during training**, allocating more compute to challenging samples to enhance learnability.

Paradigm Shift: Instead of generating longer thoughts at inference, TPT trains on pre-generated thinking trajectories

CONCLUSION

Conclusion & Future Directions

TPT: A Simple and Scalable Approach

We introduce **Thinking Augmented Pre-Training (TPT)**, a simple and scalable approach to enhance pre-training data efficiency by augmenting existing text data with thinking trajectories.

3×	100B	10%+
Data Efficiency Token reduction	Tokens Evaluated	Performance Gains

Key Achievements

- ✓ **Consistent gains** across different model sizes and training configurations
- ✓ **Notable improvements** in reasoning-intensive tasks
- ✓ **Natural up-sampling** of high-value data without manual heuristics
- ✓ **Dynamic allocation** of training compute based on difficulty

Future Research Directions

- → **Scale to larger corpora and model sizes** (beyond 100B tokens)
- → **Integrate automatic prompt optimization** techniques
- → **Explore more powerful thinking generation models**

Call to Action

We hope our findings will inspire **continued research into scalable data engineering** that maximizes foundation model potential while making more efficient use of data.

> "The future of AI lies not just in scaling compute, but in intelligently augmenting data to maximize learning efficiency.

Research Impact Summary

Model Sizes	1.5B - 8B
Training Tokens	Up to 100B
Benchmarks	15+ Datasets