Beyond Truncation: Rethinking LLM Decoding as Ensemble Pruning

Dunyao Xue[1], Chengshuo Du[1], Zhengbo Wang[1], Wenlin Dai[2], Cheng Meng[2]

[1]Institute of Statistics and Big Data, Renmin University of China, Beijing, China
[2]Center for Applied Statistics, Institute of Statistics and Big Data, Renmin University of China, Beijing, China
Correspondence: chengmeng@ruc.edu.cn; wenlin.dai@ruc.edu.cn

Abstract

We introduce Mahalanobis-Ensemble Decoding (ME-Decoding), a novel Large Language Model (LLM) decoding framework that frames candidate token selection as ensemble pruning. Existing selection strategies rely predominantly on scalar probabilities, ignoring geometric semantic relationships and causing candidate redundancy. Meanwhile, current geometry-aware methods often require complex optimization or directly reweighting the original token probabilities, leading to significant computational overhead or inference instability. To address this, we formulate decoding as a subset optimization problem using a Mahalanobis distance-driven objective to enhance semantic diversity while preserving high probabilities. Specifically, we dynamically discount redundant generation paths using a token similarity matrix, constructed via an adaptive-bandwidth kernel over token embeddings. We further devise an efficient greedy selection algorithm with near-linear complexity in the candidate size under early stopping, while establishing its theoretical approximation guarantees. This renders ME-Decoding a robust, plug-and-play module with negligible inference overhead. Extensive experiments across diverse reasoning and generation tasks demonstrate that our method consistently achieves strong performance across models, temperatures, and benchmarks.

1 Introduction

Large Language Models (LLMs) have demonstrated strong capabilities in mathematical reasoning, instruction following, and open-ended generation (Brown et al., 2020; Touvron et al., 2023; Chowdhery et al., 2023; Achiam et al., 2023). Besides model scaling and post-training alignment, the inference-time decoding strategy also plays a crucial role in determining generation quality. At each step, an autoregressive LLM produces a next-token distribution, from which the decoder selects or samples the next token. Deterministic methods such as greedy decoding and beam search (Holtzman et al., 2020) favor high-probability continuations but often produce repetitive or overly conservative outputs. Sampling-based methods introduce stochasticity and improve generation diversity, but they may also assign non-negligible probability to low-quality tokens, increasing inference uncertainty and potentially leading to illogical continuations or hallucinations.

To control this quality–diversity trade-off, many decoding methods perform probability-based truncation and reshaping of the next-token distribution. Classical approaches such as Top-k and nucleus sampling restrict sampling to high-probability tokens (Fan et al., 2018; Holtzman et al., 2020), while later methods such as Min-p, entropy-aware sampling, and p-less adapt the truncation rule according to model confidence or distributional statistics (Hewitt et al., 2022; Nguyen et al., 2025; Tan et al., 2025). Despite their effectiveness, these methods mainly operate on token probabilities and treat candidates as independent categorical outcomes. As a result, they overlook the semantic geometry among tokens, which may retain redundant candidates and limit the effectiveness of the selected sampling support (Yang et al., 2026).

Table 1: Comparison of different decoding methods.

Method	Decoding Strategy	Geometry Aware[1]	Adaptive Selection	Redundancy Control[2]
Greedy	Determin.	✗	✗	✗
Top-k	Prob. Trunc.	✗	✗	✗
Min-p	Prob. Trunc.	✗	✗	✗
p-less	Prob. Trunc.	✗	✓	✗
Top-W	Dist. Match.	✓	✓	✗
CraEG	Reweighting	✓	✗	✓
Ours	**Ensemble**	✓	✓	✓

[1] **Embedding Geometry**: Uses token embedding geometry in the decoding process.
[2] **Redundancy Control**: Explicitly controls redundancy among candidate tokens.

Although recent geometry-aware methods incorporate token-space structure, they continue to face significant challenges. For example, Top-W formulates a Wasserstein-regularized distribution matching problem (Davoodi et al., 2026), which requires complex optimization and meticulous tuning, and it does not explicitly address redundancy among tokens. Alternatively, CraEG penalizes tokens in crowded regions via a lightweight reweighting mechanism (Yang et al., 2026). However, directly modifying these probabilities distorts the original model distribution, leading to inference instability, and the approach still relies on auxiliary methods for post-processing. These limitations motivate our key question:

How can we design a simple and efficient framework that selects a compact yet informative token set while preserving high-confidence candidates?

In this work, we address this question from the perspective of ensemble pruning, a paradigm that entails selecting a compact subset of learners from a larger pool to optimize predictive outcomes. This paradigm closely mirrors the token selection problem in LLM decoding. Extensive research in ensemble learning shows that effective ensembles should balance individual accuracy with collective complementarity (Krogh and Vedelsby, 1994; Opitz and Maclin, 1999). Consequently, simply selecting the highest-scoring learners may introduce redundancy and hurt overall performance (Zhou et al., 2002; Li et al., 2012). Similarly, LLM decoding encounters an analogous challenge of token redundancy. Building on this insight, we introduce ME-Decoding, a novel framework that incorporates token geometry to reformulate decoding as a subset maximization problem driven by the Mahalanobis distance. By approximately solving this objective with an efficient greedy procedure, ME-Decoding mitigates redundancy within the selected token set during generation with negligible inference overhead, thereby improving token selection quality while maintaining reasoning performance.

Our contributions are summarized as follows:

- We reformulate LLM decoding through the novel lens of ensemble pruning.
- We define a Mahalanobis distance-driven objective to adaptively select a compact, informative candidate set while preserving token probabilities.
- We devise an efficient greedy selection algorithm with candidate-linear complexity under early stopping, while establishing its trajectory unimodality and theoretical approximation guarantees.
- Extensive experiments across diverse reasoning and generation tasks show that our method consistently outperforms strong existing baselines.

2 Motivation

2.1 Background of LLM Decoding

Large Language Models (LLMs) generate text autoregressively by predicting the next token from the preceding context. At each step, an LLM outputs a probability distribution $\mathcal{P} \in \Delta^{|\Omega|-1}$ over the full vocabulary Ω. To avoid sampling from the unreliable long tail, modern decoding methods usually first form a candidate pool $\mathcal{V}_t \subseteq \Omega$, then select a final subset $S \subseteq \mathcal{V}_t$ and sample from the renormalized distribution:

$$\boldsymbol{p}_S(i) = \frac{p_i \mathbf{1}\{i \in S\}}{\sum_{j \in S} p_j}.$$

This process can be viewed as a subset selection problem aiming to preserve the original distribution under a probability-based criterion:

$$S^\star = \arg\min_{S \subseteq \mathcal{V}_t} f(\boldsymbol{p}_S),$$
$$\text{s.t.} \quad S = \{i : p_i \geq \tau(\mathcal{P})\},$$

where $f(\cdot)$ denotes a generalized decoding objective and $\tau(\mathcal{P})$ is a probability-based truncation threshold.

Most prevailing decoding methods follow this probability-driven paradigm. Top-k and nucleus sampling truncate the distribution using fixed probability or cumulative-mass thresholds, while adaptive methods such as p-less and entropy-aware sampling adjust the threshold based on distributional statistics. Despite different truncation rules, these methods remain probability-centric and treat candidate tokens as independent outputs. They therefore overlook semantic dependencies among tokens, motivating us to incorporate inter-token semantics into the subset selection objective.

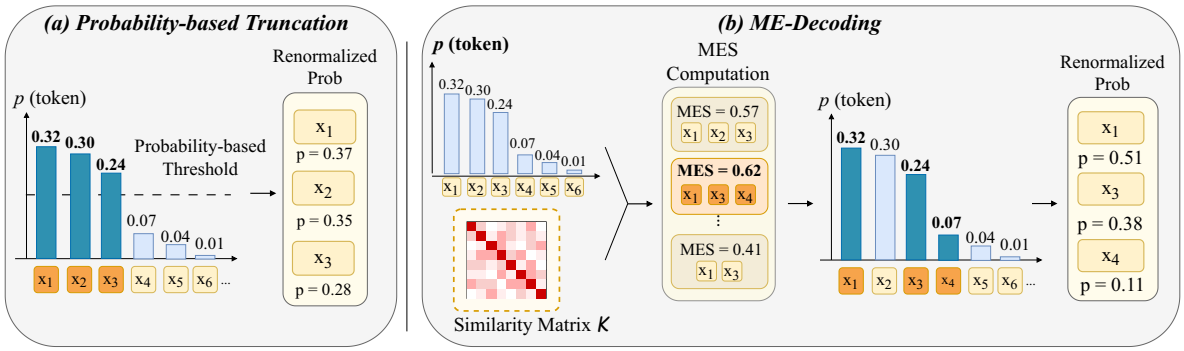

Figure 1: Comparison between probability-based truncation methods (left) and our ME-Decoding framework (right). ME-Decoding extends standard probability-only token selection by incorporating embedding-based similarity into the MES score and selecting a compact geometry-aware subset for sampling.

2.2 Connection to Ensemble Pruning

Ensemble pruning aims to select a compact sub-ensemble that preserves or improves the predictive performance of the full ensemble. Classical error analysis (Breiman, 2001) characterizes ensemble performance as a trade-off between individual accuracy and inter-model diversity. Following this idea, Zhang et al. (2006) formulate pruning as a quadratic subset-selection problem. Given a binary error indicator matrix $\boldsymbol{E}$, they construct the error co-occurrence matrix $\boldsymbol{C} = \boldsymbol{E}^\top \boldsymbol{E}$, where diagonal entries measure individual errors and off-diagonal entries measure shared failures. After normalization, the pruning objective can be written as

$$\min_{\boldsymbol{s}} \ \boldsymbol{s}^\top \widehat{\boldsymbol{C}} \boldsymbol{s} \quad \text{s.t.} \sum_i s_i = N, \ s_i \in \{0, 1\}. \quad (1)$$

This objective selects a size-N sub-ensemble by jointly penalizing individual inaccuracy and pairwise redundancy.

This paradigm naturally parallels token selection in LLM decoding. Candidate tokens can be viewed as ensemble members: their probabilities measure individual confidence, while their embedding-space similarities measure semantic redundancy. Therefore, an ideal decoding subset should retain high-likelihood tokens while suppressing redundant candidates, mirroring the accuracy–diversity trade-off in ensemble pruning.

2.3 Generalizing Ensemble Pruning to LLM Decoding via Mahalanobis Distance

The quadratic objective in Eq. 1 leverages second-order statistics to penalize redundant candidates. Motivated by this, we aim to extend this diversity-aware framework to LLM decoding. To achieve this, we observe that for any representation vector $\boldsymbol{p}$ and a symmetric positive definite matrix, this quadratic structure naturally corresponds to the Mahalanobis distance.

In machine learning, the Mahalanobis distance is widely used to construct discriminative representations, such as in metric learning, out-of-distribution detection, and representation regularization (Weinberger and Saul, 2009; Lee et al., 2018; Wan et al., 2018; Kang et al., 2026). Formally, given two vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$ and a symmetric positive definite matrix $\boldsymbol{M} \succ 0$, it is defined as

$$\mathcal{D}_M(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{M}) := \sqrt{(\boldsymbol{x} - \boldsymbol{y})^\top \boldsymbol{M}^{-1} (\boldsymbol{x} - \boldsymbol{y})}.$$

In particular, setting $\boldsymbol{y} = \boldsymbol{0}$ and letting $\boldsymbol{M}$ be the token similarity matrix $\boldsymbol{K}$ gives the Mahalanobis norm

$$\|\boldsymbol{x}\|_{\boldsymbol{K}} := \sqrt{\boldsymbol{x}^\top \boldsymbol{K}^{-1} \boldsymbol{x}}.$$

Mathematically, by explicitly incorporating the inverse matrix $\boldsymbol{K}^{-1}$, the Mahalanobis norm inherently downweights correlated directions and separates information along non-redundant axes. This geometric property aligns perfectly with our overarching goal: to suppress repetitive generation paths and enhance token diversity during decoding.

3 Method: ME-Decoding

In this section, we propose Mahalanobis-Ensemble Decoding (ME-Decoding), a geometry-aware decoding framework inspired by ensemble pruning. ME-Decoding selects a compact token subset by optimizing a Mahalanobis-style objective that combines token probabilities with embedding-based similarities. The overall structure of ME-Decoding is illustrated on the right side of Figure 1.

3.1 Mahalanobis-Ensemble Score.

Inspired by ensemble pruning, we view the candidate tokens at each decoding step as a pool of weak semantic predictors. Instead of retaining a fixed number of high-probability tokens, our goal is to construct a compact token ensemble that preserves high-confidence candidates while reducing redundancy in the token embedding space. At each decoding step, ME-Decoding is applied to a probability-based candidate pool $\mathcal{V}_t \subseteq \Omega$; for simplicity, we omit t and write $\mathcal{V}$ hereafter.

We first define the *Mahalanobis-Ensemble Energy* (MEE) for a selected token subset $\mathcal{S} \subseteq \mathcal{V}$ as

$$\mathrm{MEE}(\mathcal{S}) = \boldsymbol{p}_{\mathcal{S}}^{\top} \boldsymbol{K}_{\mathcal{S}}^{-1} \boldsymbol{p}_{\mathcal{S}},$$

where $\boldsymbol{p}_{\mathcal{S}}$ denotes the vector of token probabilities indexed by $\mathcal{S}$, and $\boldsymbol{K}_{\mathcal{S}}$ is the token similarity matrix restricted to $\mathcal{S}$. The inverse matrix $\boldsymbol{K}_{\mathcal{S}}^{-1}$ discounts redundant tokens and rewards subsets with high collective quality.

However, unlike standard ensemble pruning, the optimal subset size in decoding is unknown a priori. This requires us to maximize the MEE of the selected tokens while avoiding unnecessarily large subsets, thereby filtering out redundant tokens that contribute only marginally to the overall MEE. To this end, we define the *Mahalanobis-Ensemble Score* (MES) as follows:

$$\mathrm{MES}_{\lambda}(\mathcal{S}) = \frac{\mathrm{MEE}(\mathcal{S})}{\left[1 + \lambda H_2^T(\boldsymbol{p})\right]^{|\mathcal{S}|}},$$

where $H_2^T(p) = 1 - \sum_{i \in \mathcal{V}} p_i^2$ is the order-2 Tsallis entropy measuring the uncertainty of the next-token distribution, and $\lambda > 0$ controls the entropy-dependent size penalty.

Intuitively, the numerator rewards tokens with large non-redundant contributions, while the denominator discourages excessive subset expansion. When the distribution is flat, the unregularized MEE may keep increasing by accumulating weak marginal gains from many uncertain candidates. The entropy-dependent penalty raises the inclusion threshold, retaining only tokens with sufficiently large non-redundant contributions. Consequently, the decoding objective is given by:

$$\mathcal{S}^{\star} = \arg\max_{\mathcal{S} \subseteq \mathcal{V}} \mathrm{MES}_{\lambda}(\mathcal{S}). \tag{2}$$

By optimizing the MES, we can effectively extract a highly representative token subset from the candidate distribution.

3.2 Construction of Similarity Matrix

To measure token similarity in the embedding space while maintaining numerical stability, we construct $\boldsymbol{K}$ using a Gaussian kernel on normalized token embeddings. Let $\boldsymbol{e}_i \in \mathbb{R}^d$ denote the normalized embedding of token i. We define

$$C_{ij} = 1 - \boldsymbol{e}_i^{\top} \boldsymbol{e}_j, \qquad K_{ij} = \exp\{-C_{ij}/\epsilon\}. \tag{3}$$

Here, C_{ij} measures the semantic distance between tokens, and $\epsilon > 0$ is a bandwidth parameter controlling the smoothness of the kernel.

In our implementation, the bandwidth ϵ is chosen adaptively according to the probability-weighted semantic dispersion of the candidate tokens:

$$\epsilon = \frac{1}{2} \sum_{i,j \in \mathcal{V}} p_i p_j C_{ij},$$

where $\sum_{i,j \in \mathcal{V}} p_i p_j C_{ij}$ is the expected pairwise semantic distance between tokens sampled from the candidate distribution. When the candidate tokens are semantically concentrated, the adaptive bandwidth becomes smaller, yielding a localized kernel and making the selection more probability-driven. Conversely, when the candidates are semantically dispersed, the bandwidth becomes larger, preserving semantic relations over a broader range and making the selection more geometry-aware. Thus, the adaptive bandwidth balances probability and semantic structure according to the local geometry of the decoding distribution.

3.3 Optimization Strategy

Directly solving the maximization problem formulated in Eq. 2 is an NP-hard combinatorial task. To maintain computational feasibility, we use a greedy forward-selection procedure and maintain the inverse Cholesky factor of $\boldsymbol{K}_{\mathcal{S}}^{-1}$ incrementally (Golub and Van Loan, 2013), which avoids repeated matrix inversions during subset construction.

The greedy selection in Algorithm 1 is conceptually related to Orthogonal Matching Pursuit (OMP) (Pati et al., 1993). At each step, $(p_j - \boldsymbol{\alpha}_j^{\top} z)$ represents the conditional residual score of token j after accounting for its correlation with the selected tokens, while r_j normalizes it by the corresponding conditional variance. The marginal contribution is therefore measured by $\left[r_j(p_j - \boldsymbol{\alpha}_j^{\top} z)\right]^2$. The algorithm selects the token with the largest contribution and accepts it only when the penalized MES

Algorithm 1 Greedy Algorithm for ME-Decoding

Input: Candidate token pool $\mathcal{V}$ with $N = |\mathcal{V}|$; candidate token scores $\boldsymbol{p} \in \mathbb{R}^N$; normalized candidate token embeddings $\boldsymbol{E} = [\boldsymbol{e}_1, \ldots, \boldsymbol{e}_N]^\top \in \mathbb{R}^{N \times d}$; MES penalty λ.
Initialize $\mathcal{S} = \varnothing$.
Set $c_\lambda \leftarrow 1 + \lambda H_2^T(\boldsymbol{p})$.
Select the first token $j_1 = \arg\max_i p_i$ and update $\mathcal{S} \leftarrow \{j_1\}$.
Compute $\boldsymbol{K}_{\mathcal{S},\mathcal{S}}$ using Eq. 3.
$\boldsymbol{R} \leftarrow (\boldsymbol{K}_{\mathcal{S},\mathcal{S}}^{1/2})^{-1}, \quad \boldsymbol{z} \leftarrow \boldsymbol{R}\boldsymbol{p}_{\mathcal{S}}$.
$\mathrm{MES}^g \leftarrow \|\boldsymbol{z}\|_2^2 / c_\lambda^{|\mathcal{S}|}$.
for $t = 1$ **to** $N - 1$ **do**
  for $j \in \{1, \ldots, N\} \setminus \mathcal{S}$ **do**
    Compute $\boldsymbol{\beta}_j \leftarrow [K_{ij}]_{i \in \mathcal{S}}$ using Eq. 3, and set $b_j \leftarrow \boldsymbol{K}_{jj}$.
    $\boldsymbol{\alpha}_j \leftarrow \boldsymbol{R}\boldsymbol{\beta}_j, \quad r_j \leftarrow (b_j - \|\boldsymbol{\alpha}_j\|_2^2)^{-1/2}$.
    Compute the MEE after adding token j:
    $c_j \leftarrow \|\boldsymbol{z}\|_2^2 + \left[r_j(p_j - \boldsymbol{\alpha}_j^\top \boldsymbol{z})\right]^2$.
  end for
  Select $j_\star = \arg\max_{j \notin \mathcal{S}} c_j$, let $c^\star = c_{j_\star}$.
  Compute the candidate MES score:
  $\mathrm{MES}_{\mathrm{cand}}^g \leftarrow c^\star / c_\lambda^{|\mathcal{S}|+1}$.
  if $\mathrm{MES}_{\mathrm{cand}}^g \le \mathrm{MES}^g$ **then**
    break
  end if
  $\boldsymbol{\gamma} \leftarrow -r_{j_\star} \boldsymbol{R}^\top \boldsymbol{\alpha}_{j_\star}$.
  $v_{j_\star} \leftarrow r_{j_\star}\left(p_{j_\star} - \boldsymbol{\alpha}_{j_\star}^\top \boldsymbol{z}\right)$.
  Update $\boldsymbol{R}$ and $\boldsymbol{z}$:

$$\boldsymbol{R} \leftarrow \begin{bmatrix} \boldsymbol{R} & \boldsymbol{0} \\ \boldsymbol{\gamma}^\top & r_{j_\star} \end{bmatrix}, \quad \boldsymbol{z} \leftarrow \begin{bmatrix} \boldsymbol{z} \\ v_{j_\star} \end{bmatrix}.$$

  $\mathcal{S} \leftarrow \mathcal{S} \cup \{j_\star\}, \quad \mathrm{MES}^g \leftarrow \mathrm{MES}_{\mathrm{cand}}^g$.
end for
Output: Selected token set $\mathcal{S}$.

objective improves, favoring high-probability tokens while discounting redundant candidates.

3.4 Theoretical Properties

Furthermore, to rigorously validate the effectiveness of our proposed strategy, we establish theoretical properties of the greedy algorithm, showing its unimodality along the search path and deriving an approximation bound to the global optimum.

The following theorem states that, under a conditioning assumption on the token-similarity matrix, the greedy MES sequence cannot increase again once it stops increasing.

Theorem 3.1 (Greedy Unimodality of Mahalanobis-Ensemble Decoding)**.** *Let $|\mathcal{V}| = N$ and $\mathrm{MES}_t^g = \mathrm{MES}_\lambda(\mathcal{S}_t)$, where $\{\mathcal{S}_t\}_{t=0}^N$ is the greedy path generated by Algorithm 1. For any $T \subseteq \mathcal{V}$, let $\boldsymbol{K}_T = (K_{ij})_{i,j \in T}$ and define $\kappa_N(\boldsymbol{K}) = \max_{T \subseteq \mathcal{V},\, |T| \le N} \frac{\lambda_{\max}(\boldsymbol{K}_T)}{\lambda_{\min}(\boldsymbol{K}_T)}$. If $\kappa_N(\boldsymbol{K}) \le 1 + \lambda H_2^T(p)$, once for some $t < N$,*

$$\mathrm{MES}_{t+1}^g \le \mathrm{MES}_t^g,$$

then for all $s \ge t$, $\mathrm{MES}_{s+1}^g \le \mathrm{MES}_s^g$. Consequently, with $\tau = \min\{t : \mathrm{MES}_{t+1}^g \le \mathrm{MES}_t^g\}$, we have

$$\mathrm{MES}_\tau^g = \max_{0 \le r \le N} \mathrm{MES}_r^g.$$

Theorem 3.1 justifies the early stopping rule in Algorithm 1: the greedy search can terminate once the MES score drops, without traversing all candidates. The next theorem compares the stopped greedy value with the global optimum, showing that it provides an effective approximation to the optimal subset.

Theorem 3.2 (Approximation Guarantee for Mahalanobis-Ensemble Decoding)**.** *Let*

$$\mathrm{MES}_\lambda^\star = \max_{\mathcal{S} \subseteq \mathcal{V}} \mathrm{MES}_\lambda(\mathcal{S}), \qquad |\mathcal{V}| = N.$$

Let $\{\mathcal{S}_t\}_{t=0}^N$ be the greedy path generated by Algorithm 1, and let τ be the stopping time defined in Theorem 3.1. Under the assumptions of Theorem 3.1, we have

$$\mathrm{MES}_\tau^g \ge (1 - \exp\{-\lambda_{\min}(\boldsymbol{K}, N)\})\, \mathrm{MES}_\lambda^\star,$$

where $\lambda_{\min}(\boldsymbol{K}, r) \triangleq \min_{T \subseteq \mathcal{V},\, |T| = r} \lambda_{\min}(\boldsymbol{K}_T)$ denotes the smallest eigenvalue among all $r \times r$ principal submatrices of $\boldsymbol{K}$.

Theorem 3.2 establishes the relation between the optimal greedy value and the globally optimal MES value. The approximation factor depends on the restricted minimum eigenvalue of $\boldsymbol{K}$, which reflects the non-redundancy and conditioning of the selected token representations.

4 Experiments

4.1 Experimental Details

Models and Baselines. We evaluate ME-Decoding on three instruction-tuned language models: Qwen3-4B-Instruct (Yang et al., 2025), Phi-4-mini-Instruct (Abouelenin et al., 2025), and

Method	Qwen3-4B-Inst.			Phi-4-mini-Inst.			Mistral-7B-Inst.			Avg.
	$T = 1.0$	$T = 1.5$	$T = 2.0$	$T = 1.0$	$T = 1.5$	$T = 2.0$	$T = 1.0$	$T = 1.5$	$T = 2.0$	
Min-p	70.74	62.02	54.06	79.68	70.81	30.86	48.67	36.32	17.97	52.35
Top-p	68.01	55.19	21.00	80.06	11.75	0.30	48.90	23.05	0.45	34.30
p-less	78.92	72.71	63.15	84.05	83.09	69.52	54.06	50.80	46.47	66.97
Top-H	76.80	67.17	58.53	81.65	73.77	34.65	52.01	46.78	22.52	57.10
Top-W	77.71	76.65	74.98	82.64	83.24	82.18	53.98	53.15	51.02	70.62
Ours	**80.67**	**79.76**	**80.06**	**84.08**	**84.23**	**82.41**	**55.34**	**53.45**	**53.90**	**72.66**

Table 2: GSM8K accuracy (%) across different temperatures and decoding methods. The Avg. column reports the average accuracy over all models and temperatures. The best result is in **bold** and the second best is underlined.

Method	Qwen3-4B-Inst.			Phi-4-mini-Inst.			Mistral-7B-Inst.			Avg.
	$T = 1.0$	$T = 1.5$	$T = 2.0$	$T = 1.0$	$T = 1.5$	$T = 2.0$	$T = 1.0$	$T = 1.5$	$T = 2.0$	
Min-p	34.38	32.59	35.04	27.01	29.02	26.56	26.12	28.12	25.22	29.34
Top-p	34.60	35.49	30.58	31.25	27.01	15.40	25.89	26.41	14.06	26.74
p-less	35.27	34.82	35.49	33.26	33.26	27.90	27.68	27.01	24.78	31.08
Top-H	34.38	33.93	34.15	**34.38**	33.93	29.91	28.12	27.23	23.88	31.10
Top-W	34.82	**36.83**	35.27	31.70	29.46	31.47	27.23	**29.02**	28.23	31.56
Ours	**35.49**	35.49	**35.71**	**34.38**	**36.16**	**33.93**	**28.35**	28.79	**28.35**	**32.96**

Table 3: GPQA accuracy (%) across different temperatures and decoding methods. The Avg. column reports the average accuracy over all models and temperatures. The best result is in **bold** and the second best is underlined.

Mistral-7B-Instruct (Jiang et al., 2023). We compare our method with several representative decoding methods, including Min-p (Nguyen et al., 2025), Top-p (Holtzman et al., 2020), p-less (Tan et al., 2025), Top-H (Baghaei Potraghloo et al., 2026), and Top-W (Davoodi et al., 2026). For a fair comparison, all methods are evaluated under the same prompts, temperature settings, maximum generation length, and stopping criteria. Unless otherwise specified, we evaluate all methods at temperatures $T \in \{1.0, 1.5, 2.0\}$. All reported experiments use $N = 512$. We additionally implement CraEG (Yang et al., 2026) and combine its post-softmax reweighting with p-less, following its plug-in formulation. Detailed repeated results are reported in Appendix B.1.

Benchmarks and Metrics. We consider two types of tasks. First, for reasoning benchmarks, we evaluate on GSM8K (Cobbe et al., 2021) and GPQA (Rein et al., 2024), where performance is measured by answer accuracy after extracting the final answer. These benchmarks test whether a decoding method can maintain reliable reasoning performance under different sampling temperatures. Second, for instruction-following and chat-style generation, we evaluate on AlpacaEval (Li et al., 2023) and MT-Bench (Zheng et al., 2023). For AlpacaEval, we report the candidate win-rate (%), and for MT-Bench, we report the average judge score. Since these tasks involve open-ended generation, we use DeepSeek-V4-Pro (DeepSeek-AI, 2026) as the judge to compare the generated responses under a fixed evaluation protocol. For reproducibility, AlpacaEval compares each response with a fixed reference for the same prompt, whereas MT-Bench independently scores each response on a 1–10 scale. Appendix B.5 reports a second-judge evaluation and paired prompt-level bootstrap confidence intervals.

ME-Decoding Configuration. ME-Decoding selects a compact token subset by optimizing a Mahalanobis-style objective that balances token probability and embedding geometry. For Top-W and ME-Decoding, both of which require an explicit candidate pool, we use the same top-N pool with $N = 512$. Except for the candidate-pool size, Top-W uses its default hyperparameters. Top-p, Min-p, and Top-H follow Davoodi et al. (2026), and p-less is evaluated under its original parameter-free rule. ME-Decoding uses $\lambda = 0.9$ by default, following Appendix C.1. Our code is available at https://github.com/anonymouspaper1202/ME-Decoding.

Temperature Placement. At temperature T, every method operates on the same distribution $p_T = \mathrm{softmax}(z/T)$. ME-Decoding uses p_T for candidate scoring and samples from the renormalized selected support, while each baseline applies its trun-

cation or reweighting rule to the same temperature-adjusted distribution.

4.2 Main Results

Reasoning Benchmarks. We first evaluate ME-Decoding on GSM8K and GPQA across three models and three temperatures. The results are reported in Tables 2 and 3. Overall, ME-Decoding achieves the best average performance across both datasets. Appendix B.1 reports three-seed repetitions as mean $\pm$ sample standard deviation and includes the CraEG+p-less comparison.

Figure 2 further shows the accuracy–temperature curves of different decoding methods. Compared with existing baselines, ME-Decoding maintains a more stable accuracy profile as the temperature increases. Higher temperatures enlarge the sampling space, making probability-only truncation methods more prone to noisy tokens. By incorporating token geometry, ME-Decoding better preserves high-confidence candidates while filtering out uninformative ones, leading to a better balance between exploration and reasoning reliability.

Instruction-following and Chat. We further evaluate ME-Decoding on instruction-following and chat-style generation tasks using MT-Bench and AlpacaEval. Experiments are conducted on Qwen3-4B, Phi-4-mini, and Mistral-7B under $T \in \{1.0, 1.5, 2.0\}$. Figure 3 summarizes the results. The left and middle panels report the average AlpacaEval win rate and MT-Bench score over the three models at each temperature, while the right panel presents the overall average rank across all models, temperatures, and benchmarks. Detailed numerical results are provided in the Appendix.

As shown in Figure 3, ME-Decoding achieves strong performance on both open-ended benchmarks. It obtains the best averaged AlpacaEval win rate and MT-Bench score. The overall rank comparison further shows that ME-Decoding attains the best average rank among all compared decoding methods, indicating its stable advantage across models, temperatures, and benchmarks. These results suggest that ME-Decoding is not limited to accuracy-oriented reasoning tasks, but also improves open-ended generation by selecting a compact and informative token set. The same conclusion holds under a second automatic judge. Paired prompt-level bootstrap intervals against Top-W, p-less, and Top-H are positive for both benchmarks and both judges, as reported in Appendix B.5.

4.3 Analysis

Complexity Analysis. We analyze the time complexity of ME-Decoding in Algorithm 1. Let N be the candidate-pool size, d the embedding dimension, and τ the number of selected tokens before early stopping. The kernel computation costs $\mathcal{O}(N\tau d)$, and the greedy evaluation costs $\mathcal{O}(N\tau^3)$, leading to $\mathcal{O}\big(N\tau(d+\tau^2)\big)$. Since early stopping usually yields a small τ with $\tau \ll N$, the practical cost scales nearly linearly with N and approaches $O(Nd)$ when τ is treated as a small constant. A detailed analysis is provided in Appendix C.2.

Efficiency Analysis. To compare inference-time efficiency, we use a CPU-only synthetic logits benchmark that isolates sampling and logits-processing overhead. We measure the average time for filtering logits and sampling one token under two settings: Medium with vocabulary size $32{,}000$, embedding dimension 256, and top-$N = 512$; and Large with vocabulary size $128{,}000$, embedding dimension 1024, and top-$N = 2048$. All experiments use batch size 1, one CPU thread, temperature 1.0, 50 warm-up steps, and 500 measured steps over 10 repeats. Results are reported in Table 4.

As shown in Table 4, ME-Decoding introduces moderate overhead compared with purely probability-based methods due to its use of token embeddings, but remains highly efficient. It is about $2.7\times$ faster than Top-W in the Medium setting and $7.4\times$ faster in the Large setting. In the Large setting, its overhead is also comparable to Top-p and Top-H, demonstrating that ME-Decoding can exploit token-geometry information with substantially lower cost than existing geometry-aware decoding methods. Appendix B.6 additionally reports end-to-end GPU latency and shows that model inference remains the dominant cost under the measured settings.

Diversity Analysis. We analyze the accuracy–diversity trade-off on GSM8K with Qwen3-4B. For each method, we randomly sample 500 questions and generate 10 responses per question. Accuracy is computed over all generations, and diversity is measured by Distinct-1/2 (Li et al., 2016) and Self-BLEU (Zhu et al., 2018). We further average the min-max normalized Distinct-1, Distinct-2, and $1 - \text{Self-BLEU}$ within each temperature as an aggregated diversity score.

Figure 4 shows a clear accuracy–diversity trade-off. More exploratory methods, such as Top-p,

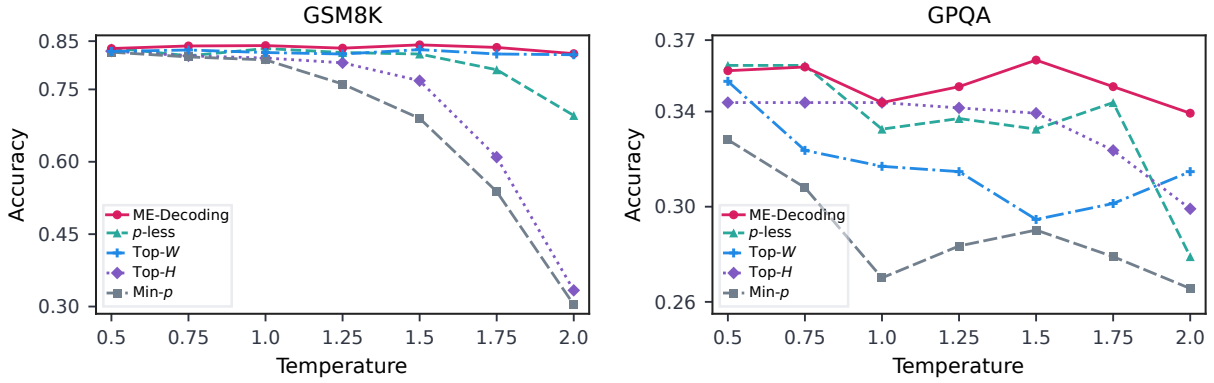

Figure 2: Accuracy vs. temperature curves of different decoding methods on GSM8K and GPQA.

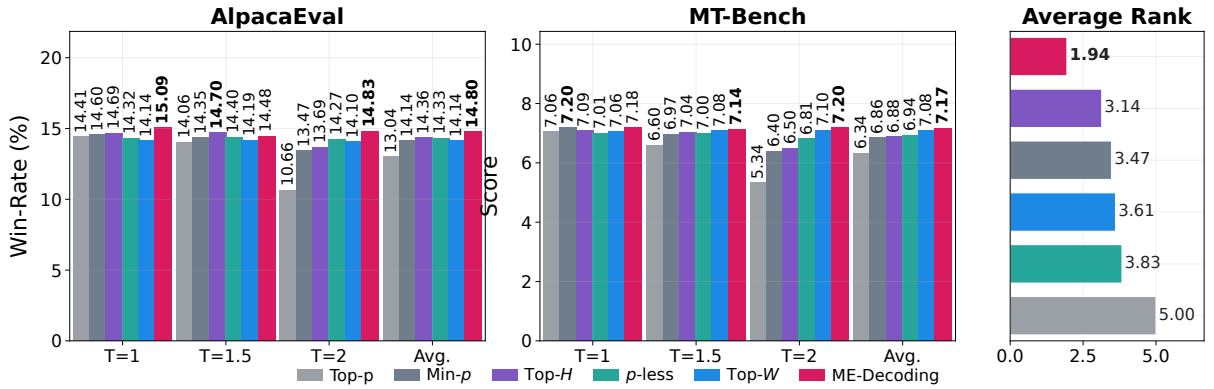

Figure 3: Open-ended generation performance and overall ranking comparison. The **left** and **middle** panels report AlpacaEval win rate and MT-Bench score across different temperatures, averaged over three models and aggregated over 3 runs. The **right** panel summarizes the average rank of each decoding method across all models, temperatures, and benchmarks, with bold numbers indicating the best overall rank.

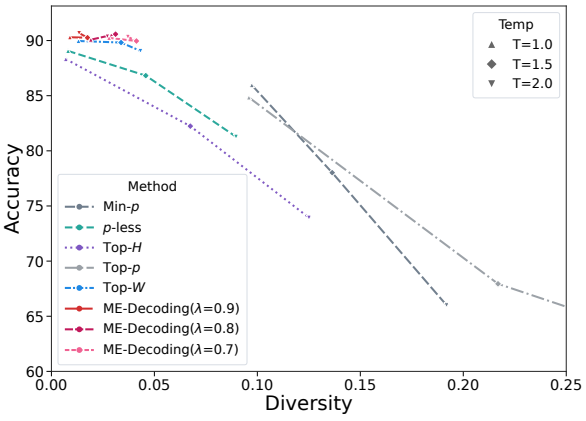

Figure 4: Accuracy–diversity trade-off on GSM8K using Qwen3-4B. Each method generates 10 responses for 500 sampled questions under each temperature.

achieve higher diversity, especially at high temperatures, but suffer from notable accuracy degradation. In contrast, ME-Decoding consistently achieves the highest accuracy and maintains stronger reasoning performance at comparable diversity levels. This suggests that the proposed Mahalanobis-Ensemble objective improves token selection reliability by constructing a compact and informative sampling support, rather than simply increasing output randomness. Detailed results are provided in Table 15.

4.4 Support-Level Redundancy Analysis

Output diversity and selected-support redundancy describe different stages of generation. We therefore measure support redundancy at each decoding step by averaging the cosine similarity over all unordered pairs of selected, L2-normalized token embeddings. Table 5 reports the core results on Qwen2.5-1.5B at $T = 1.0$ using three generation seeds.

On GSM8K, ME-Decoding achieves the highest accuracy and the lowest support cosine while retaining a support size and probability mass comparable to the baselines. On GPQA, it attains the highest accuracy and the smallest average support, although its cosine is not the lowest. This pattern is consistent with the design of MES, which balances token

Setting	Statistic	Top-p	Min-p	p-less	Top-H	Top-W	**ME-Decoding**
Large	Mean s/token	0.01407	0.00196	0.00314	0.01493	0.13337	0.01799
	Standard Deviation	0.00006	0.00002	0.00000	0.00009	0.00098	0.00560
Medium	Mean s/token	0.00335	0.00054	0.00080	0.00330	0.00487	0.00179
	Standard Deviation	0.00003	0.00001	0.00001	0.00005	0.00003	0.00010

Table 4: Average CPU sampling overhead per token on synthetic logits. **Large**: vocabulary size 128K, embedding dimension 1024, top-$N = 2048$. **Medium**: vocabulary size 32K, embedding dimension 256, top-$N = 512$.

Dataset	Method	Acc. (%)	Avg. $\lvert S\rvert$	Mass	Cos.
GSM8K	**ME-Decoding**	$\mathbf{63.20 \pm 0.57}$	1.076	0.846	**0.151**
	Top-W	61.64 ± 1.19	1.093	0.841	0.208
	p-less	61.33 ± 0.23	1.164	0.842	0.179
	Top-H	61.16 ± 0.42	2.240	0.847	0.241
GPQA	**ME-Decoding**	$\mathbf{32.66 \pm 1.05}$	**1.134**	0.759	0.243
	Top-W	27.69 ± 0.91	1.213	0.692	0.185
	p-less	28.21 ± 1.05	1.265	0.770	0.192
	Top-H	27.52 ± 0.66	2.877	0.778	**0.184**

Table 5: Core support-level diagnostics on Qwen2.5-1.5B at $T = 1.0$. Accuracy is reported as mean $\pm$ sample standard deviation over three seeds. Lower cosine indicates less redundant selected token embeddings.

confidence and relational redundancy rather than minimizing cosine alone. Appendix B.2 reports entropy, MES, and approximately size-, entropy-, and cosine-matched probability-only controls.

5 Conclusion

In this paper, we introduce ME-Decoding, a decoding framework that reformulates LLM token selection as ensemble pruning. ME-Decoding maximizes the *Mahalanobis-Ensemble Score* (MES) to select a compact token subset, combining token probabilities with an embedding-based similarity matrix to discount semantic redundancy. We further develop an efficient greedy algorithm with theoretical guarantees, including greedy-trajectory unimodality and an approximation bound. Experiments on reasoning and open-ended generation tasks show that ME-Decoding improves over representative probability-based and geometry-aware baselines with low inference overhead.

These results demonstrate the potential of ensemble pruning as a principled perspective for constructing compact, diverse, and reliable token candidate sets. Future work includes developing simpler and more effective objectives under this perspective, and extending ME-Decoding to broader scenarios such as long-form generation, multi-sample reasoning, and verification-augmented decoding.

Limitations

Although ME-Decoding introduces an ensemble-pruning perspective for LLM decoding and achieves strong performance on several tasks, it still has several limitations. First, its performance depends on the construction of the similarity matrix, which controls the trade-off between token confidence and redundancy. While we provide a practical geometry-based kernel, more effective and adaptive similarity designs may further improve performance. The current kernel uses static token embeddings as a soft redundancy signal and may not capture all context-dependent semantic distinctions. Contextualized or hidden-state-conditioned kernels are therefore a useful direction for future work. Second, although ME-Decoding consistently achieves higher accuracy than competing methods under comparable diversity levels, its output diversity is not always the highest among all decoding strategies. This suggests that the current kernel construction may still be conservative in promoting diverse generations. How to further improve generation diversity while preserving the accuracy gains of ME-Decoding remains an important direction for future work. Third, determining the optimal subset size remains challenging, as in existing truncation-based decoding methods. The proposed MES criterion provides a principled selection rule, but it may not be optimal for all token distributions. Exploring adaptive subset-size rules and alternative objectives is another promising direction.

Ethics Statement

This work proposes ME-Decoding, a geometry-aware decoding framework that reformulates candidate token selection from the perspective of ensemble pruning. By selecting compact yet informative token sets, ME-Decoding aims to improve generation quality and reasoning performance without modifying model parameters or requiring addi-

tional training. A potential positive impact is that such plug-and-play inference-time methods may improve the usability of existing language models with limited computational overhead, thereby reducing deployment costs and the need for repeated model retraining.

At the same time, improved decoding and reasoning performance may also amplify downstream risks associated with generative systems. These risks include misinformation generation, hallucinated but plausible outputs, biased or harmful content, and the automation of low-quality or malicious text at scale. Since ME-Decoding operates at inference time and can be applied to a wide range of pretrained language models, its broader impact depends strongly on the deployment context, access control, and downstream use cases.

We encourage practitioners to use ME-Decoding together with established responsible deployment practices, including safety evaluation, bias and toxicity assessment, hallucination analysis, usage policies, and rate limits in open-ended settings. For high-stakes applications such as medical, legal, financial, or educational decision-making, model outputs should be carefully verified by qualified human experts. In addition, when ME-Decoding is applied to models trained on copyrighted, private, or sensitive data, developers should follow appropriate data governance, documentation, and licensing practices.

All datasets, models, and evaluation tools used in this work are publicly available. We follow their respective licenses and terms of use, and use them solely for research evaluation purposes. Our use of these artifacts is consistent with their intended use as benchmarks, pretrained models, and evaluation tools for research. We do not redistribute or repurpose any artifact beyond the scope allowed by its original access conditions, and we cite the original creators of all benchmarks, models, and baseline methods used in our experiments.

We do not collect any new user data or personally identifying information. The datasets used in our experiments are publicly available benchmarks released for research evaluation. We use them only under their standard evaluation settings and do not attempt to identify individuals or recover private information from any dataset.

References

Abouelenin et al. 2025. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-LoRAs. *arXiv preprint arXiv:2503.01743*.

Achiam et al. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

Erfan Baghaei Potraghloo, Seyedarmin Azizi, Souvik Kundu, and Massoud Pedram. 2026. Top-H decoding: Adapting the creativity and coherence with bounded entropy in text generation. *Advances in Neural Information Processing Systems*, 38:28482–28513.

Leo Breiman. 2001. Random forests. *Machine Learning*, 45(1):5–32.

Brown et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Chowdhery et al. 2023. PaLM: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Cobbe et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Abhimanyu Das and David Kempe. 2018. Approximate submodularity and its applications: Subset selection, sparse approximation and dictionary selection. *Journal of Machine Learning Research*, 19(3):1–34.

Arash Gholami Davoodi, Navid Rezazadeh, Seyed Pouyan Mousavi Davoudi, and Pouya Pezeshkpour. 2026. Geometry-aware decoding with Wasserstein-regularized truncation and mass penalties for large language models. *In International Conference on Machine Learning*.

DeepSeek-AI. 2026. DeepSeek-V4: Towards highly efficient million-token context intelligence. https://huggingface.co/deepseek-ai/DeepSeek-V4-Pro. Technical report.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898.

Gene H Golub and Charles F Van Loan. 2013. *Matrix Computations*. JHU Press.

John Hewitt, Christopher D Manning, and Percy Liang. 2022. Truncation sampling as language model desmoothing. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3414–3427.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. *In International Conference on Learning Representations*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Xinlai Kang, Dunyao Xue, Zhengbo Wang, Chengshuo Du, Xinghao Chen, Hang Zhou, Hanting Chen, and Cheng Meng. 2026. Breaking the echo chamber: A dynamic ensemble pruning perspective on MoE. In *Proceedings of the 43rd International Conference on Machine Learning*.

Anders Krogh and Jesper Vedelsby. 1994. Neural network ensembles, cross validation, and active learning. *Advances in neural information processing systems*, 7:231–238.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. 2018. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems*, volume 31, pages 7167–7177.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and William B Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 110–119.

Nan Li, Yang Yu, and Zhi-Hua Zhou. 2012. Diversity regularized ensemble pruning. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 330–345. Springer.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Alpacaeval: An automatic evaluator of instruction-following models. `https://github.com/tatsu-lab/alpaca_eval`.

Minh Nguyen, Andrew Baker, Clement Neo, Allen Roush, Andreas Kirsch, and Ravid Shwartz-Ziv. 2025. Turning up the heat: Min-p sampling for creative and coherent LLM outputs. In *International Conference on Learning Representations*, volume 2025, pages 70333–70366.

David Opitz and Richard Maclin. 1999. Popular ensemble methods: An empirical study. *Journal of artificial intelligence research*, 11:169–198.

Yagyensh Chandra Pati, Ramin Rezaiifar, and Perinkulam Sambamurthy Krishnaprasad. 1993. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Proceedings of 27th Asilomar conference on signals, systems and computers*, pages 40–44. IEEE.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. 2024. GPQA: A graduate-level google-proof q&a benchmark. *In Proceedings of the First Conference on Language Modeling*.

Runyan Tan, Shuang Wu, and Phillip Howard. 2025. p-less sampling: A robust hyperparameter-free approach for LLM decoding. *In International Conference on Learning Representations*.

Touvron et al. 2023. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Weitao Wan, Yuanyi Zhong, Tianpeng Li, and Jiansheng Chen. 2018. Rethinking feature distribution for loss functions in image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9117–9126.

Kilian Q Weinberger and Lawrence K Saul. 2009. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10(9):207–244.

Yixin Yang, Qingxiu Dong, and Zhifang Sui. 2026. Decoding in geometry: Alleviating embedding-space crowding for complex reasoning. *arXiv preprint arXiv:2601.22536*.

Yang et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Yi Zhang, Samuel Burer, and W. Nick Street. 2006. Ensemble pruning via semi-definite programming. *Journal of Machine Learning Research*, 7(48):1315–1338.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36, pages 46595–46623. Curran Associates, Inc.

Zhi-Hua Zhou, Jianxin Wu, and Wei Tang. 2002. Ensembling neural networks: many could be better than all. *Artificial intelligence*, 137(1-2):239–263.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 1097–1100.

A Proofs of Theoretical Results

Proof of Theorem 3.1. Let $\mathrm{MEE}(S) = \boldsymbol{p}_S^\top \boldsymbol{K}_S^{-1} \boldsymbol{p}_S$ and $c_\lambda = 1 + \lambda H_2^T(p)$. Write $F_t = \mathrm{MEE}(S_t)$ and $\Delta_t = \mathrm{MEE}(S_{t+1}) - \mathrm{MEE}(S_t)$. Then the greedy objective is defined as $\mathrm{MES}_t^g = \frac{F_t}{c_\lambda^t}$.

We first show that the restricted condition number controls the conditional residual correlations. Fix any subset $S \subseteq \mathcal{V}$ and two indices $i, k \notin S$. Define the conditional residual covariance matrix of (i, k) given S as the Schur complement:

$$\boldsymbol{C}_{\{i,k\}|S} = K_{\{i,k\}} - \boldsymbol{K}_{\{i,k\},S} \boldsymbol{K}_S^{-1} \boldsymbol{K}_{S,\{i,k\}}.$$

Write

$$\boldsymbol{C}_{\{i,k\}|S} = \begin{pmatrix} v_i & c_{ki|S} \\ c_{ki|S} & v_k \end{pmatrix},$$

where $v_i = K_{ii} - \boldsymbol{K}_{iS} \boldsymbol{K}_S^{-1} \boldsymbol{K}_{Si}$, $v_k = K_{kk} - \boldsymbol{K}_{kS} \boldsymbol{K}_S^{-1} \boldsymbol{K}_{Sk}$, and $c_{ki|S} = K_{ki} - \boldsymbol{K}_{kS} \boldsymbol{K}_S^{-1} \boldsymbol{K}_{Si}$. The corresponding conditional residual correlation is

$$\rho_{ki|S} = \frac{c_{ki|S}}{\sqrt{v_i v_k}}.$$

Since $\boldsymbol{C}_{\{i,k\}|S}$ is the Schur complement of $\boldsymbol{K}_S$ in $\boldsymbol{K}_{S\cup\{i,k\}}$, its eigenvalues are bounded by those of $\boldsymbol{K}_{S\cup\{i,k\}}$. More precisely,

$$\lambda_{\min}(\boldsymbol{K}_{S\cup\{i,k\}}) \le \lambda_{\max}(\boldsymbol{C}_{\{i,k\}|S}) \le \lambda_{\max}(\boldsymbol{K}_{S\cup\{i,k\}}).$$

These inequalities follow from the variational characterization of eigenvalues and the Schur-complement residualization argument. Therefore,

$$\kappa(\boldsymbol{C}_{\{i,k\}|S}) \le \kappa(\boldsymbol{K}_{S\cup\{i,k\}}) \le \kappa_N(\boldsymbol{K}).$$

On the other hand, for any 2×2 positive definite matrix

$$\boldsymbol{C} = \begin{pmatrix} v_i & c \\ c & v_k \end{pmatrix},$$

with correlation $\rho = \frac{c}{\sqrt{v_i v_k}}$, we have:

$$|\rho| \le \frac{\kappa(\boldsymbol{C}) - 1}{\kappa(\boldsymbol{C}) + 1},$$

which gives

$$\kappa(\boldsymbol{C}) \ge \frac{1 + |\rho|}{1 - |\rho|}.$$

Applying this to $\boldsymbol{C}_{\{i,k\}|S}$ gives

$$\frac{1 + |\rho_{ki|S}|}{1 - |\rho_{ki|S}|} \le \kappa(\boldsymbol{C}_{\{i,k\}|S}) \le \kappa_N(\boldsymbol{K}).$$

By the assumption $\kappa_N(\boldsymbol{K}) \le c_\lambda$, we obtain

$$\frac{1 + |\rho_{ki|S}|}{1 - |\rho_{ki|S}|} \le c_\lambda. \tag{4}$$

Next, we prove the one-step growth control of greedy marginal gains. Fix a greedy step t, let $S = S_t$, and let i be the element added at step t, i.e., $S_{t+1} = S_t \cup \{i\}$. For any remaining token $k \notin S \cup \{i\}$, define the standardized residual scores

$$z_i = \frac{p_i - \boldsymbol{K}_{iS} \boldsymbol{K}_S^{-1} \boldsymbol{p}_S}{\sqrt{K_{ii} - \boldsymbol{K}_{iS} \boldsymbol{K}_S^{-1} \boldsymbol{K}_{Si}}},$$

and define z_k analogously. By the Schur-complement formula, $\Delta\text{MEE}(i \mid S) = z_i^2$ and $\Delta\text{MEE}(k \mid S) = z_k^2$. Since i is selected greedily, $z_k^2 \le z_i^2$. If $z_i = 0$, then all remaining marginal gains are zero and the conclusion is immediate. Otherwise, define $r_k = \frac{z_k}{z_i}$, where $|r_k| \le 1$. After adding i, the marginal gain of k is

$$\Delta\text{MEE}(k \mid S \cup \{i\}) = \frac{(z_k - \rho_{ki|S} z_i)^2}{1 - \rho_{ki|S}^2}.$$

Hence

$$\frac{\Delta\text{MEE}(k \mid S \cup \{i\})}{\Delta\text{MEE}(i \mid S)} = \frac{(r_k - \rho_{ki|S})^2}{1 - \rho_{ki|S}^2} \le \frac{(1 + |\rho_{ki|S}|)^2}{1 - \rho_{ki|S}^2} = \frac{1 + |\rho_{ki|S}|}{1 - |\rho_{ki|S}|} \le c_\lambda, \tag{5}$$

where the last inequality follows from (4). Taking the maximum over all remaining k gives

$$\Delta_{t+1} \le c_\lambda \Delta_t. \tag{6}$$

Now suppose $\text{MES}_{t+1}^g \le \text{MES}_t^g$. Since $\text{MES}_t^g = \frac{F_t}{c_\lambda^t}$, this is equivalent to

$$\frac{F_{t+1}}{c_\lambda^{t+1}} \le \frac{F_t}{c_\lambda^t}.$$

Using $F_{t+1} = F_t + \Delta_t$, we get $F_t + \Delta_t \le c_\lambda F_t$. Define $R_t = \frac{\Delta_t}{F_t}$. Then

$$\text{MES}_{t+1}^g \le \text{MES}_t^g \quad \Longleftrightarrow \quad R_t \le c_\lambda - 1. \tag{7}$$

Using (6), we have

$$\begin{aligned} R_{t+1} &= \frac{\Delta_{t+1}}{F_{t+1}} = \frac{\Delta_{t+1}}{F_t + \Delta_t} \\ &\le \frac{c_\lambda \Delta_t}{F_t + \Delta_t} = \frac{c_\lambda R_t}{1 + R_t}. \end{aligned}$$

The function $h(R) = \frac{c_\lambda R}{1+R}$ is increasing for $R \ge 0$. Therefore, if $R_t \le c_\lambda - 1$, then

$$R_{t+1} \le h(R_t) \le h(c_\lambda - 1) = c_\lambda - 1.$$

By induction, $R_s \le c_\lambda - 1$ for all $s \ge t$. Using (7), we obtain

$$\text{MES}_{s+1}^g \le \text{MES}_s^g, \qquad \forall s \ge t.$$

Finally, define

$$\tau = \min\{t \in \{0, \dots, N-1\} : \text{MES}_{t+1}^g \le \text{MES}_t^g\},$$

the sequence increases strictly before τ and is nonincreasing after τ. Hence

$$\text{MES}_\tau^g = \max_{0 \le r \le N} \text{MES}_r^g.$$

This completes the proof. □

Lemma A.1 (Approximation Guarantee for Greedy Selection)**.** *Let $f : 2^U \to \mathbb{R}_{\ge 0}$ be a normalized ($f(\emptyset) = 0$), nonnegative, monotone set function, and let $\text{OPT} = \max_{|S| \le k} f(S)$ denote the maximum value obtained by any set of size at most k. Let S be the set selected by the Greedy algorithm. Then, the solution satisfies the following approximation guarantee:*

$$f(S) \ge \left(1 - e^{-\gamma_{U,k}}\right) \cdot \text{OPT}, \tag{8}$$

where $\gamma_{U,k}$ is the submodularity ratio of f. Formally, $\gamma_{U,k}$ is defined as the minimum ratio of the marginal gain of a set to the marginal gain of its individual elements:

$$\gamma_{U,k} = \min_{L \subseteq U,\, A : |A| \le k,\, L \cap A = \emptyset} \frac{\sum_{x \in A} \left(f(L \cup \{x\}) - f(L)\right)}{f(L \cup A) - f(L)}. \tag{9}$$

The ratio is taken over pairs (L, A) such that $f(L \cup A) > f(L)$; if the denominator is zero, the ratio is defined as 1.

Proof. This is the standard approximation guarantee for greedy maximization of a nonnegative monotone set function with submodularity ratio $\gamma_{U,k}$. Proof can be found in Das and Kempe (2018). □

While Lemma A.1 offers a general guarantee, the ratio $\gamma_{\mathcal{V},k}$ is typically intractable. We further provide a concrete bound for our Mahalanobis-Ensemble Energy (MEE) by considering the specific objective

$$f(S) = \mathrm{MEE}(S) = \boldsymbol{p}_S^\top \boldsymbol{K}_S^{-1} \boldsymbol{p}_S.$$

The following lemma bounds $\gamma_{\mathcal{V},k}$ via the restricted minimum eigenvalue of the kernel matrix $\boldsymbol{K}$.

Lemma A.2 (Schur Complement Preserves the Minimum Eigenvalue)**.** *Let $\boldsymbol{K} \in \mathbb{R}^{n\times n}$ be a symmetric positive definite matrix. Write*

$$\boldsymbol{K} = \begin{pmatrix} \boldsymbol{K}_{11} & \boldsymbol{s} \\ \boldsymbol{s}^\top & K_{nn} \end{pmatrix}, \quad \boldsymbol{K}_{11} \in \mathbb{R}^{(n-1)\times(n-1)},\ \boldsymbol{s} \in \mathbb{R}^{(n-1)\times 1},\ K_{nn} > 0.$$

Define the Schur complement

$$\boldsymbol{K}' = \boldsymbol{K}_{11} - \frac{1}{K_{nn}} \boldsymbol{s}\boldsymbol{s}^\top.$$

Then

$$\lambda_{\min}(\boldsymbol{K}) \le \lambda_{\min}(\boldsymbol{K}').$$

Proof. Let $\lambda_1' = \lambda_{\min}(\boldsymbol{K}')$, and choose a nonzero eigenvector $\boldsymbol{e}' \in \mathbb{R}^{n-1}$ such that $\boldsymbol{K}'\boldsymbol{e}' = \lambda_1'\boldsymbol{e}'$. We construct $\boldsymbol{e} = \begin{pmatrix} \boldsymbol{e}' \\ e_n \end{pmatrix}$, where $e_n = -\frac{1}{K_{nn}} \boldsymbol{s}^\top \boldsymbol{e}'$. Using the block form of $\boldsymbol{K}$, we have:

$$\boldsymbol{K}\boldsymbol{e} = \begin{pmatrix} \boldsymbol{K}_{11}\boldsymbol{e}' + \boldsymbol{s}e_n \\ \boldsymbol{s}^\top \boldsymbol{e}' + K_{nn} e_n \end{pmatrix}.$$

By the definition of e_n, the lower block becomes $\boldsymbol{s}^\top \boldsymbol{e}' + K_{nn} e_n = 0$. For the upper block, we have $\boldsymbol{K}_{11}\boldsymbol{e}' + \boldsymbol{s}e_n = \boldsymbol{K}_{11}\boldsymbol{e}' - \frac{1}{K_{nn}} \boldsymbol{s}\boldsymbol{s}^\top \boldsymbol{e}' = \left(\boldsymbol{K}_{11} - \frac{1}{K_{nn}} \boldsymbol{s}\boldsymbol{s}^\top\right) \boldsymbol{e}' = \boldsymbol{K}'\boldsymbol{e}' = \lambda_1'\boldsymbol{e}'$. Therefore, $\boldsymbol{K}\boldsymbol{e} = \begin{pmatrix} \lambda_1'\boldsymbol{e}' \\ 0 \end{pmatrix}$.

By the Rayleigh quotient, we know $\lambda_{\min}(\boldsymbol{K}) = \min_{\boldsymbol{x}\neq 0} \frac{\boldsymbol{x}^\top \boldsymbol{K}\boldsymbol{x}}{\boldsymbol{x}^\top \boldsymbol{x}} \le \frac{\boldsymbol{e}^\top \boldsymbol{K}\boldsymbol{e}}{\boldsymbol{e}^\top \boldsymbol{e}}$. Since $\boldsymbol{e}^\top \boldsymbol{K}\boldsymbol{e} = \lambda_1' \|\boldsymbol{e}'\|_2^2$ and $\boldsymbol{e}^\top \boldsymbol{e} = \|\boldsymbol{e}'\|_2^2 + e_n^2 \ge \|\boldsymbol{e}'\|_2^2$, we can bound the quotient as follows:

$$\lambda_{\min}(\boldsymbol{K}) \le \frac{\boldsymbol{e}^\top \boldsymbol{K}\boldsymbol{e}}{\boldsymbol{e}^\top \boldsymbol{e}} = \frac{\lambda_1' \|\boldsymbol{e}'\|_2^2}{\|\boldsymbol{e}'\|_2^2 + e_n^2} \le \lambda_1'.$$

Since $\boldsymbol{K}' \succ 0$, we have $\lambda_1' > 0$, which implies $\lambda_1' = \lambda_{\min}(\boldsymbol{K}')$. This proves the claim. □

Lemma A.3 (Spectral Bound on Submodularity Ratio)**.** *Assume that $\boldsymbol{K} \succ 0$ is a correlation matrix. For $f(S) = \boldsymbol{p}_S^\top \boldsymbol{K}_S^{-1} \boldsymbol{p}_S$ and disjoint sets L, A, let $\boldsymbol{K}_{L\cup A}$ be partitioned naturally into blocks $\boldsymbol{K}_L, \boldsymbol{K}_A, \boldsymbol{K}_{LA}, \boldsymbol{K}_{AL}$. We define the residual statistics $\tilde{\boldsymbol{p}}$ and $\tilde{\boldsymbol{K}}$ (Schur complement) as:*

$$\tilde{\boldsymbol{p}} = \boldsymbol{p}_A - \boldsymbol{K}_{AL}\boldsymbol{K}_L^{-1}\boldsymbol{p}_L, \quad \tilde{\boldsymbol{K}} = \boldsymbol{K}_A - \boldsymbol{K}_{AL}\boldsymbol{K}_L^{-1}\boldsymbol{K}_{LA}.$$

The submodularity ratio $\gamma_{\mathcal{V},k}$, which involves minimizing the term $\frac{\tilde{\boldsymbol{p}}^\top [\mathrm{diag}(\tilde{\boldsymbol{K}})]^{-1} \tilde{\boldsymbol{p}}}{\tilde{\boldsymbol{p}}^\top \tilde{\boldsymbol{K}}^{-1} \tilde{\boldsymbol{p}}}$, is bounded from below by the eigenvalues of $\boldsymbol{K}$:

$$\gamma_{\mathcal{V},k} \ge \min_{\substack{L\subseteq\mathcal{V},\ A\subseteq\mathcal{V}\setminus L \\ |A|\le k}} \lambda_{\min}(\boldsymbol{K}, |L\cup A|) \ge \lambda_{\min}(\boldsymbol{K}, N),$$

where $\lambda_{\min}(\boldsymbol{K}, k) \triangleq \min_{S:|S|=k} \lambda_{\min}(\boldsymbol{K}_S)$ denotes the smallest eigenvalue among all $k\times k$ principal submatrices of $\boldsymbol{K}$, and $N = |\mathcal{V}|$.

Proof. Consider disjoint L and A. Using the block inverse formula for $\boldsymbol{K}_{L\cup A}^{-1}$, one obtains the decomposition

$$f(L\cup A) = \boldsymbol{p}_L^\top \boldsymbol{K}_L^{-1}\boldsymbol{p}_L \;+\; \tilde{\boldsymbol{p}}^\top \tilde{\boldsymbol{K}}^{-1}\tilde{\boldsymbol{p}},$$

hence

$$f(L\cup A) - f(L) = \tilde{\boldsymbol{p}}^\top \tilde{\boldsymbol{K}}^{-1}\tilde{\boldsymbol{p}}.$$

For a singleton $a \in A$, the same calculation with $|A| = 1$ gives

$$f(L\cup\{a\}) - f(L) = \frac{\tilde{p}_a^2}{\tilde{K}_{aa}}.$$

Therefore, for any $L \subseteq \mathcal{V}$ and A with $|A| \le k$ and $A \cap L = \emptyset$, the ratio appearing in the definition of the submodularity ratio becomes

$$\frac{\sum_{a\in A}\big(f(L\cup\{a\}) - f(L)\big)}{f(L\cup A) - f(L)} = \frac{\tilde{\boldsymbol{p}}^\top\big[\mathrm{diag}(\tilde{\boldsymbol{K}})\big]^{-1}\tilde{\boldsymbol{p}}}{\tilde{\boldsymbol{p}}^\top \tilde{\boldsymbol{K}}^{-1}\tilde{\boldsymbol{p}}}.$$

Then, let $\boldsymbol{D} = \mathrm{diag}(\tilde{\boldsymbol{K}})$ and define the correlation matrix $\tilde{\boldsymbol{K}}_\rho = \boldsymbol{D}^{-1/2}\tilde{\boldsymbol{K}}\boldsymbol{D}^{-1/2}$. Let $\boldsymbol{v} = \boldsymbol{D}^{-1/2}\tilde{\boldsymbol{p}}$. Then

$$\tilde{\boldsymbol{p}}^\top \boldsymbol{D}^{-1}\tilde{\boldsymbol{p}} = \|\boldsymbol{v}\|_2^2, \qquad \tilde{\boldsymbol{p}}^\top \tilde{\boldsymbol{K}}^{-1}\tilde{\boldsymbol{p}} = \boldsymbol{v}^\top \tilde{\boldsymbol{K}}_\rho^{-1}\boldsymbol{v}.$$

Hence the ratio equals $\frac{\boldsymbol{v}^\top\boldsymbol{v}}{\boldsymbol{v}^\top\tilde{\boldsymbol{K}}_\rho^{-1}\boldsymbol{v}}$. By Rayleigh–Ritz, $\boldsymbol{v}^\top\tilde{\boldsymbol{K}}_\rho^{-1}\boldsymbol{v} \le \lambda_{\max}(\tilde{\boldsymbol{K}}_\rho^{-1})\,\boldsymbol{v}^\top\boldsymbol{v} = \frac{1}{\lambda_{\min}(\tilde{\boldsymbol{K}}_\rho)}\boldsymbol{v}^\top\boldsymbol{v}$, so

$$\frac{\tilde{\boldsymbol{p}}^\top \boldsymbol{D}^{-1}\tilde{\boldsymbol{p}}}{\tilde{\boldsymbol{p}}^\top \tilde{\boldsymbol{K}}^{-1}\tilde{\boldsymbol{p}}} \ge \lambda_{\min}(\tilde{\boldsymbol{K}}_\rho).$$

We can eliminate the elements of L one by one via residualization; each elimination replaces the current covariance by the covariance of residuals. By Lemma A.2, each such residualization step cannot decrease the smallest eigenvalue. After eliminating all of L, we obtain the Schur complement $\tilde{\boldsymbol{K}}$ corresponding to conditioning on L, and thus

$$\lambda_{\min}(\boldsymbol{K}_{L\cup A}) \le \lambda_{\min}(\tilde{\boldsymbol{K}}).$$

Finally, normalizing $\tilde{\boldsymbol{K}}$ to unit variance gives $\tilde{\boldsymbol{K}}_\rho$, and $\lambda_{\min}(\tilde{\boldsymbol{K}}) \le \lambda_{\min}(\tilde{\boldsymbol{K}}_\rho)$. Combining the last two displays:

$$\lambda_{\min}(\tilde{\boldsymbol{K}}_\rho) \ge \lambda_{\min}(\boldsymbol{K}_{L\cup A}) \ge \lambda_{\min}(\boldsymbol{K}, |L\cup A|).$$

By definition of $\gamma_{\mathcal{V},k}$, we conclude

$$\gamma_{\mathcal{V},k} \ge \min_{\substack{L\subseteq\mathcal{V},\ A\subseteq\mathcal{V}\setminus L\\ |A|\le k}} \lambda_{\min}(\boldsymbol{K}, |L\cup A|).$$

Since $\boldsymbol{K}_{L\cup A}$ is a principal submatrix of $\boldsymbol{K}$, Cauchy's interlacing theorem gives

$$\lambda_{\min}(\boldsymbol{K}_{L\cup A}) \ge \lambda_{\min}(\boldsymbol{K}).$$

In our notation, $\lambda_{\min}(\boldsymbol{K}) = \lambda_{\min}(\boldsymbol{K}, N)$, which naturally completes the proof. □

Lemma A.4 (Approximation Bound for Greedy MEE)**.** *Assume that $\boldsymbol{K} \succ 0$ is a correlation matrix defined on ground set $\mathcal{V}$ ($|\mathcal{V}| = N$). Let S_k be the greedy set after k steps. Then the approximation bound holds:*

$$f(S_k) \geq (1 - \exp\{-\lambda_{\min}(\boldsymbol{K}, N)\}) \max_{\substack{S \subseteq \mathcal{V} \\ |S| \leq k}} f(S).$$

Proof. Let $\mathrm{OPT} \triangleq \max_{S \subseteq \mathcal{V}, |S| \leq k} \mathrm{MEE}(S)$.

First, since the Mahalanobis-Ensemble Energy objective $\mathrm{MEE}(S) = \boldsymbol{p}_S^\top \boldsymbol{K}_S^{-1} \boldsymbol{p}_S$ is a normalized, nonnegative, and monotone set function, we can apply the general greedy guarantee from Lemma A.1. This yields:

$$\mathrm{MEE}(S_k) \geq \left(1 - e^{-\gamma_{\mathcal{V},k}}\right) \cdot \mathrm{OPT},$$

where $\gamma_{\mathcal{V},k}$ is the submodularity ratio of MEE on the ground set $\mathcal{V}$.

Next, we lower bound the submodularity ratio $\gamma_{\mathcal{V},k}$. By Lemma A.3, the submodularity ratio satisfies:

$$\gamma_{\mathcal{V},k} \geq \min_{\substack{L \subseteq \mathcal{V},\ A \subseteq \mathcal{V} \setminus L \\ |A| \leq k}} \lambda_{\min}(\boldsymbol{K}, |L \cup A|).$$

Since $L \cup A \subseteq \mathcal{V}$, we have $|L \cup A| \leq N$. By the interlacing theorem, every principal submatrix of $\boldsymbol{K}$ has a minimum eigenvalue at least $\lambda_{\min}(\boldsymbol{K}, N)$. Hence:

$$\gamma_{\mathcal{V},k} \geq \lambda_{\min}(\boldsymbol{K}, N).$$

Substituting this spectral lower bound back into the greedy approximation guarantee gives:

$$\mathrm{MEE}(S_k) \geq \left(1 - e^{-\lambda_{\min}(\boldsymbol{K}, N)}\right) \cdot \mathrm{OPT}.$$

This completes the proof. □

Proof of Theorem 3.2. Let $\mathrm{MEE}(S) = \boldsymbol{p}_S^\top \boldsymbol{K}_S^{-1} \boldsymbol{p}_S$ and $c_\lambda = 1 + \lambda H_2^T(\boldsymbol{p})$. The objective can be written as $\mathrm{MES}_\lambda(S) = \frac{\mathrm{MEE}(S)}{c_\lambda^{|S|}}$. Define the greedy approximation factor as $\alpha_N = 1 - \exp\{-\lambda_{\min}(\boldsymbol{K}, N)\}$.

By Lemma A.4, for every $k = 1, \ldots, N$, the greedy set S_k satisfies:

$$\mathrm{MEE}(S_k) \geq \alpha_N \max_{S \subseteq \mathcal{V}\ |S| \leq k} \mathrm{MEE}(S). \tag{10}$$

Fix any $k \in \{1, \ldots, N\}$. Dividing both sides of (10) by c_λ^k gives:

$$\begin{aligned}
\mathrm{MES}_\lambda(S_k) = \frac{\mathrm{MEE}(S_k)}{c_\lambda^k} &\geq \alpha_N \frac{\max_{S \subseteq \mathcal{V}, |S| \leq k} \mathrm{MEE}(S)}{c_\lambda^k} \\
&\geq \alpha_N \frac{\max_{S \subseteq \mathcal{V}, |S| = k} \mathrm{MEE}(S)}{c_\lambda^k} \\
&= \alpha_N \max_{\substack{S \subseteq \mathcal{V} \\ |S| = k}} \frac{\mathrm{MEE}(S)}{c_\lambda^{|S|}} = \alpha_N \max_{\substack{S \subseteq \mathcal{V} \\ |S| = k}} \mathrm{MES}_\lambda(S).
\end{aligned}$$

Taking the maximum over all subset sizes $k = 1, \ldots, N$ on both sides, we obtain the global bound for the greedy trajectory:

$$\max_{1 \leq k \leq N} \mathrm{MES}_\lambda(S_k) \geq \alpha_N \max_{1 \leq k \leq N} \max_{S \subseteq \mathcal{V}\ |S| = k} \mathrm{MES}_\lambda(S). \tag{11}$$

By Theorem 3.1, the greedy saturation stopping point τ achieves the exact maximum of the entire greedy sequence, meaning $\mathrm{MES}_\tau^g = \max_{0 \leq r \leq N} \mathrm{MES}_r^g$. Since $\mathrm{MES}_r^g = \mathrm{MES}_\lambda(S_r)$, we have:

$$\mathrm{MES}_\tau^g \geq \max_{1\leq k\leq N} \mathrm{MES}_\lambda(S_k).$$

Combining this with (11) and substituting the definition of α_N, we arrive at our final conclusion:

$$\mathrm{MES}_\tau^g \geq (1-\exp\{-\lambda_{\min}(\boldsymbol{K},N)\}) \max_{\emptyset\neq S\subseteq\mathcal{V}} \mathrm{MES}_\lambda(S).$$

This completes the proof. □

B Additional Evidence from the Discussion Phase

This section incorporates the additional controls requested during review. Unless stated otherwise, repeated generation results use three shared seeds and report the mean ± sample standard deviation.

B.1 Repeated Reasoning Evaluation and CraEG Comparison

Tables 6 and 7 repeat the reasoning evaluation over three seeds and include our implementation of CraEG combined with p-less. This composition follows CraEG's role as a post-softmax reweighting module rather than a standalone support-selection rule.

B.2 Support Diagnostics and Matched Controls

We measure the retained probability mass, average support size, entropy before and after pruning, and selected-support redundancy. For each decoding step with at least two selected tokens, cosine similarity is averaged over all unordered pairs of L2-normalized token embeddings and then averaged across decoding steps. The matched Min-p controls are calibrated on independent prompts to approximately match ME-Decoding's overall average support size, post-pruning entropy, or pairwise cosine.

ME-Decoding remains more accurate than the approximately matched controls on both datasets. These controls indicate that its gains are not explained solely by selecting fewer tokens or by reducing post-pruning entropy.

B.3 Probability and Kernel Controls

We use Greedy decoding to test whether repeatedly selecting the locally most probable token is sufficient. At $T = 1.0$, ME-Decoding improves GSM8K accuracy from 61.41% to $63.20 \pm 0.57\%$ and GPQA accuracy from 32.37% to $32.66 \pm 1.05\%$ on Qwen2.5-1.5B. Greedy is deterministic, whereas ME-Decoding results average three generation seeds.

We then compare the semantic kernel with two controlled alternatives. The identity kernel $K = I$ removes pairwise geometry, reducing the MEE numerator to $\sum_{i \in S} p_i^2$. The permuted kernel $K_{\text{perm}} = PK_{\text{true}}P^\top$ preserves the eigenvalues, condition number, entry multiset, and numerical scale of the semantic kernel while disrupting its alignment with token identities. All variants use the same prompts, probabilities, generation seeds, and pruning hyperparameters.

Dataset	True K	Permuted K	$K = I$
GSM8K	**62.32**	61.15	60.85
GPQA	**32.64**	32.09	29.81

Table 9: Accuracy (%) averaged over $T \in \{1.0, 1.5, 2.0\}$ in the Qwen2.5-1.5B kernel-control experiment. Both controls reduce accuracy relative to the correctly aligned semantic kernel.

B.4 Empirical Greedy-Trajectory Behavior

The condition in Theorem 3.1 is sufficient rather than necessary for trajectory unimodality. To examine whether the predicted behavior occurs in practice, we disable early stopping and record the complete greedy MES trajectory for a 512-trajectory probe from each benchmark using Qwen2.5-1.5B at $T = 1.0$.

Dataset	Unimodal Trajectories	Rate
GSM8K	512/512	**100%**
GPQA	512/512	**100%**

Table 10: Empirical unimodality of complete greedy MES trajectories.

The observation supports the practical relevance of the stopping behavior without replacing the theorem's sufficient-condition analysis. The approximation result has a separate scope and depends on restricted minimum eigenvalues of selected kernel submatrices.

B.5 Open-Ended Evaluation Robustness

AlpacaEval-style evaluation compares each generated response with a fixed reference for the same prompt, while MT-Bench-style evaluation independently assigns a score from 1 to 10. All methods use matched samples at $T \in \{1.0, 1.5, 2.0\}$ over three generation runs. We evaluate the same outputs with DeepSeek-V4-Pro and GLM5-2.

All intervals are positive for both judges and benchmarks. Additional length, style, and refusal diagnostics indicate that these gains are not explained by verbosity or conservative refusal behavior.

B.6 End-to-End GPU Efficiency

We benchmark 200 GSM8K problems on one NVIDIA GeForce RTX 4090, generating 64 tokens per problem with $T = 1.5$, batch size 1, and $N = 512$. Each method therefore generates 12,800

Method	Qwen3-4B-Instruct			Phi-4-mini-Instruct			Mistral-7B-Instruct			Avg.
	$T = 1.0$	$T = 1.5$	$T = 2.0$	$T = 1.0$	$T = 1.5$	$T = 2.0$	$T = 1.0$	$T = 1.5$	$T = 2.0$	
p-less	78.82 ± 0.24	72.53 ± 0.50	64.47 ± 1.00	82.99 ± 0.76	82.99 ± 0.39	71.14 ± 0.99	53.53 ± 0.73	51.96 ± 0.74	46.07 ± 1.30	67.17
Top-W	77.63 ± 0.35	76.65 ± 0.23	75.54 ± 0.72	83.24 ± 0.13	83.52 ± 0.23	82.99 ± 0.18	53.07 ± 0.40	53.50 ± 1.05	52.44 ± 1.26	70.95
CraEG+p-less	72.23 ± 1.71	65.07 ± 1.00	58.12 ± 1.32	83.52 ± 0.55	81.91 ± 1.07	65.38 ± 1.26	52.72 ± 0.12	50.82 ± 0.57	40.56 ± 1.64	63.37
ME-Decoding	$\mathbf{80.04 \pm 0.69}$	$\mathbf{79.56 \pm 0.35}$	$\mathbf{80.04 \pm 0.80}$	$\mathbf{83.90 \pm 0.24}$	$\mathbf{83.95 \pm 0.31}$	82.44 ± 0.12	$\mathbf{54.18 \pm 1.03}$	$\mathbf{53.55 \pm 0.12}$	$\mathbf{53.98 \pm 0.57}$	**72.40**

Table 6: GSM8K flexible-extract accuracy (%) over three generation seeds. The Avg. column averages the nine model–temperature means.

Method	Qwen3-4B-Instruct			Phi-4-mini-Instruct			Mistral-7B-Instruct			Avg.
	$T = 1.0$	$T = 1.5$	$T = 2.0$	$T = 1.0$	$T = 1.5$	$T = 2.0$	$T = 1.0$	$T = 1.5$	$T = 2.0$	
p-less	35.57 ± 2.29	35.86 ± 1.57	33.85 ± 1.35	32.81 ± 1.77	32.66 ± 0.78	30.06 ± 2.79	28.42 ± 1.44	28.72 ± 1.23	27.68 ± 1.39	31.74
Top-W	35.49 ± 0.45	34.90 ± 0.78	35.86 ± 0.26	31.62 ± 0.78	30.58 ± 0.77	30.65 ± 1.49	28.79 ± 1.02	29.39 ± 0.34	29.39 ± 1.49	31.85
CraEG+p-less	35.71 ± 1.24	34.08 ± 2.17	33.93 ± 0.97	32.22 ± 1.42	32.14 ± 0.80	29.54 ± 0.68	28.65 ± 0.68	28.72 ± 2.03	28.20 ± 1.23	31.46
ME-Decoding	$\mathbf{36.68 \pm 0.13}$	$\mathbf{36.76 \pm 0.68}$	$\mathbf{36.61 \pm 0.59}$	$\mathbf{34.30 \pm 0.68}$	$\mathbf{34.45 \pm 0.78}$	$\mathbf{34.08 \pm 0.56}$	$\mathbf{29.09 \pm 1.01}$	$\mathbf{29.84 \pm 0.34}$	29.32 ± 0.46	**33.46**

Table 7: GPQA accuracy (%) over three generation seeds. The Avg. column averages the nine model–temperature means.

tokens. ME-Decoding uses a custom Triton kernel, and all methods use identical filtering and sampling infrastructure.

B.7 CPU Logits-Processing Efficiency

The available Top-W implementation is CPU-only, so we retain a separate CPU benchmark that isolates logits processing and sampling. The Medium setting uses a vocabulary of 32K, embedding dimension 256, and $N = 512$; the Large setting uses a vocabulary of 128K, embedding dimension 1024, and $N = 2048$.

Dataset	Method	Acc. (%)	Mass	Cos.	Avg. $\lvert S \rvert$	H_{before}	H_{after}	MES
GSM8K	**ME-Decoding**	$\mathbf{63.20 \pm 0.57}$	0.846	0.151	1.076	0.614	0.052	0.756
	Top-W	61.64 ± 1.19	0.841	0.208	1.093	0.692	0.057	0.688
	p-less	61.33 ± 0.23	0.842	0.179	1.164	0.659	0.075	0.689
	Top-H	61.16 ± 0.42	0.847	0.241	2.240	0.736	0.191	0.660
	Approx. size-matched	61.79 ± 1.26	0.850	0.210	1.067	0.615	0.044	0.707
	Approx. entropy-matched	61.79 ± 0.99	0.851	0.208	1.070	0.614	0.045	0.708
	Approx. cosine-matched	62.70 ± 0.35	0.840	0.198	1.014	0.598	0.010	0.715
GPQA	**ME-Decoding**	$\mathbf{32.66 \pm 1.05}$	0.759	0.243	1.134	0.983	0.091	0.589
	Top-W	27.69 ± 0.91	0.692	0.185	1.213	1.454	0.124	0.498
	p-less	28.21 ± 1.05	0.770	0.192	1.265	1.002	0.144	0.553
	Top-H	27.52 ± 0.66	0.778	0.184	2.877	1.277	0.431	0.475
	Approx. size-matched	28.47 ± 1.17	0.760	0.289	1.106	0.947	0.069	0.558
	Approx. entropy-matched	28.47 ± 1.17	0.760	0.289	1.106	0.947	0.069	0.558
	Approx. cosine-matched	29.25 ± 0.91	0.737	0.259	1.084	1.018	0.055	0.537

Table 8: Support diagnostics and approximately matched probability-only controls on Qwen2.5-1.5B at $T = 1.0$. Accuracy reports the mean $\pm$ sample standard deviation over three seeds.

Judge	Method	AlpacaEval (%)	MT-Bench
DeepSeek-V4-Pro	Top-p	13.04 ± 2.07	6.34 ± 0.89
	Min-p	14.14 ± 0.60	6.86 ± 0.41
	Top-H	14.36 ± 0.58	6.88 ± 0.33
	p-less	14.33 ± 0.07	6.94 ± 0.11
	Top-W	14.14 ± 0.04	7.08 ± 0.02
	ME-Decoding	$\mathbf{14.80 \pm 0.30}$	$\mathbf{7.17 \pm 0.03}$
GLM5-2	Top-H	9.48 ± 0.78	5.42 ± 0.23
	p-less	10.18 ± 0.24	5.54 ± 0.05
	Top-W	10.11 ± 0.20	5.58 ± 0.03
	ME-Decoding	$\mathbf{10.63 \pm 0.16}$	$\mathbf{5.64 \pm 0.01}$

Table 11: Open-ended evaluation under two automatic judges. Results average temperature-level aggregates over three generation runs.

Judge	Baseline	Δ AlpacaEval (pp)	95% CI	Δ MT-Bench	95% CI
DeepSeek-V4-Pro	Top-W	+0.656	$[0.320, 0.980]$	+0.092	$[0.018, 0.169]$
	p-less	+0.469	$[0.115, 0.810]$	+0.234	$[0.158, 0.305]$
	Top-H	+0.440	$[0.076, 0.799]$	+0.297	$[0.221, 0.373]$
GLM5-2	Top-W	+0.529	$[0.239, 0.819]$	+0.069	$[0.018, 0.120]$
	p-less	+0.456	$[0.138, 0.782]$	+0.107	$[0.052, 0.163]$
	Top-H	+1.157	$[0.824, 1.481]$	+0.221	$[0.162, 0.281]$

Table 12: Paired-bootstrap gains of ME-Decoding. Resampling preserves prompt, model, temperature, and generation-run matching.

Model	Method	Model Inference (s)	Decoding and Sampling (s)	Other (s)	Total (s)	ms/token
Qwen2.5-1.5B	Top-p	233.20	4.29	0.99	238.48	18.63
	Min-p	233.19	2.17	0.57	235.93	18.43
	p-less	233.23	2.17	0.63	236.03	18.44
	Top-H	233.56	4.68	1.00	239.25	18.69
	ME-Decoding	242.85	5.64	1.03	249.52	19.49
Qwen3-4B	Top-p	358.20	2.89	0.50	361.58	28.25
	Min-p	356.42	1.97	0.55	358.94	28.04
	p-less	356.63	1.97	0.59	359.18	28.06
	Top-H	359.14	2.91	0.63	362.68	28.33
	ME-Decoding	360.29	3.01	1.15	364.45	28.47

Table 13: End-to-end GPU timing breakdown.

Setting	Statistic	Top-p	Min-p	p-less	Top-H	Top-W	ME-Decoding
Large	Mean s/token	0.01407	0.00196	0.00314	0.01493	0.13337	0.01799
	Sample SD	0.00006	0.00002	0.00000	0.00009	0.00098	0.00560
Medium	Mean s/token	0.00335	0.00054	0.00080	0.00330	0.00487	0.00179
	Sample SD	0.00003	0.00001	0.00001	0.00005	0.00003	0.00010

Table 14: CPU logits-processing and sampling overhead on synthetic inputs.

C Additional Results

All experiments in this work were conducted on a single NVIDIA GeForce RTX 4090 GPU.

C.1 Additional Experiments

Hyperparameter Sensitivity. We study the sensitivity of ME-Decoding to the hyperparameter λ, which controls the compactness of the selected token set. A larger λ imposes a stronger penalty on the subset size and therefore encourages a tighter candidate set, while a smaller λ allows more tokens to be retained. Table 17 reports the accuracy of ME-Decoding under different values of λ across three models, three temperatures, and two reasoning datasets. The results show that ME-Decoding is relatively stable over a range of λ values and consistently achieves strong performance under different settings. Among the tested values, $\lambda = 0.9$ obtains the best average accuracy on both GSM8K and GPQA. Therefore, unless otherwise specified, we use $\lambda = 0.9$ as the default setting in the main experiments.

Detailed Diversity Results. Table 15 reports the detailed numerical results corresponding to the diversity analysis in the main text. For each temperature, we report Acc, Distinct-1, Distinct-2, $1 - \text{Self-BLEU}$, and the aggregated Diversity Avg. score. The diversity score is computed by min-max normalizing the three diversity metrics within each temperature and then taking their average. Although Top-p often obtains the highest diversity score, especially at high temperature, its accuracy drops substantially. In contrast, ME-Decoding maintains the strongest accuracy across temperatures, showing that it provides a better accuracy–diversity trade-off for reasoning-oriented generation.

Ablation Study. We conduct ablation studies to examine the contribution of the adaptive bandwidth ϵ and the kernel matrix K. The bandwidth ϵ controls the scale of the kernel and thus plays an important role in constructing a stable and meaningful token similarity matrix. As shown in Table 16, removing the adaptive bandwidth leads to consistent performance degradation, especially at higher temperatures, indicating that an appropriately scaled kernel is important for robust optimization. We also replace K with the identity matrix, in which case the objective largely degenerates to a probability-driven selection criterion and becomes closer to p-less. This variant also underperforms the full model, showing that the embedding-based similarity structure provides useful geometric information. Overall, the ablation results confirm that both the adaptive kernel construction and the geometry-aware objective are necessary for the effectiveness of ME-Decoding.

Detailed Results for Instruction-Following and Chat. In the main text, we report the averaged instruction-following and chat performance over three instruction-tuned models. Here, we provide the corresponding per-model visualizations and detailed numerical results. Figure 5 shows the performance curves on MT-Bench and AlpacaEval across temperatures for each model. Figure 6 further summarizes the relative ranking of each decoding method under every model-temperature setting, averaged over MT-Bench and AlpacaEval. Lower ranks indicate better performance. The heatmap shows that ME-Decoding obtains competitive ranks

Table 15: Accuracy and diversity comparison under different temperatures. Diversity Avg. is computed by first min-max normalizing Distinct-1, Distinct-2, and $1 - \text{Self-BLEU}$ within each temperature, and then averaging the three normalized scores.

T	Method	Acc ↑	Distinct-1 ↑	Distinct-2 ↑	$1 - \text{Self-BLEU}$ ↑	Diversity Avg. ↑
1.0	Top-p	0.8484	0.0050	0.0695	0.2570	0.9842
	Min-p	0.8596	0.0050	0.0705	0.2595	1.0000
	p-less	0.8906	0.0046	0.0445	0.0607	0.1012
	Top-H	0.8832	0.0045	0.0418	0.0614	0.0043
	Top-W	0.8996	0.0049	0.0480	0.0686	0.3550
	ME-Decoding ($\lambda = 0.9$)	0.9030	0.0049	0.0472	0.0588	0.3294
	ME-Decoding ($\lambda = 0.8$)	0.9008	0.0049	0.0504	0.0813	0.4039
	ME-Decoding ($\lambda = 0.7$)	0.9024	0.0049	0.0528	0.1014	0.4652
1.5	Top-p	0.6794	0.0224	0.1503	0.4136	1.0000
	Min-p	0.7802	0.0054	0.0849	0.3421	0.3884
	p-less	0.8682	0.0046	0.0546	0.1464	0.0783
	Top-H	0.8224	0.0047	0.0585	0.1974	0.1434
	Top-W	0.8982	0.0049	0.0542	0.1148	0.0516
	ME-Decoding ($\lambda = 0.9$)	0.9026	0.0052	0.0526	0.0735	0.0112
	ME-Decoding ($\lambda = 0.8$)	0.9058	0.0051	0.0564	0.1046	0.0528
	ME-Decoding ($\lambda = 0.7$)	0.8996	0.0053	0.0592	0.1268	0.0879
2.0	Top-p	0.1886	0.2494	0.7750	0.9054	1.0000
	Min-p	0.6600	0.0068	0.1122	0.4487	0.1833
	p-less	0.8128	0.0049	0.0659	0.2454	0.0786
	Top-H	0.7394	0.0053	0.0784	0.3209	0.1149
	Top-W	0.8906	0.0047	0.0543	0.1397	0.0310
	ME-Decoding ($\lambda = 0.9$)	0.9068	0.0050	0.0499	0.0668	0.0004
	ME-Decoding ($\lambda = 0.8$)	0.9036	0.0051	0.0553	0.0959	0.0146
	ME-Decoding ($\lambda = 0.7$)	0.9034	0.0052	0.0573	0.1187	0.0247

Dataset	T	ME-Decoding	Without ϵ	$K = I$
GSM8K	1.0	84.08	−2.20	−0.15
	1.5	84.23	−4.62	−0.83
	2.0	82.41	−8.19	−1.36
GPQA	1.0	34.38	−0.22	−0.67
	1.5	36.16	−1.56	−2.90
	2.0	33.93	−0.22	−0.89

Table 16: Ablation results under different temperatures. The ME-Decoding column reports absolute accuracy, while the other columns report relative changes compared with ME-Decoding.

in multiple settings and achieves the best overall average rank, suggesting that its improvement is stable across models and temperatures. Tables 18 and 19 report the exact MT-Bench judge scores and AlpacaEval candidate win-rates, respectively. All results are evaluated using the same judge-based evaluation protocol as in the main text and aggregated over 3 runs.

C.2 Detailed Complexity Analysis

Let N denote the size of the candidate token pool, d denote the token embedding dimension, and τ denote the number of tokens selected before the early stopping criterion is triggered. ME-Decoding does not explicitly construct the full $N \times N$ similarity matrix. The adaptive bandwidth can also be computed without any pairwise construction. Since we use normalized token embeddings and $\boldsymbol{C}_{ij} = 1 - \boldsymbol{e}_i^\top \boldsymbol{e}_j$, we have

$$\epsilon = \frac{1}{2} \sum_{i,j \in \mathcal{V}} \boldsymbol{p}_i \boldsymbol{p}_j \boldsymbol{C}_{ij} = \frac{1}{2} \left(1 - \left\| \sum_{i \in \mathcal{V}} \boldsymbol{p}_i \boldsymbol{e}_i \right\|_2^2 \right),$$

where p is normalized over the candidate pool. Therefore, computing ϵ only requires a probability-weighted average of token embeddings, with complexity $O(Nd)$.

After obtaining ϵ, kernel entries are dynamically computed according to Eq. 3. Specifically, whenever a new token is selected, the algorithm computes and caches its similarities to all candidate tokens, which costs $O(Nd)$ per selected token and

λ	Qwen3-4B-Inst.			Phi-4-mini-Inst.			Mistral-7B-Inst.			Avg.
	$T=1.0$	$T=1.5$	$T=2.0$	$T=1.0$	$T=1.5$	$T=2.0$	$T=1.0$	$T=1.5$	$T=2.0$	
GSM8K										
1.0	81.20	81.35	79.76	83.78	84.38	82.03	52.84	52.77	53.45	72.39
0.9	80.67	79.76	80.06	84.08	84.23	82.41	55.34	53.45	53.90	**72.66**
0.8	80.44	80.06	80.44	83.70	84.08	81.65	55.19	53.45	53.53	72.50
0.7	79.68	80.14	79.98	84.15	83.55	82.11	53.45	54.51	53.75	72.37
GPQA										
1.0	34.82	34.15	34.82	35.04	35.27	35.71	27.90	28.79	27.23	32.64
0.9	35.49	35.49	35.71	34.38	36.16	33.93	28.35	28.79	28.35	**32.96**
0.8	36.16	36.38	35.27	35.27	33.26	34.15	27.90	25.67	27.68	32.42
0.7	36.38	35.04	34.60	33.71	31.92	33.93	27.90	29.24	26.34	32.12

Table 17: Detailed hyperparameter selection results for ME-Decoding. Accuracy is reported in percentage. The highlighted row corresponds to the selected setting, λ=0.9, which achieves the best average performance on both GSM8K and GPQA.

Method	Qwen3-4B-Inst.			Phi-4-mini-Inst.			Mistral-7B-Inst.			Avg.
	$T=1.0$	$T=1.5$	$T=2.0$	$T=1.0$	$T=1.5$	$T=2.0$	$T=1.0$	$T=1.5$	$T=2.0$	
Min-p	8.71	<u>8.73</u>	8.51	**6.19**	5.63	4.61	<u>6.69</u>	6.55	6.10	6.86
Top-p	<u>8.76</u>	8.69	8.47	5.79	4.85	2.37	6.63	6.27	5.19	6.34
p-less	8.70	8.62	<u>8.75</u>	5.91	5.91	5.15	6.42	6.47	<u>6.53</u>	6.94
Top-H	8.70	8.65	8.62	5.97	5.79	4.63	6.59	<u>6.70</u>	6.25	6.88
Top-W	8.62	8.72	8.64	<u>6.12</u>	**5.96**	**6.16**	6.44	6.55	6.51	<u>7.08</u>
Ours	**8.84**	**8.76**	**8.87**	5.93	<u>5.94</u>	<u>6.05</u>	**6.78**	**6.72**	**6.68**	**7.17**

Table 18: Detailed MT-Bench judge scores. Results are reported across three instruction-tuned models and three temperatures, aggregated over 3 runs. The Avg. column gives the unweighted average over all model-temperature combinations. Best results are shown in **bold**, and second-best results are <u>underlined</u>.

therefore $O(N\tau d)$ in total. For the greedy selection stage, at the t-th step, the selected set has size t, and the algorithm evaluates at most $N-t$ remaining candidates. For each candidate j, the dominant operation is computing $\boldsymbol{\alpha}_j = \mathbf{R}\boldsymbol{\beta}_j$, where $\mathbf{R} \in \mathbb{R}^{t\times t}$ and $\boldsymbol{\beta}_j \in \mathbb{R}^t$, which costs $O(t^2)$. Hence, the total greedy evaluation cost before stopping is

$$\sum_{t=1}^{\tau-1} O\big((N-t)t^2\big) \le \sum_{t=1}^{\tau-1} O(Nt^2) = O(N\tau^3).$$

After each selected token, updating $\mathbf{R}$ and z through the block update costs $O(t^2)$ at step t, giving an additional $O(\tau^3)$ cost, which is dominated by $O(N\tau^3)$ since $\tau \le N$. Therefore, the overall time complexity of ME-Decoding is

$$O(Nd + N\tau d + N\tau^3) = O\big(N\tau(d+\tau^2)\big).$$

The memory overhead is $O(N\tau)$ for cached kernel entries and $O(\tau^2)$ for maintaining $\mathbf{R}$. In contrast, explicitly constructing the full kernel matrix would require $O(N^2 d)$ time and $O(N^2)$ memory. Since the early stopping rule usually yields $\tau \ll N$, ME-Decoding avoids the quadratic dependence on the candidate pool size and remains efficient in practice.

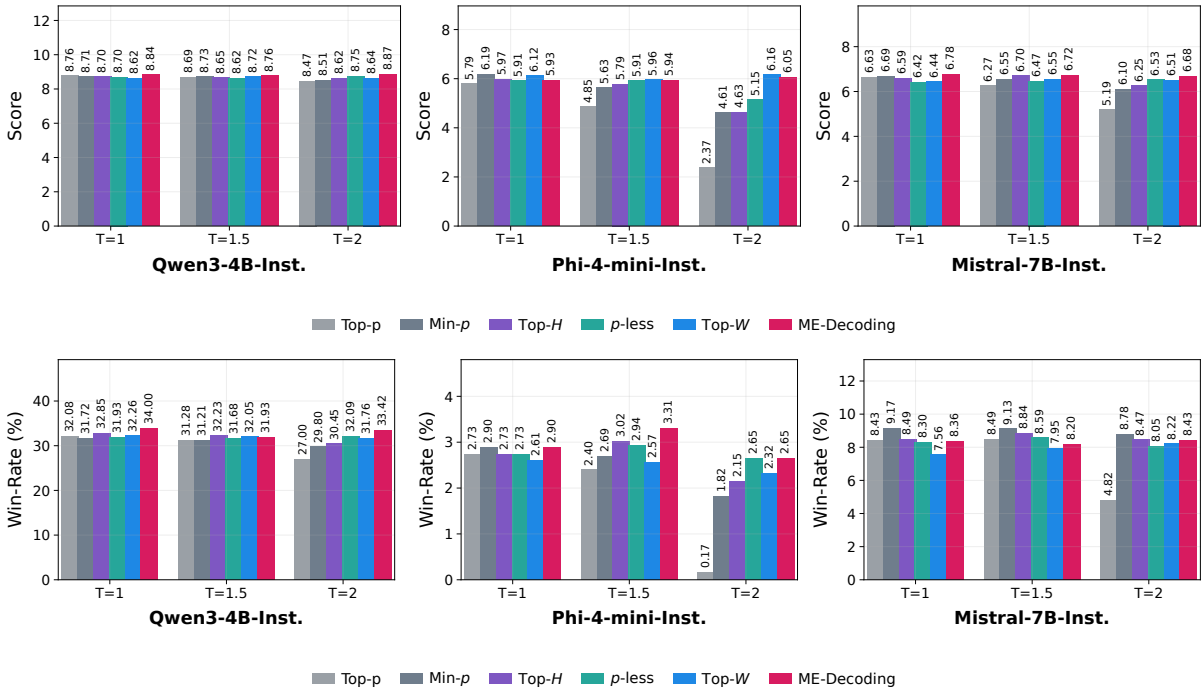

Figure 5: Detailed instruction-following and chat performance across temperatures. Top: MT-Bench judge scores. Bottom: AlpacaEval candidate win-rate. Results are reported for three instruction-tuned models and aggregated over 3 runs.

	Qwen3-4B-Inst.			Phi-4-mini-Inst.			Mistral-7B-Inst.			
Method	$T = 1.0$	$T = 1.5$	$T = 2.0$	$T = 1.0$	$T = 1.5$	$T = 2.0$	$T = 1.0$	$T = 1.5$	$T = 2.0$	Avg.
Min-p	31.72	31.21	29.80	**2.90**	2.69	1.82	**9.17**	**9.13**	**8.78**	14.14
Top-p	32.08	31.28	27.00	2.73	2.40	0.17	8.43	8.49	4.82	13.04
p-less	31.93	31.68	32.09	2.73	2.94	**2.65**	8.30	8.59	8.05	14.33
Top-H	32.85	**32.23**	30.45	2.73	3.02	2.15	8.49	8.84	8.47	14.36
Top-W	32.26	32.05	31.76	2.61	2.57	2.32	7.56	7.95	8.22	14.14
Ours	**34.00**	31.93	**33.42**	**2.90**	**3.31**	**2.65**	8.36	8.20	8.43	**14.80**

Table 19: Detailed AlpacaEval candidate win-rate (%). Results are reported across three instruction-tuned models and three temperatures, aggregated over 3 runs. The Avg. column gives the unweighted average over all model-temperature combinations. Best results are shown in **bold**, and second-best results are underlined.

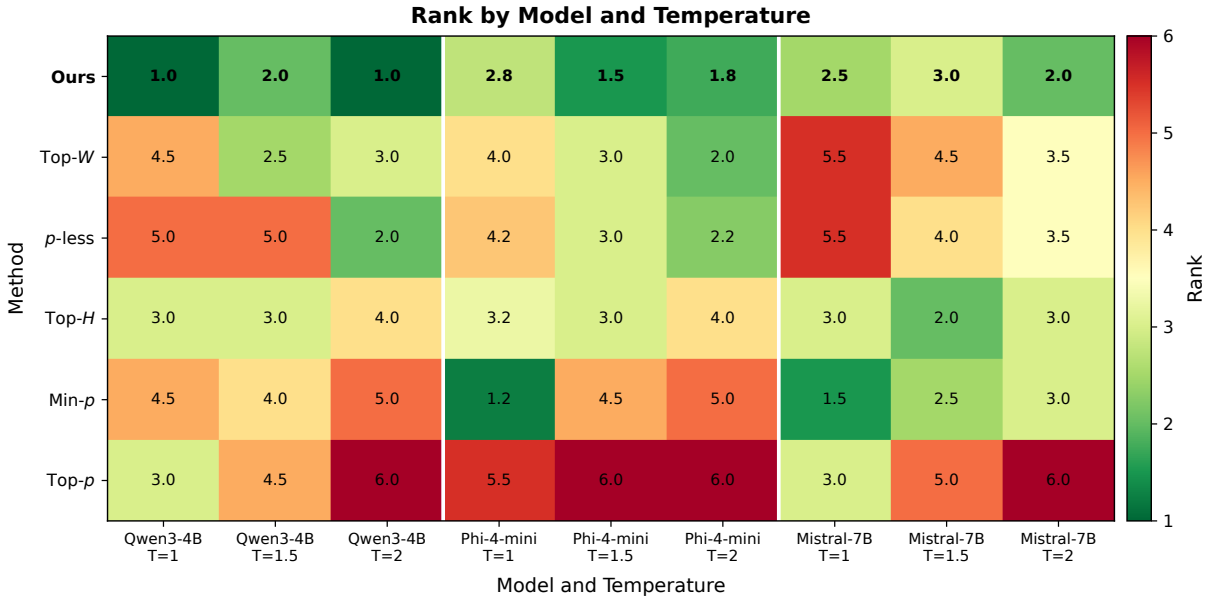

Figure 6: Detailed average-rank comparison on open-ended generation benchmarks. The heatmap reports the rank of each decoding method under each model-temperature setting, averaged over AlpacaEval and MT-Bench.