Learning Linear Attention in Polynomial Time

Convex regression, identifiable functions, and bounded computation

Paper by Morris Yau, Ekin Akyürek, Jiayuan Mao,
Joshua B. Tenenbaum, Stefanie Jegelka, Jacob Andreas

arXiv:2410.10101v4 (v4, 23 Oct 2025)
Research group discussion, 3 Nov 2026

Published at NeurIPS 2025 Main Conference.
No verifiable public implementation located (3 Nov 2026).

Expressibility leaves a learning problem open

The gap

A hand-designed Transformer can simulate a computation; examples do not tell us how to recover that computation efficiently.

- Nonconvex attention parameters obscure the global optimum.
- Multiple functions may fit the observed inputs equally well.
- The paper asks for an efficient learner and a data-dependent test of functional uniqueness.

Yau et al., Sec. 1, PDF pp. 1–2.

Three claims form the paper's argument

1. **Learning:** linearize a single-layer multi-head linear attention model, solve regression, recover heads by SVD.
2. **Identification:** a full-rank feature moment matrix makes all exact fits compute the same function.
3. **Application:** represent bounded computation histories autoregressively, then apply the learning and identification results.

Scope

Last-token squared-loss regression for one layer without softmax.

Yau et al., Secs. 2–3, PDF pp. 3–6.

The model is nonlinear in its parameters

Multi-head linear attention (MHLA)

$$Z \in \mathbb{R}^{d\times n}, \quad V_h, Q_h \in \mathbb{R}^{d\times d}, \quad \widehat{y}_\Theta(Z) = \sum_{h=1}^{H} V_h Z(Z^\top Q_h z_n).$$

- z_n is the last input token; $\widehat{y} \in \mathbb{R}^d$.
- Q_h fuses key and query projections; no softmax is applied.
- Variable sequence length n is allowed at fixed embedding dimension d.

Definition 2.1 and Eq. (5), PDF p. 3.

A fixed cubic feature map exposes linear prediction

$$S = ZZ^\top, \qquad \mathcal{X}(Z)_{j,(k,\ell)} = S_{jk} z_{\ell n} = \langle z_{j:}, z_{k:} \rangle z_{\ell n}.$$

$$\widehat{y}_a(Z) = \sum_{h,j,k,\ell} V_{h,aj} Q_{h,k\ell} \mathcal{X}(Z)_{j,(k,\ell)}.$$

- The input feature has fixed size $d \times d^2$, independent of n.
- The network is linear in the *combined* coefficients, although not in each head's matrices.

Eqs. (5), (7)–(8), PDF pp. 3–4.

Head count becomes a matrix-rank constraint

$$W_\Theta = \sum_{h=1}^{H} \mathrm{vec}(V_h)\,\mathrm{vec}(Q_h)^\top \in \mathbb{R}^{d^2 \times d^2}, \qquad \mathrm{rank}(W_\Theta) \le H.$$

Key step

Allow an unrestricted W. Its regression objective is convex, and any resulting W has rank at most d^2.

- SVD expresses W as at most d^2 outer products.
- Folding each pair of vectors yields attention heads with exactly the same input–output function.

Sec. 2.1, PDF p. 4; Algorithm 1, PDF p. 3.

Algorithm 1: regression followed by SVD

1. From each sequence, compute $\mathcal{X}(Z_i)$ and the output-coordinate features $X_{i,a}$.
2. Solve ordinary least squares:

$$\widehat{W} \in \arg\min_{W} \sum_{i=1}^{N} \sum_{a=1}^{d} \big(\langle W, X_{i,a} \rangle - y_{i,a}\big)^2.$$

3. Factor $\widehat{W} = \sum_{h=1}^{\widehat{H}} u_h v_h^\top$; fold u_h, v_h into $\widehat{V}_h, \widehat{Q}_h$.

Result: $\widehat{H} \le d^2$ and the recovered MHLA predicts exactly as $\widehat{W}$.

Algorithm 1, PDF p. 3; proof in Appendix D, pp. 17–18.

Theorem 2.2 states an agnostic risk bound

As stated in the paper

With i.i.d. samples and squared loss, the returned $\widehat{\Theta}$ satisfies, with probability at least $1-\delta$,

$$R(\widehat{\Theta}) - \min_{\Theta \in \Omega_H} R(\Theta) \le \epsilon, \qquad R(\Theta) = \mathbb{E}\|\widehat{y}_\Theta(Z) - y\|^2.$$

$$N = O\left(\frac{d^4 + \log(1/\delta)}{\epsilon}\right), \qquad T = O\left(\frac{N d^4 n_{\max}}{\epsilon}\right).$$

No realizable-label assumption; output width may grow to d^2 heads.

Theorem 2.2, PDF pp. 3–4. Statistical assumptions are revisited later.

A low training loss does not identify a function

Question

Can two empirical minimizers agree on every training sequence but disagree on a new sequence?

- Raw (V_h, Q_h) values are redundant: rescaling and reallocation across heads can preserve predictions.
- The relevant object is a coefficient vector $p(\Theta)$ in a *nonredundant* input-feature basis.
- Identification asks whether the dataset determines that coefficient vector.

Sec. 2.2, PDF p. 5; Appendix A, pp. 13–14.

The certificate checks the rank of cubic features

$$\mathcal{H}(Z)_{\{j,k\},\ell} = \langle z_{j:}, z_{k:} \rangle z_{\ell n}, \qquad j \le k,$$

$$\psi = \binom{d}{2} d + d^2 = \frac{d^2(d+1)}{2}, \qquad \Lambda_D = \frac{1}{N} \sum_{i=1}^{N} \mathcal{H}(Z_i)\mathcal{H}(Z_i)^\top.$$

Certificate

$\lambda_{\min}(\Lambda_D) > 0$: the observed features span the whole relevant feature space.

At least $N \ge \psi$ samples are necessary for this empirical matrix to be full rank.

Lemma 2.3, PDF p. 5; Algorithm 2 and Lemma A.3, pp. 13–14.

Full rank turns exact fit into equality everywhere

1. For each output coordinate, $\widehat{y}_{\Theta,a}(Z) = \langle p_a(\Theta), \mathcal{H}(Z) \rangle$.
2. If data are generated by $\bar{\Theta}$, every empirical minimizer has zero residual.
3. For $\Delta p_a = p_a(\Theta) - p_a(\bar{\Theta})$:

$$0 = \frac{1}{N} \sum_i \langle \Delta p_a, \mathcal{H}(Z_i) \rangle^2 = \Delta p_a^\top \Lambda_D \Delta p_a \geq \lambda_{\min}(\Lambda_D) \|\Delta p_a\|^2.$$

4. Hence $\Delta p_a = 0$, so the two models agree on *all* input sequences.

Lemma A.3 and proof, PDF pp. 13–14, 19.

Realizable and agnostic identification differ

Realizable data

Full rank + exact fit implies every empirical minimizer equals the generating function on all inputs.

Arbitrary labels

With at least d^2 heads, full rank makes the least-squares *optimal function* unique; no ground-truth function is assumed.

Small $\lambda_{\min}$ also signals sensitivity to finite fitting error and numerical noise.

Lemmas A.3 and A.6, PDF pp. 13–15, 20.

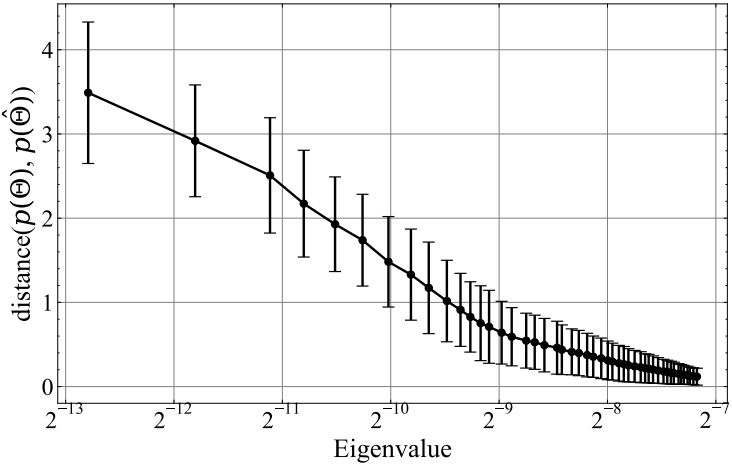
Associative memory supplies an interpretable test

$$Z = \begin{bmatrix} k_1 & \cdots & k_d & q \\ v_1 & \cdots & v_d & \zeta \end{bmatrix}, \qquad y = \begin{bmatrix} 0 \\ \sum_{j=1}^{d} \langle q, k_j \rangle v_j \end{bmatrix}.$$

- Gaussian key/value vectors create identifiable examples; orthogonal key/value tables create rank-deficient ones.
- Mix the two sources to vary $\lambda_{\min}(\Lambda_D)$.
- Compare models using $\|p(\Theta) - p(\widehat{\Theta})\|_F$: distance between *functions*, not raw weights.

Sec. 4.2, Eqs. (13)–(15), PDF pp. 7–8. $d = 4$, $N = 2^{14}$.

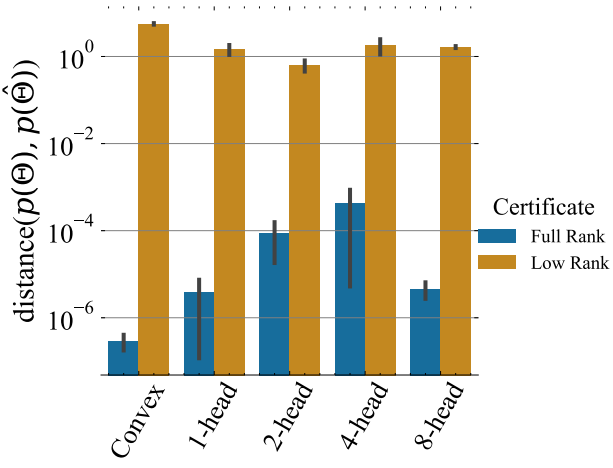
A larger certificate tracks better recovery



Algorithm 1: minimum eigenvalue increases, distance in p-space falls.

Original Fig. 2a, PDF p. 8. The vertical axis is function-coefficient distance, not test error.

Gradient-trained heads separate by certificate rank



Full rank: $\lambda_{\min} = 0.06$; low rank: $\lambda_{\min} = 0$.

Original Fig. 2b, PDF p. 8. Logarithmic distance axis; error bars: standard error over 3 runs.

Synthetic regression probes width versus depth

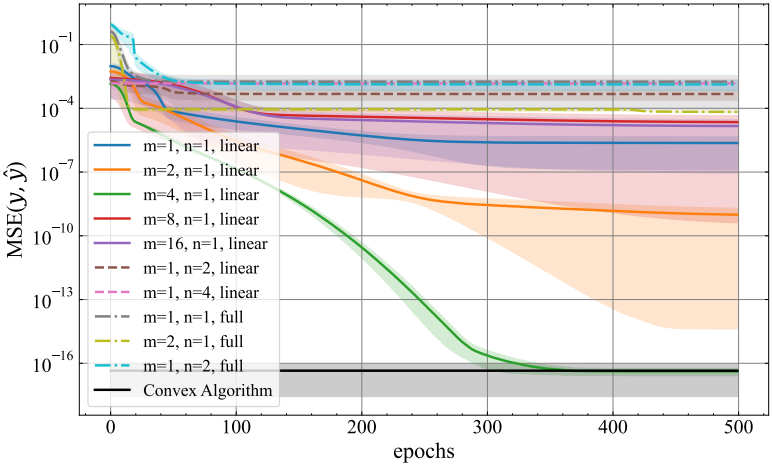
- A randomly initialized one-layer linear-attention teacher generates targets from Gaussian sequences of length $T = 100$.
- Compare multi-head linear attention, single-head multi-layer attention, a full Transformer, and Algorithm 1.
- Figure 1 reports mean squared prediction error against epochs; lower is better.

Terminology in the paper

The body says "SGD"; Appendix G.1 lists AdamW as the training optimizer.

Sec. 4.1, PDF p. 6; Appendix G.1, p. 29.

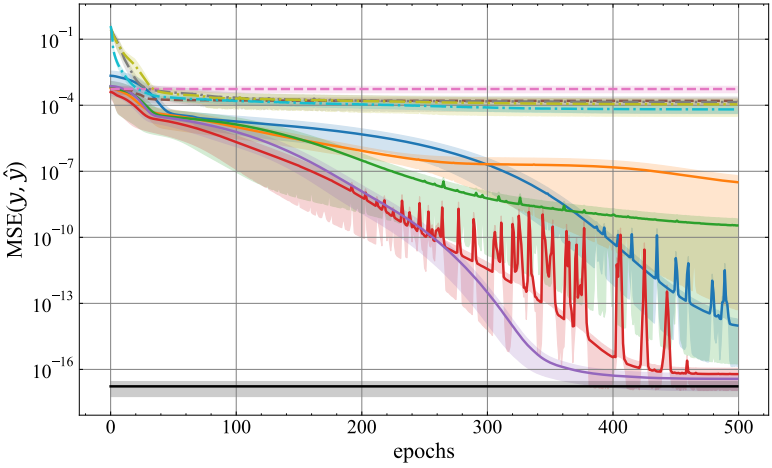
$d = 2$: four heads reach the convex baseline



$N = 512$; m is head count and n is layer count. MSE is on a log scale.

Original Fig. 1a, PDF p. 7. Shading: standard error over 3 runs.

$d = 4$: sixteen heads converge near numerical precision



$N = 2048$; the $m = d^2 = 16$ curve reaches the convex baseline.

Original Fig. 1b, PDF p. 7. Other widths can also improve; more heads are not monotonic in every run.

A bounded Turing-machine history can be generated

1. Encode a machine's transition table and input as orthogonal token and position vectors.
2. Use attention heads for lookup, copy, and conditional updates.
3. Autoregressively write each new tape/state snapshot as another history segment.

Construction, not an optimization experiment

The same one-layer MHLA module is reused at each generation step; embedding width scales with the predeclared computation bounds.

Sec. 3, PDF pp. 5–6; Appendices B and F, pp. 16, 23–28.

The UTM result has explicit size conditions

$$H = O(\widehat{n}\widehat{\Phi}\widehat{\Sigma}), \qquad d = O\big(\widehat{n}\widehat{\Phi}\widehat{\Sigma}\max(\widehat{\Sigma}, \widehat{\mathcal{Q}})\big).$$

- Bounds cover input length $\widehat{n}$, steps $\widehat{\Phi}$, alphabet $\widehat{\Sigma}$, and states $\widehat{\mathcal{Q}}$.
- Distributional trajectory learning uses a union bound across generated tokens.
- With an exact fit and the full-rank certificate, functional equality extends to every input *within the bounded class*.

Lemmas B.1, C.5, A.8; PDF pp. 15–17.

The DFA appendix tests a different architecture

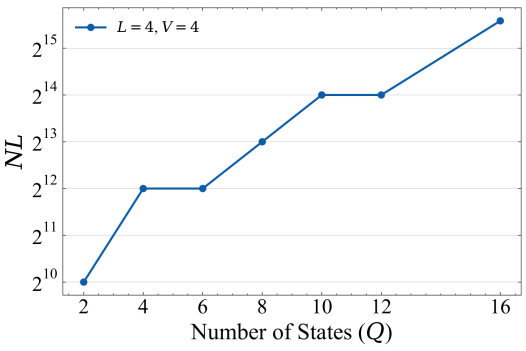
Task

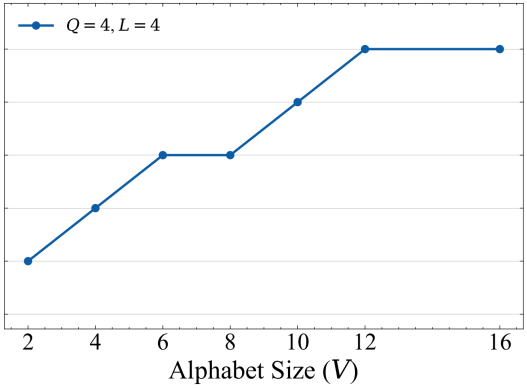
Input: DFA transition table + word. Target: autoregressive state-transition history.

- Varied: state count Q, word length L, or alphabet size V; the other two equal 4.
- Measure: training tokens NL needed for 99% next-token accuracy; lower is better.
- Trained model: a 4-layer, 16-head, width-2048 Llama-style *full Transformer*.

Appendix G.2 and G.4, PDF pp. 30–31; Fig. 4, p. 31.

DFA complexity raises the observed data demand

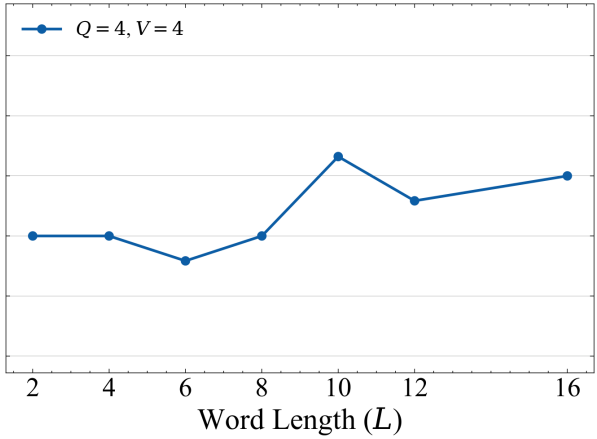




State count and alphabet size generally increase the tokens needed for 99% accuracy.

Cropped from original Fig. 4, PDF p. 31. These are full-Transformer results.

Word length shows a less monotonic DFA trend



For $Q = V = 4$, the required tokens NL fluctuate across tested word lengths.

Cropped from original Fig. 4, PDF p. 31. A finite plotted range does not establish an asymptotic rate.

Read the guarantees at their precise scope

- Convex regression solves the *expanded* class: up to d^2 heads, one layer, linear attention, squared loss.
- The all-input certificate needs full feature rank and an exact realizable fit; arbitrary labels yield uniqueness of the optimal function only in the wide class.
- Theorem 2.2 states a distribution-free squared-risk sample rate, but Appendix D only invokes generic VC theory; bounded-loss or moment conditions needed for that statistical step are not specified.
- Figure 4 studies a full Transformer, so it is a separate empirical illustration of DFA learning.

Theorem 2.2, PDF pp. 3–4; Appendix D, pp. 17–18; Lemma A.6, p. 20; Appendix G.4, p. 31.

Take-home messages

1. A cubic feature map converts MHLA prediction into linear regression; SVD turns the solution into at most d^2 attention heads.
2. A full-rank moment matrix certifies that exact fits are the same function on every input.
3. Bounded automaton computations connect these two ideas to algorithm learning; synthetic experiments probe their mechanisms.

Research direction

Keep a checkable function certificate while extending beyond single-layer ungated linear attention.

Yau et al., Secs. 2–4 and 6, PDF pp. 3–9.

Paper and publication links

- Morris Yau et al., *Learning Linear Attention in Polynomial Time*. arXiv:2410.10101v4 (v4; this presentation uses the arXiv PDF).
- NeurIPS 2025 Main Conference, *Advances in Neural Information Processing Systems 38*; DOI: 10.52202/085713-5109.
- No verifiable public implementation located as of 3 Nov 2026.

Figures 1, 2, and 4 are reproduced from the arXiv v4 author source and labeled at their original PDF pages.