Rotary Positional Embeddings for Length Generalization in Decision Transformers

Lasse von Danwitz

Implementing Transformers Course
Winter Semester 2025/26

February 2026

Outline

The Length Generalization Problem

Decision Transformers (DT)

- Frame RL as sequence modeling.
- **Strength:** Excellent in-distribution performance.
- **Weakness:** Struggle to generalize to longer trajectories.

Research Question

Can **Rotary Positional Embeddings (RoPE)** enable robust length generalization where absolute encodings fail?

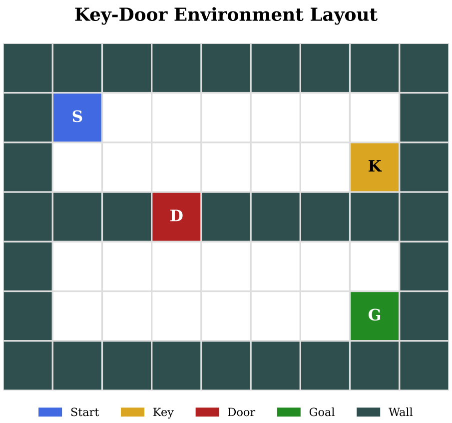


Figure: Key-Door Maze Task:
Collect key → Unlock door → Goal

Motivation: Absolute vs. Relative

The Core Challenge

Traditional encodings bind behavior to specific indices ($t = 1, t = 2 \ldots$). This rigid association breaks when inference exceeds the training horizon ($t > T_{\text{train}}$).

Absolute Positioning

- **Mechanism:** Unique vector for position t.
- **Flaw:** No definition for $t > T_{\text{max}}$.
- **Result:** Model is "lost" in unseen time steps.

Relative Positioning (RoPE)

- **Mechanism:** Encodes position via rotation.
- **Benefit:** Attention depends on distance ($n - m$).
- **Result:** Translation invariant $\rightarrow$ Robust extrapolation.

Positional Encoding Strategies

1. Sinusoidal (Absolute)

$$PE(m, 2i) = \sin(m/10000^{2i/d})$$
$$PE(m, 2i+1) = \cos(m/10000^{2i/d})$$

- Additive to input embeddings.
- Deterministic, but fixed to specific indices.

2. RoPE (Relative)

$$R_{\theta,m} = \begin{pmatrix} \cos(m\theta) & -\sin(m\theta) \\ \sin(m\theta) & \cos(m\theta) \end{pmatrix}$$

- Multiplicative on Query/Key vectors.
- Encodes relative distance.

Key RoPE Property

$$q_m'^T k_n' = (R_{\theta,m} q)^T (R_{\theta,n} k) = q^T R_{\theta,(n-m)} k$$

Attention score depends **only** on relative distance $(n - m)$.

Experimental Design

Environment & Task

- **Domain:** Key-Door Maze.
- **Training:** 8×8 grids *only*.
- **Eval:** Extrapolation up to 20×20.
- **Data:** 5,000 trajectories (70% optimal).

Architecture (DT)

- 8 Layers, 10 Heads, 320 Emb Dim.
- Context Window: 30 timesteps.

Compared Variants

1. **Baseline:** Learned absolute embeddings.
2. **Sinus:** Fixed sinusoidal embeddings.
3. **RoPE:** Rotary embeddings.

Protocol

- 3 Random Seeds.
- 100 Epochs (AdamW).
- Metric: Success Rate (100 eps/eval).

Note: All models use sinusoidal encoding for global timestep; only context position varies.

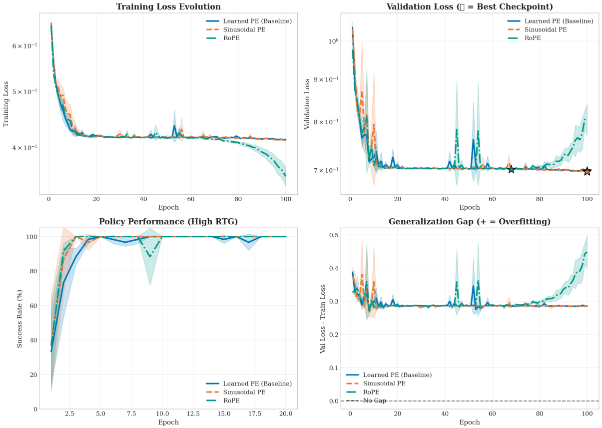
Training: The Loss-Generalization Disconnect

Table: Training Metrics (8×8 Grid)

Model	Loss	Val	Succ.%
Baseline	0.413	0.698	97.3
Sinus	0.411	**0.697**	98.7
RoPE	**0.356**	0.805	**99.3**

Key Observation

- All models converge well.
- Validation loss **fails** to predict extrapolation capability.
- Models may memorize coordinates rather than learning navigation.



Main Result: Length Generalization

Table: Success Rates (Mean $\pm$ Std, 3 seeds). **Bold** indicates best performance.

Grid Size	**Baseline**	**Sinus**	**RoPE**	**Δ vs Base**
8×8 (Train)	100.0	100.0	100.0	-
10×10	64.0 ± 25	40.0 ± 41	86.0 ± 12	$+22.0$
12×12	18.3 ± 20	1.7 ± 1	43.3 ± 18	$+25.0$
15×15	1.3 ± 0.9	0.0 ± 0	13.7 ± 13	$+12.4$
20×20	2.3 ± 3.3	0.0 ± 0	5.3 ± 5.6	$+3.0$

RoPE Advantage

- **Significant gains** on 10×10 and 12×12.
- Maintains non-zero success at 20×20.

Stability

- Baseline $\sigma \approx 25.2\%$ (Unstable).
- RoPE $\sigma \approx 12.1\%$ (More robust).

Visualizing Extrapolation

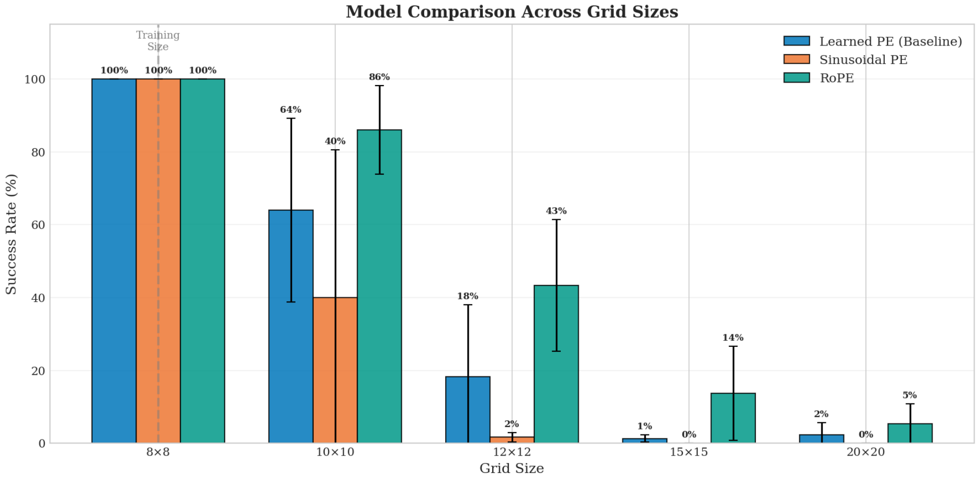


Figure: RoPE (Blue) maintains substantially higher success rates beyond the training distribution compared to Baseline (Orange) and Sinusoidal (Green).

Mechanism: Context vs. Position

Hypothesis: Does RoPE simply handle more tokens better?

Table: Avg Success Rate across Grid Sizes

Context	**Sinus**	**RoPE**	Δ
30 (1.0×)	28.7	50.1	+21.3
45 (1.5×)	30.0	50.2	+21.2
60 (2.0×)	29.6	50.1	+20.5
90 (3.0×)	29.7	50.1	+20.4

Finding

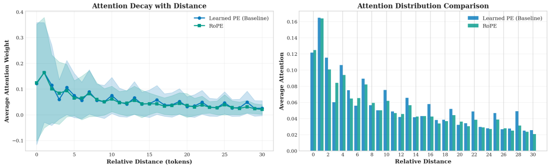
- RoPE advantage is **constant** ($\sim$ 21%) across context lengths.
- Performance does not degrade with longer context.

Conclusion

- Benefit is **Position Extrapolation**, not Context Capacity.
- RoPE transfers causal navigation rules to new indices.

Attention & Statistics

Attention Patterns



- **RoPE:** Consistent banded patterns (relative offsets).
- **Baseline:** Position-specific; fails beyond training.
- **Trans. Inv:** Mean Absolute Error $< 3\%$.

Statistical Significance

Grid	Cohen's d	p-value	Sig.
10×10	1.23	0.208	No
12×12	**2.56**	**0.035**	**Yes**
15×15	1.21	0.213	No

- Large effect sizes ($d > 0.8$) across board.
- Significance limited by seed count ($N = 3$).

Why RoPE Works

Structural Guarantees

- Enforces $Attention = f(m - n)$.
- Absolute indices are never explicitly computed.
- Rules like "key seen 2 steps ago" transfer perfectly.

The Role of Global Time

- Global t uses Sinusoidal (exploration horizon).
- Context uses RoPE (local navigation).
- **Result:** Decouples exploration from immediate navigation logic.

Synthesis

Absolute encodings link behavior to specific training positions (overfitting to t).

RoPE's relative structure prevents this overfitting, allowing learned navigation strategies to extrapolate seamlessly.

Limitations & Future Work

Current Limitations

1. **Statistical Power:** 3 seeds is insufficient for high confidence ($p > 0.05$).
2. **Encoder Bottleneck:** 20×20 grid is $6\times$ larger than 8×8. State aliasing is likely.
3. **Task Specificity:** Results currently limited to Grid World.

Future Directions

- **Architecture:** Scale-aware encoders / Multi-resolution processing.
- **RoPE Variants:** Apply to global timesteps; learnable frequency bases.
- **Domains:** Continuous control & Multi-agent RL.

Positional encoding is necessary but not sufficient; spatial resolution must also scale.

Summary

1. **Generalization Hierarchy:**
   - RoPE: **86%** (10×10)
   - Baseline: 64%
   - Sinusoidal: 40%

2. **Stability vs. Variance:**
   - Learned embeddings are unstable ($\pm$25%).
   - RoPE is robust and consistent ($\pm$12%).

3. **Mechanism:**
   - Success driven by **position extrapolation**, not context capacity.
   - Structural enforcement of relative distance is key.

Final Takeaway

RoPE is the optimal choice for Decision Transformers requiring robust length generalization, eliminating the bottleneck of absolute positioning.

Thank You

Questions?

Rotary Positional Embeddings for Length Generalization in Decision Transformers

Backup: Path Efficiency

Table: Average Steps to Goal (Efficiency Ratio = RoPE/Baseline)

Grid	Baseline	Sinus	RoPE	Efficiency
8×8	12.6	12.6	13.1	1.04
10×10	30.5	38.5	**22.3**	1.4
12×12	56.0	59.7	**49.1**	2.2
15×15	74.6	75.0	**70.0**	2.4
20×20	98.5	100.0	**98.7**	2.4

Note: RoPE models that succeed do so efficiently. Baselines often succeed via inefficient random walks on larger grids.

Backup: Decision Transformer Formulation

Trajectory Representation

$$\tau = (\hat{R}_1, s_1, a_1, \ldots, \hat{R}_T, s_T, a_T)$$

where $\hat{R}_t$ is return-to-go, s_t is state, and a_t is action.

Standard Input Construction

$$\mathrm{x}_t = \mathrm{Embed}(\mathrm{modality}_t) + \mathrm{Embed}_{\mathrm{time}}(t)$$

Problem

Explicit absolute embeddings hinder generalization when t exceeds training ranges. This is precisely where RoPE provides its advantage.