Diffuser

Efficient Transformers with Multi-hop Attention Diffusion for Long Sequences

Paper authors: Aosong Feng, Irene Li, Yuang Jiang, Rex Ying

Yale University

Research group discussion
arXiv:2210.11794v2 — 31 January 2023
Reading prepared: 9 October 2026

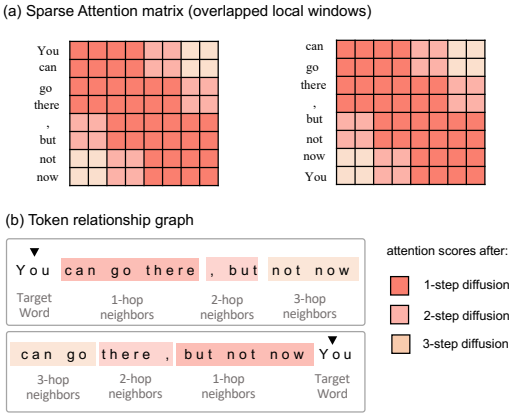
Compute sparse attention once; propagate values several times.

Sparse edges can delay important information

- A sparse attention layer directly reads only its allowed neighbors.
- Distant cues may require several network layers to arrive.
- Fixed block boundaries also change the routing after a small input shift.

Key point

Can one layer reach farther while retaining sparse computation?

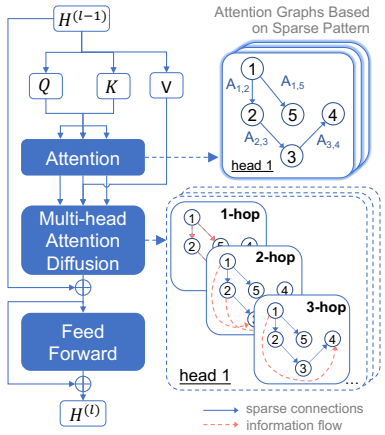


Introduction; Fig. S.1, PDF pp. 1–2.

One set of edge weights supports repeated propagation

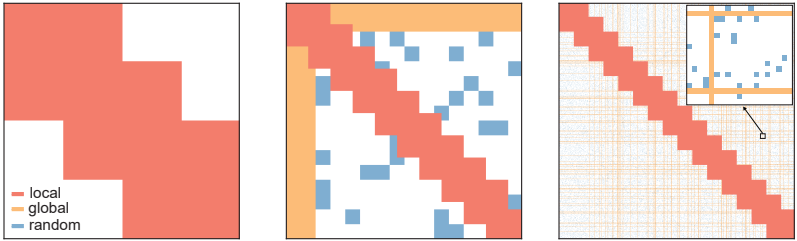
1. Project the input into Q, K, V.
2. Compute attention scores on sparse edges.
3. Reuse those scores for K propagation steps.
4. Apply output projection, residuals and the feed-forward module.

No new query–key scores are computed between diffusion steps.



Model Architecture; Fig. S.3, PDF p. 4.

Local, global and random edges serve different roles



(a) Longformer (b) BigBird (c) Diffuser

Local windows
Preserve nearby structure; overlap adjacent windows.

Global tokens
Provide short routes across the entire sequence.

Random edges
Distribute long-range routes across individual tokens.

Global and random selections use individual tokens, unlike BigBird's blocks.

Sparse Pattern Design; Fig. S.2, PDF pp. 3–4.

Multi-hop weights are products along attention paths

$$A_{ij} = \frac{\exp(Q_i K_j^\top / \sqrt{d})}{\sum_{u \in Ne(i)} \exp(Q_i K_u^\top / \sqrt{d})} \quad (j \in Ne(i)); \qquad A_{ij} = 0 \text{ otherwise.}$$

- $Ne(i)$ is the allowed key set; every row of A sums to one.
- Q, K have n rows; $V \in \mathbb{R}^{n \times d_v}$.

$$(A^2)_{ij} = \sum_u A_{iu} A_{uj}, \qquad \mathcal{A} = \sum_{k=0}^{\infty} \theta_k A^k.$$

Key point

An indirect relationship is composed from existing edges. It need not equal the missing direct query–key score.

Eqs. S.5–S.6, PDF p. 3. $\theta_k \geq 0$ and $\sum_k \theta_k = 1$.

PPR computes diffusion through sparse matrix–value products

$$\boxed{Z_{(0)} = V, \qquad Z_{(k+1)} = (1-\alpha)AZ_{(k)} + \alpha V}$$

- K steps per layer; the original V is injected at every step.
- α controls how much the output retains the source values.
- Compute AZ directly with sparse kernels.

For $0 < \alpha \leq 1$, the limiting operator is

$$\mathcal{A} = \sum_{k=0}^{\infty} \alpha(1-\alpha)^k A^k = \alpha[I - (1-\alpha)A]^{-1}.$$

The implementation forms neither the inverse nor the dense diffusion matrix.

Eq. S.7, PDF p. 3; Appendix A, Proposition 4, PDF p. 11.

Five steps give a finite polynomial in the sparse attention

$$Z_{(K)} = \left[(1-\alpha)^K A^K + \alpha \sum_{i=0}^{K-1} (1-\alpha)^i A^i \right] V.$$

At $K = 5$ and $\alpha = 0.1$:

$$Z_{(5)} = \big(0.1I + 0.09A + 0.081A^2 + 0.0729A^3 + 0.06561A^4 + 0.59049A^5\big) V.$$

- Reach is limited to paths of at most K edges.
- Bidirectional global tokens make every pair reachable within two hops.
- Task saturation at five steps does not measure PPR approximation error.

Appendix A, PDF p. 11; coefficient expansion is a teaching derivation.

The cost depends on both edge count and propagation steps

For one head, with $|\mathcal{E}|$ retained edges:

$$\underbrace{O(|\mathcal{E}|d)}_{\text{sparse scores}} \quad + \quad \underbrace{O(K|\mathcal{E}|d_v)}_{\text{value propagation}} \ .$$

- Fixed local/global/random budgets give $|\mathcal{E}| = O(n)$.
- Linear attention-module cost also requires fixed K.
- Edge weights use $O(|\mathcal{E}|)$ storage; one state uses $O(nd_v)$.

Nonzero entries / n^2	Longformer	BigBird	Diffuser
$n = 4{,}096$	18.0%	16.9%	11.2%
$n = 8{,}192$	9.2%	8.7%	5.7%

Tables S.1/S.8, PDF pp. 4/12. Cost accounting excludes projections and FFNs; training may retain multiple states.

The theoretical claims concern different objects

Universal approximation

Construct sequence identifiers, then map them to outputs.
The appendix uses $K > n$ and a designated global token.

Expander graphs

A sparse regular graph approximates a *scaled* complete graph in Laplacian quadratic forms.

Fast mixing

A small nontrivial eigenvalue magnitude gives rapid random-walk mixing.

These results do not give an arbitrary learned full-attention error bound for the practical $K = 5$ model.

Theorems 1–3, PDF p. 5; Appendices C–D, PDF pp. 12–17.

Evaluation asks about quality, long-range structure and cost

Experiment	Setting	Metric
MLM	RoBERTa initialization; continue training at 4K	BPC ↓
Classification / QA	Five classification sets; WikiHop, TriviaQA	F1/Acc/EM ↑
Image density	CIFAR-10 and ImageNet-64 sequences	BPD ↓
LRA	Five text, reasoning and vision tasks	Acc ↑
Resource use	Sequence lengths 1K–4K	GB, sec ↓

Language configuration: 12 layers, width 768, 12 heads; $K = 5$, $\alpha = 0.1$.

Experiments, PDF pp. 6–7; Appendix E, Tables S.9–S.12, PDF pp. 17–18.

Changing the operator requires continued pretraining

Model / stage	MLM BPC ↓
RoBERTa	2.02
Longformer	1.86
BigBird	1.82
Diffuser, initialization	3.52
Diffuser, 10K steps	1.96
Diffuser, 50K steps	**1.68**

- Books, CC-News and Wikipedia.
- 15% masked tokens; maximum length 4,096.
- Initialization degrades before adaptation improves the result.
- At 50K steps: 0.14 lower BPC than BigBird.

Table S.2, PDF p. 6; Appendix E, PDF p. 17. BPC is the paper's MLM evaluation measure.

Text classification improves on average

Model	Five-task mean F1 ↑	A-2048 F1 ↑
BigBird	80.14	63.6
BigBird + diffusion	80.84	63.4
Longformer	80.86	61.2
Longformer + diffusion	81.24	62.5
Diffuser	**81.80**	**64.8**

- **+0.94 percentage points** over the original Longformer.
- +0.56 points over Longformer with diffusion.
- Individual winners differ: IMDB favors Longformer; 20NG favors its diffusion variant.

Table S.3, PDF p. 6. Mean over HYP, 20NG, IMDB, A-512 and A-2048; selected columns.

Question answering improves across the reported metrics

Model	WikiHop Acc	TriviaQA F1	TriviaQA EM
RoBERTa	71.82	74.02	66.87
Longformer	75.30	74.82	67.24
BigBird	74.54	73.16	68.26
Diffuser	**75.80**	**75.84**	**70.20**

- WikiHop selects an answer using evidence across documents.
- TriviaQA predicts an answer span.
- Gains over the best baseline in each column: **+0.50 / +1.02 / +1.94 percentage points**.

Table S.4, PDF p. 7; Appendix E, PDF p. 18. Higher is better for all three metrics.

Image density gains are small and dataset-dependent

Dataset	Sequence length	Sparse Trans.	Diffuser
CIFAR-10	3,072	2.80	**2.78**
ImageNet-64	12,288	**3.44**	**3.44**

Bits per dimension (BPD): lower is better.

- CIFAR-10: 0.02 BPD improvement.
- ImageNet-64: a tie at the reported precision.

Use the dataset-specific appendix configuration

CIFAR-10: 8 layers, width 512.
ImageNet-64: 20 layers, width 768.

Table S.5, PDF p. 7; Appendix E, PDF p. 18. The main text broadly describes an 8-layer/512 model.

LRA has three task wins and an inconsistent reported mean

Model	ListOps	Text	Retrieval	Image	Pathfinder
Transformer	36.37	64.27	57.46	42.44	71.40
BigBird	36.05	64.02	59.29	40.83	74.87
Diffuser	**37.52**	65.20	**61.28**	**45.20**	76.58

Recompute from the five listed scores

Diffuser: $57.156 \approx 57.16$; BigBird: $55.012 \approx 55.01$.
Improvement: **2.144 percentage points**.

- Paper reports 57.31 and +2.30 points; the arithmetic differs.
- Best Text: Linear Transformer (65.90).
  Best Pathfinder: Performer (77.05).

Table S.6, PDF p. 7. Bold marks Diffuser's wins over all listed baselines. The overall mean ranking is unchanged.

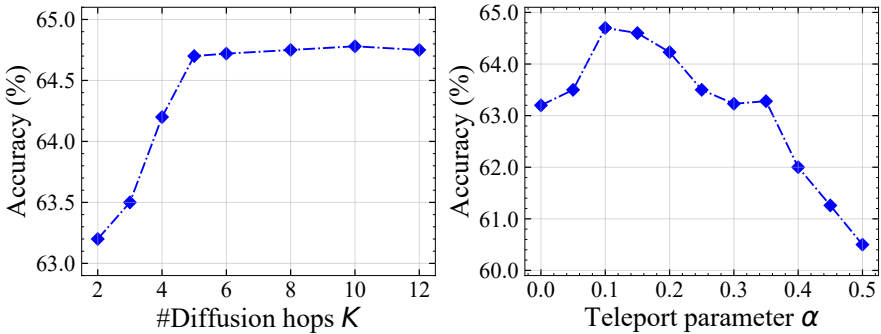
Removing diffusion or local edges hurts most

Variant	Recomputed LRA mean	Drop from full model
Full Diffuser	57.16	—
No local edges	52.65	4.51 points
No random edges	55.74	1.41 points
No global edges	55.19	1.97 points
No diffusion	52.36	**4.79 points**
Uniformly spaced edges	57.07	0.09 points

- *Uniform* changes edge placement, not attention weights.
- Distributed routes remain useful even without random placement.
- Removing edge types also changes the total edge budget.

Table S.7, PDF p. 7; Appendix B, PDF p. 12. Means and drops recomputed from the five task scores.

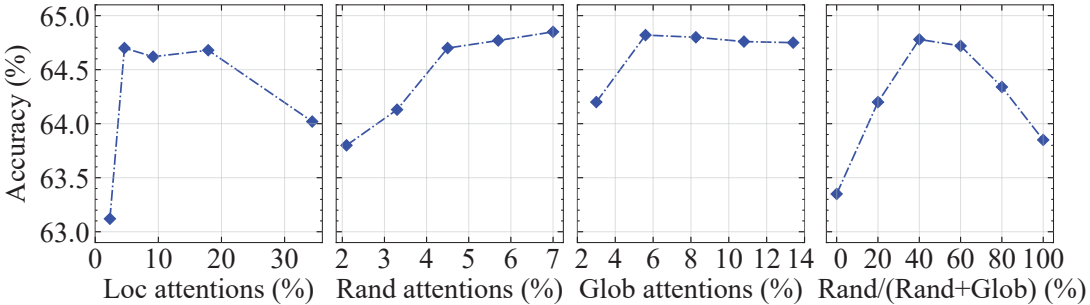
A small number of diffusion steps captures most task gains



- A-2048 performance plateaus around $K = 5$.
- Too much restart weakens propagation; the authors choose $\alpha = 0.1$.

Fig. S.5, PDF p. 7. Original y-axis: Accuracy (%); classification Table S.3 reports F1.

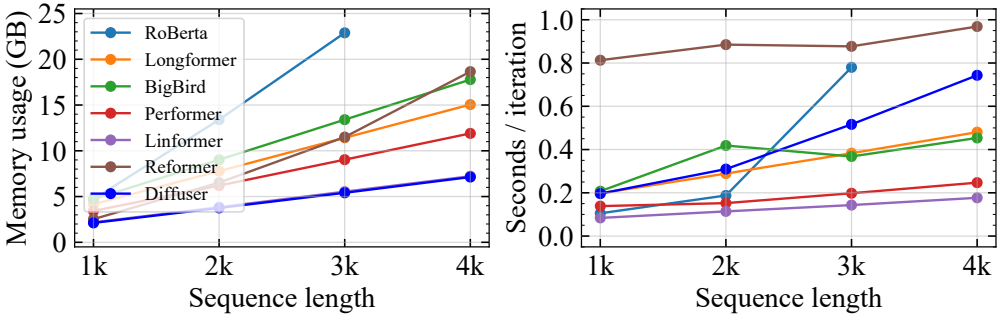
The mix of edge types matters as much as their presence



- More connections eventually saturate or reduce performance.
- A mixture of global and random routes beats either endpoint.
- The edge budget trades locality, coverage and compute.

Fig. S.6, PDF p. 7. Left to right: local, random, global budgets; random share of random + global edges.

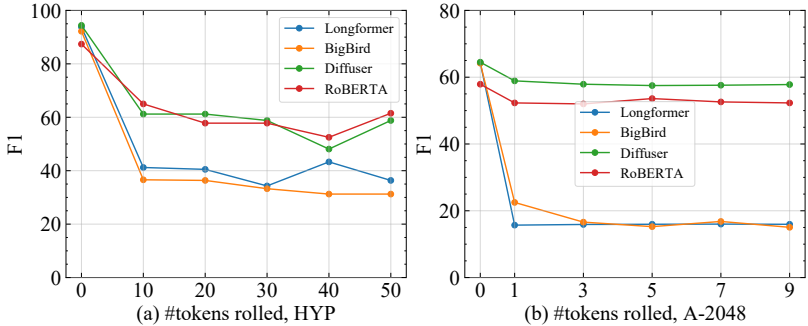
Low memory comes with a propagation-time cost



- At 4K: ∼7.2 GB vs. Performer's ∼12 GB; Linformer is close.
- Diffuser ∼0.75 sec/iteration vs. Performer ∼0.25.

Fig. S.4, PDF p. 6. Approximate plot readings. The claimed 1.67× memory ratio corresponds to about 40% less memory.

Diffuser is less sensitive to the tested input shifts



- Input tokens and masks are rolled together at length 4,096.
- The test changes both sparse routing and content–position alignment.

Appendix F; Fig. S.11, PDF pp. 18–19. HYP (left), A-2048 (right); y-axis is F1.

The useful idea is to separate scoring from propagation

Mechanism

Reuse sparse attention weights to aggregate values along multiple paths within one layer.

Evidence

Classification, QA and LRA improve overall; removing diffusion causes a large loss. Image-density gains are modest.

Tradeoff

Fewer stored edges help memory, while repeated aggregation costs time. Practical performance depends on the graph and sparse kernels.

Synthesis of Eqs. S.5–S.7, Tables S.2–S.7 and Fig. S.4, PDF pp. 3–7.

Paper and discussion prompts

Aosong Feng, Irene Li, Yuang Jiang, Rex Ying

Diffuser: Efficient Transformers with Multi-hop Attention Diffusion for Long Sequences

arXiv:2210.11794v2 (31 January 2023)
Author-linked implementation: github.com/asFeng/Diffuser

- At equal edge-access cost, when is diffusion better than adding edges?
- Which tasks need direct query–key scores that paths cannot preserve?
- How does the tradeoff change with optimized sparse kernels?

Figures reproduced from the paper's TeX assets. Source page numbers refer to the 19-page v2 PDF.
Main talk: approximately 20–30 minutes. Backup slides follow.

Backup: a three-token example exposes the mechanism

$$A = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 1 \end{bmatrix}, \qquad \alpha = 0.1.$$

$$Z_{(2)} = (0.1I + 0.09A + 0.81A^2)V.$$

For token 1:

$$\boxed{Z_{(2),1} = 0.3475\,V_1 + 0.45\,V_2 + 0.2025\,V_3.}$$

- Direct weight $A_{13} = 0$.
- The path through token 2 contributes $0.9^2 A_{12}A_{23} = 0.2025$.

Teaching example derived from Eq. S.7; not an experiment from the paper.

Backup: convergence is geometric, but five steps need not suffice

Let $Z_\star = \mathcal{A}V$ and $E_k = Z_{(k)} - Z_\star$. Then

$$E_{k+1} = (1-\alpha)AE_k, \qquad E_K = (1-\alpha)^K A^K (V - Z_\star).$$

Because A is row-stochastic, with $\|X\|_{\max} = \max_{i,j} |X_{ij}|$:

$$\|E_K\|_{\max} \leq (1-\alpha)^K \|V - Z_\star\|_{\max}.$$

- At $\alpha = 0.1$, $K = 5$, the worst-case factor is 0.59049.
- The actual error depends on A and V.
- Accuracy saturation and operator convergence are different measurements.

Appendix A, Proposition 4, PDF p. 11; explicit finite-step error bound derived for this reading.

Backup: what the universal-approximation construction uses

1. Approximate a continuous function on a compact domain by gridwise constants.
2. Quantize inputs, add positional identifiers and use hardmax heads to encode context.
3. Approximate the modified construction with ordinary softmax/FFN layers.

Conditions that matter

The main statement assumes self-loops and a Hamiltonian path.
The appendix additionally uses $K > n$ and a chosen global token.

Reducing the count of network layers does not eliminate the cost of propagation.

Theorem 1, PDF p. 5; Appendix C, especially PDF p. 15. The guarantee uses integrated L^q error.

Backup: spectral approximation needs a scaled complete graph

For an undirected, unweighted, d-regular expander:

$$|\mu_i| \le \epsilon d \quad (i \ge 2), \qquad H = \frac{d}{n} K_n.$$

$$(1-\epsilon) L_H \preceq L_G \preceq (1+\epsilon) L_H.$$

For the random-walk matrix $\widehat{A} = A/d$:

$$\|\widehat{A}^t v - u\|_1 \le \sqrt{n}\,\beta^t, \qquad \beta = \frac{\max(|\mu_2|, |\mu_n|)}{d} < 1.$$

The object here is graph topology under regularity assumptions, not an arbitrary content-weighted attention matrix.

Theorems 2–3, PDF p. 5; Appendix D, PDF pp. 16–17. Here d means degree, not hidden width.

Backup: the PPR operator preferentially preserves low frequencies

For $L = I - A$ with a real spectral decomposition:

$$\mathcal{A}: \quad h(\lambda) = \frac{\alpha}{\alpha + (1-\alpha)\lambda}, \qquad \mathcal{L} = I - \mathcal{A}: \quad \widetilde{\lambda} = 1 - h(\lambda).$$

- $h(0) = 1$: the constant mode is retained.
- $h(\lambda)$ decreases as frequency λ increases.
- This low-pass interpretation assumes a symmetric or reversible normalized graph.
- The feature response $h(\lambda)$ and the new Laplacian eigenvalue $\widetilde{\lambda}$ are different quantities.

Appendix A, Proposition 5 and Eq. S.10, PDF p. 11; response-form explanation derived for this reading.

Backup: dispersed random edges change the topology spectrum

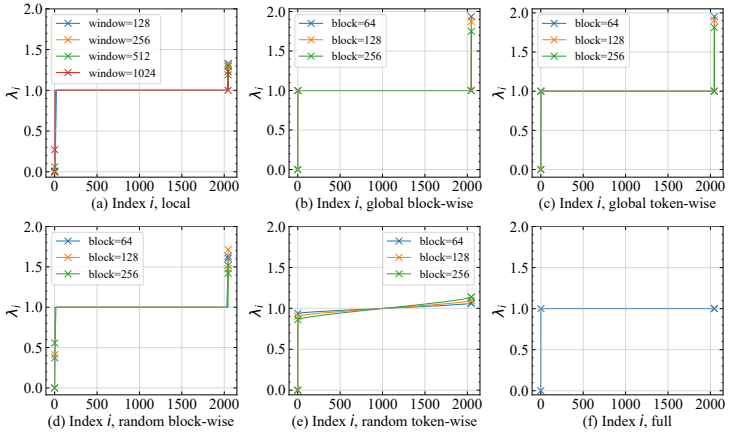


Fig. S.8, PDF p. 13. Normalized Laplacian eigenvalue vs. index; marked extreme eigenvalues are joined by lines.