Representational Strengths and Limitations of Transformers

Clayton Sanford

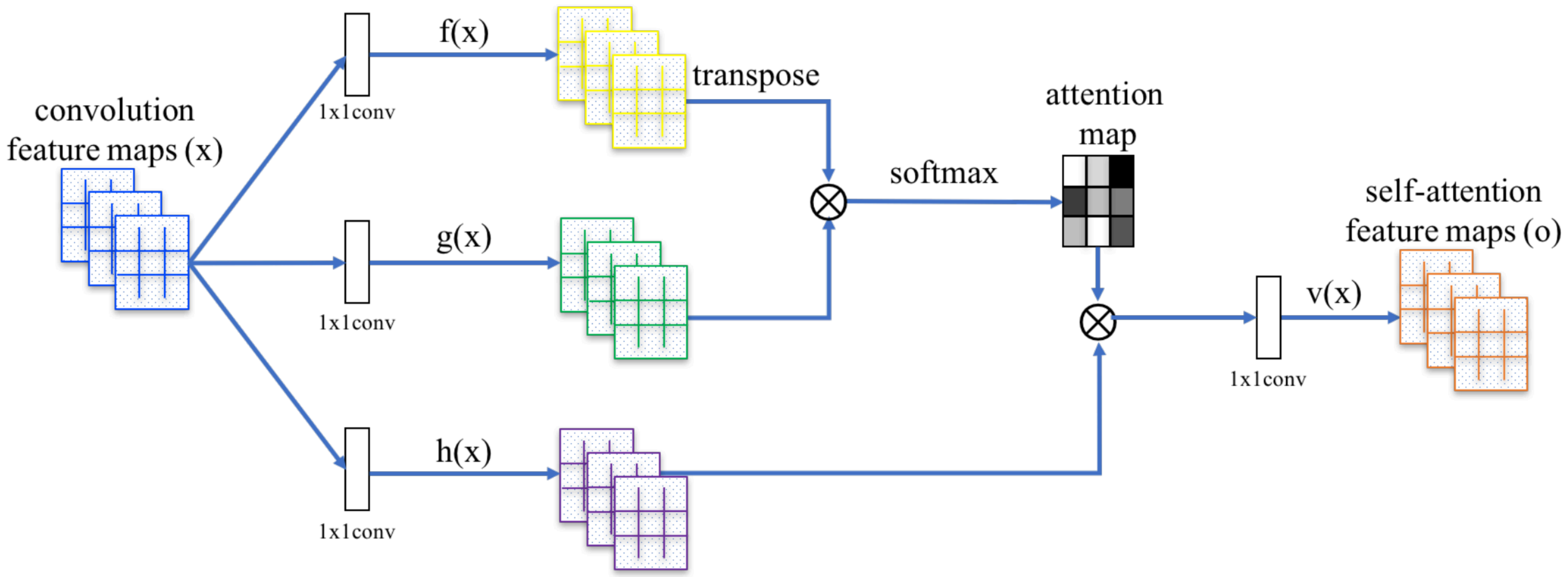
May 18th, 2023

Joint work with Daniel Hsu and Matus Telgarsky

Transformer architecture

What is it?

- **Self-attention unit:** $f(X) = \mathrm{softmax}(XQK^TX^T)XV$ for input $X \in \mathbb{R}^{N\times d}$, model parameters $Q, K, V \in \mathbb{R}^{d\times m}$.
- **Multi-headed attention:**
$$L(X) = X + \sum_{h=1}^{H} f_h(X)$$
- **Element-wise multi-layer perceptron (MLP):** $\phi(X) = (\phi(x_1), \ldots, \phi(x_N))$
- **Full transformer:** $T(X) = (\phi_D \circ L_D \circ \ldots \circ L_1 \circ \phi_0)(X)$



Source: https://lilianweng.github.io/posts/2018-06-24-attention/

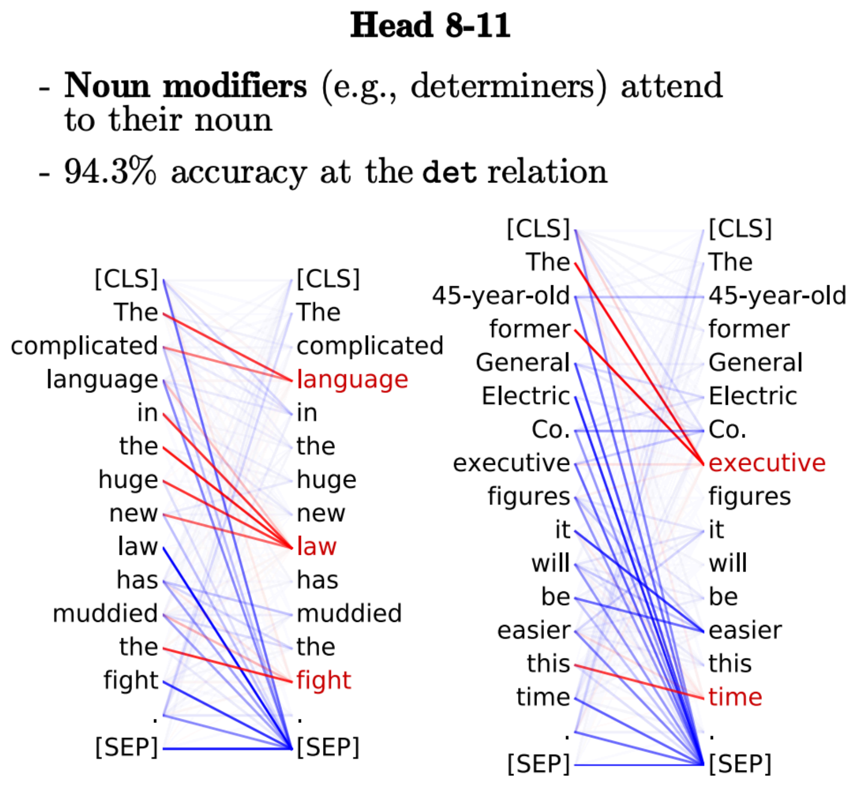
Transformer architecture

What is it?

- **Self-attention unit:** $f(X) = \text{softmax}(XQK^T X^T)XV$ for input $X \in \mathbb{R}^{N \times d}$, model parameters $Q, K, V \in \mathbb{R}^{d \times m}$.
- **Multi-headed attention:**
$$L(X) = X + \sum_{h=1}^{H} f_h(X)$$
- **Element-wise multi-layer perceptron (MLP):** $\phi(X) = (\phi(x_1), \ldots, \phi(x_N))$
- **Full transformer:** $T(X) = (\phi_D \circ L_D \circ \ldots \circ L_1 \circ \phi_0)(X)$

Key features

- **Computationally efficient training:** parallelizable training, unlike RNNs
- **Attuned to pairwise linguistic structure**: self-attention encodes syntactic and semantic linkages between words*



- **Backbone of modern NLP and vision models.**

*I am not a linguist!

Transformer architecture

What is it?

- **Self-attention unit:**
  $f(X) = \mathrm{softmax}(XQK^TX^T)XV$ for input $X \in \mathbb{R}^{N \times d}$, model parameters $Q, K, V \in \mathbb{R}^{d \times m}$.
- **Multi-headed attention:**
  $L(X) = X + \sum_{h=1}^{H} f_h(X)$
- **Element-wise multi-layer perceptron (MLP):**
  $\phi(X) = (\phi(x_1), \ldots, \phi(x_N))$
- **Full transformer:**
  $T(X) = (\phi_D \circ L_D \circ \ldots \circ L_1 \circ \phi_0)(X)$

Our questions

Can the strengths and limitations of transformers be understood via function approximation?

1. Power of transformers over fully-connected & recurrent NNs?
2. Representational impact of model parameters m, H, D?
3. Tasks that transformers struggle with?

Transformer architecture

Our questions

Can the strengths and limitations of transformers be understood via function approximation?

1. Power of transformers over fully-connected & recurrent NNs for sequential tasks?
2. Representational impact of model parameters m, H, D?
3. Tasks that transformers struggle with?

Our contributions

Provide two “natural” tasks that exhibit key separations between transformers and other models:

- **Sparse averaging** is efficient for transformers, inefficient for RNNs, FNNs.
- **Pair finding** is easy for transformers, **triple finding** is not.

What is already known theoretically?

- **Universality:** Turing completeness of sufficiently large transformers [PMB19, YBR+20, WCM22]
- **Formal language recognition:**
    - Recognize counter languages [BAG20], bounded-depth Dyck languages [YPPN21], bounded-size automata [LAG+22]
    - **Fixed-size** transformer cannot represent infinite-depth Dyck languages [HAF22]
- **Learnability:** Generalization bounds via covering numbers [EGKZ22, BPKP22]
- **Graph neural networks:** Message-passing analogue to attention, equivalence to CONGEST distributed communication model [Lou19]

Transformer architecture

Our questions

Can the strengths and limitations of transformers be understood via function approximation?

1. Power of transformers over fully-connected & recurrent NNs for sequential tasks?
2. Representational impact of model parameters m, H, D?
3. Tasks that transformers struggle with?

Modeling decisions

Model	Context length (*N*)	#layers (*D*)	#heads (*H*)	*#param self-attn (m)*	*#param MLP (k)*
GPT-3	2048	96	96	128	12288
GPT-4	32k	🙃	🙃	🙃	🙃

- Context length $N \gg$ #params in self-attention unit (depth D, heads H, and embedding dim m)

  $\Longrightarrow$ **restricted pairwise computation between elements, model size independent of N**

- #params in MLP $k \gg$ #params in self-attention

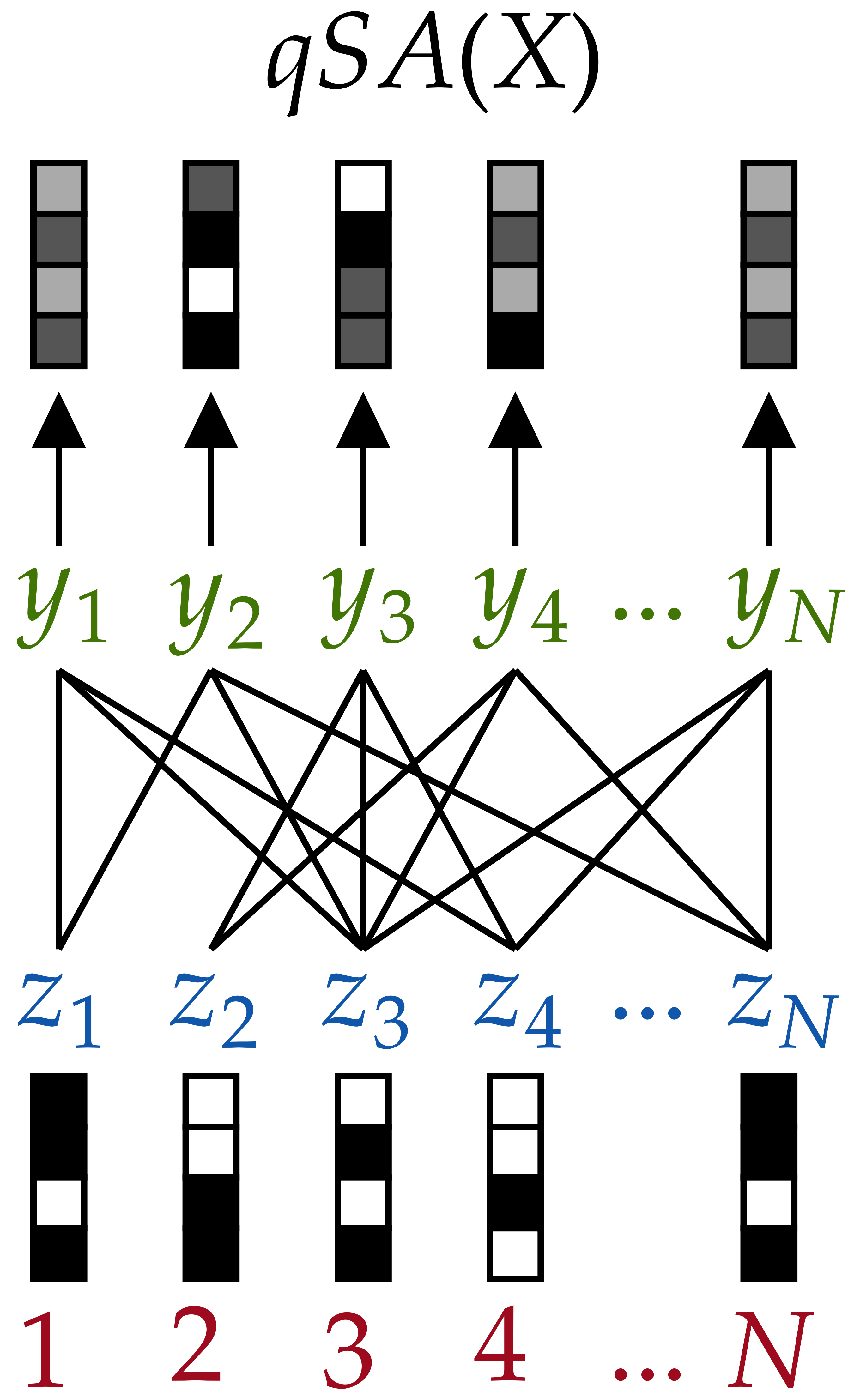
  $\Longrightarrow$ **unlimited element-wise computational power**

Part 1: Sparse averaging

The task

Input: $X = ((y_1, z_1), \ldots, (y_N, z_N))$ for $y_i \in \binom{[N]}{q}$ and $z_i \in \mathbb{R}^d$.

$$qSA(X)_i = \frac{1}{q} \sum_{j \in y_i} z_i$$



Part 1: Sparse averaging

The task

Input: $X = ((y_1, z_1), \ldots, (y_N, z_N))$ for $y_i \in \binom{[N]}{q}$ and $z_i \in \mathbb{R}^d$.

$$qSA(X)_i = \frac{1}{q} \sum_{j \in y_i} z_i$$

Results

1. Inefficient representation with FNNs or RNNs.
   - Any FNN requires width $\Omega(Nd)$.
   - Any RNN requires $\Omega(N)$-bit hidden state.
2. **There exists a single unit of self attention that approximates $qSA(X)$ iff embedding dimension $m \gtrsim q$.**

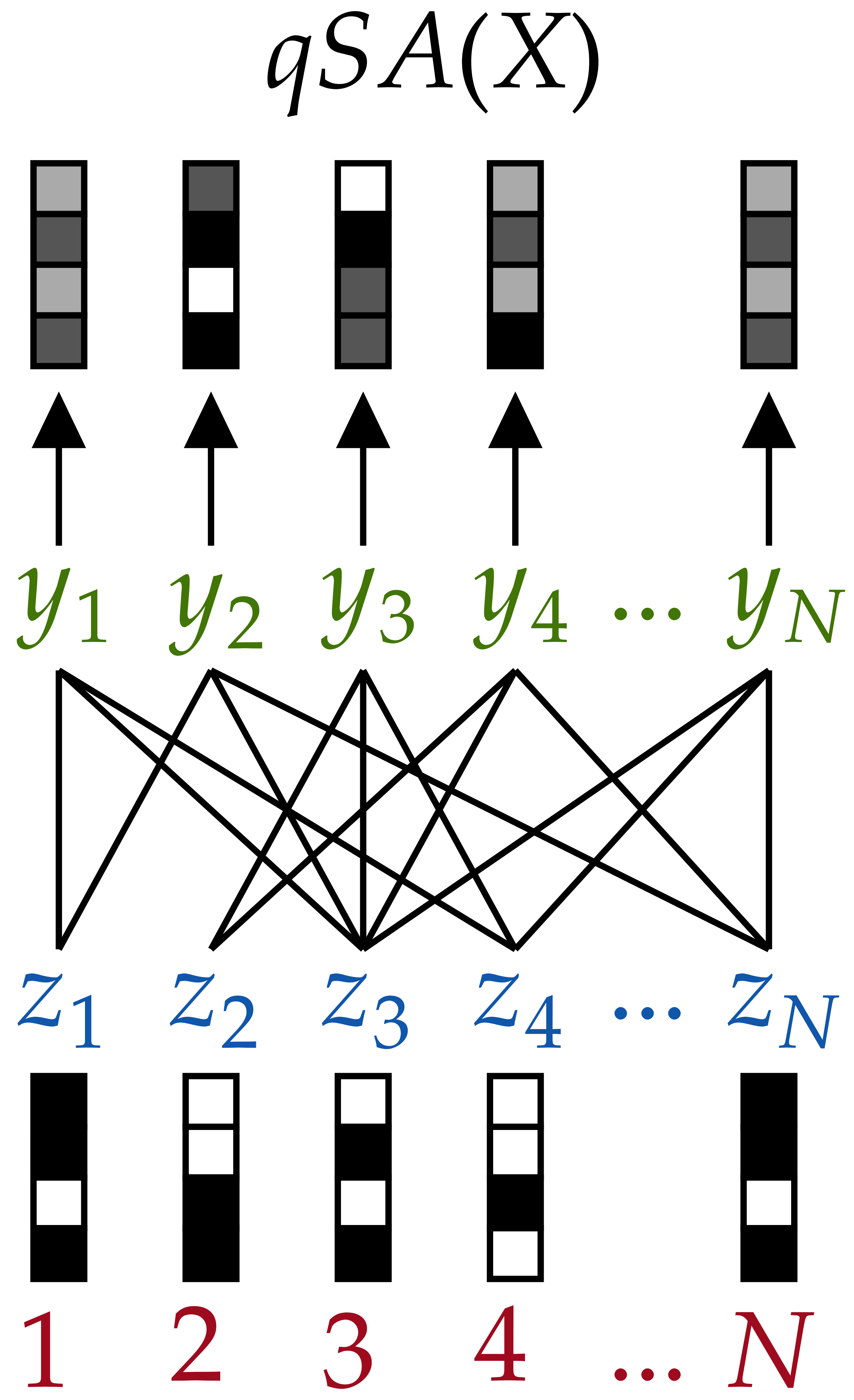
Part 1: Sparse averaging

The positive result

Theorem: For all q, there exists a self-attention unit f with embedding dimension $m = O(d + q \log N)$ that approximates qSA at all X with $\log(N)$-bit precision* arithmetic.

Think: $\log N, d \ll q \ll N$

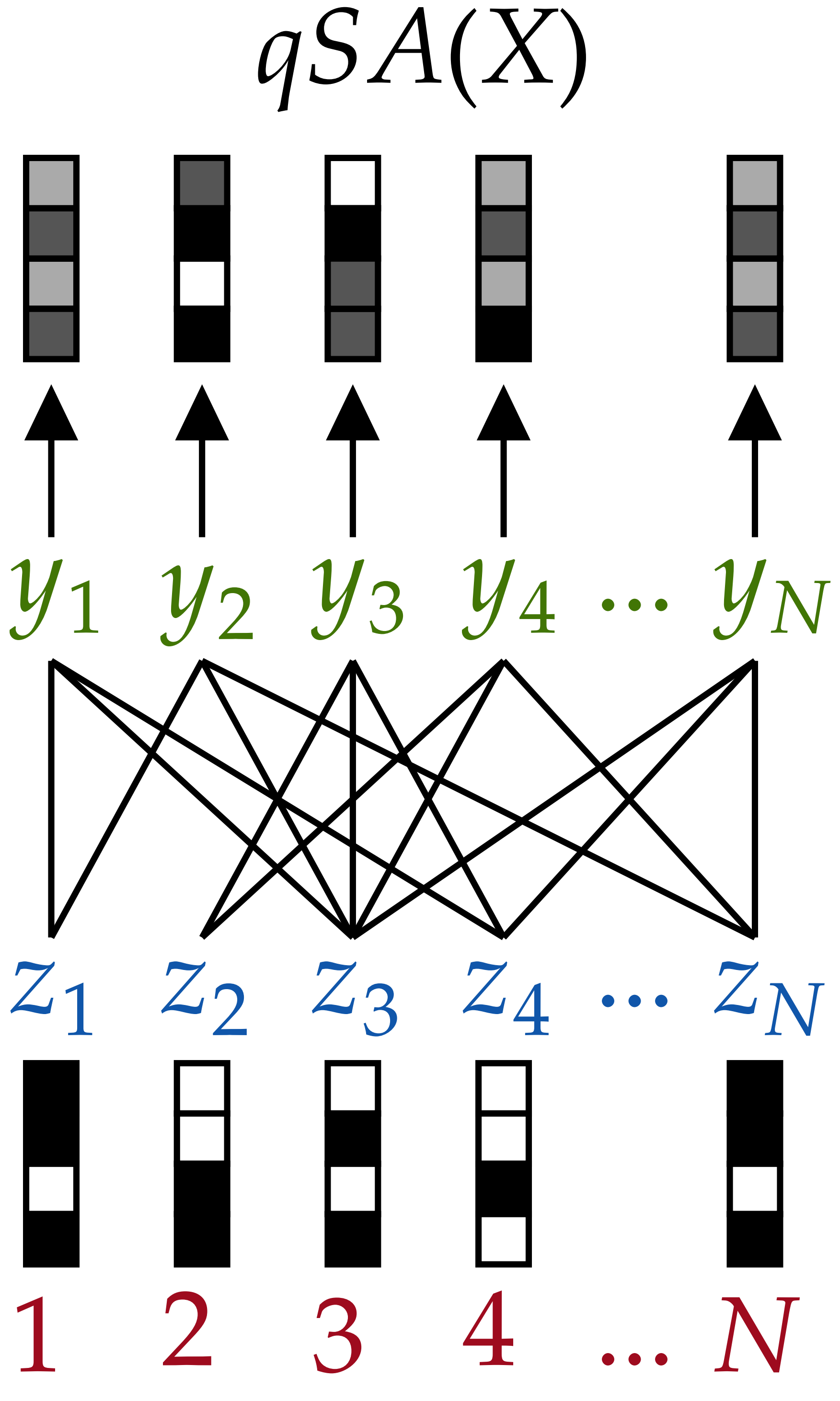
*The $\log N$ factor can be eliminated by using infinite-bit precision.

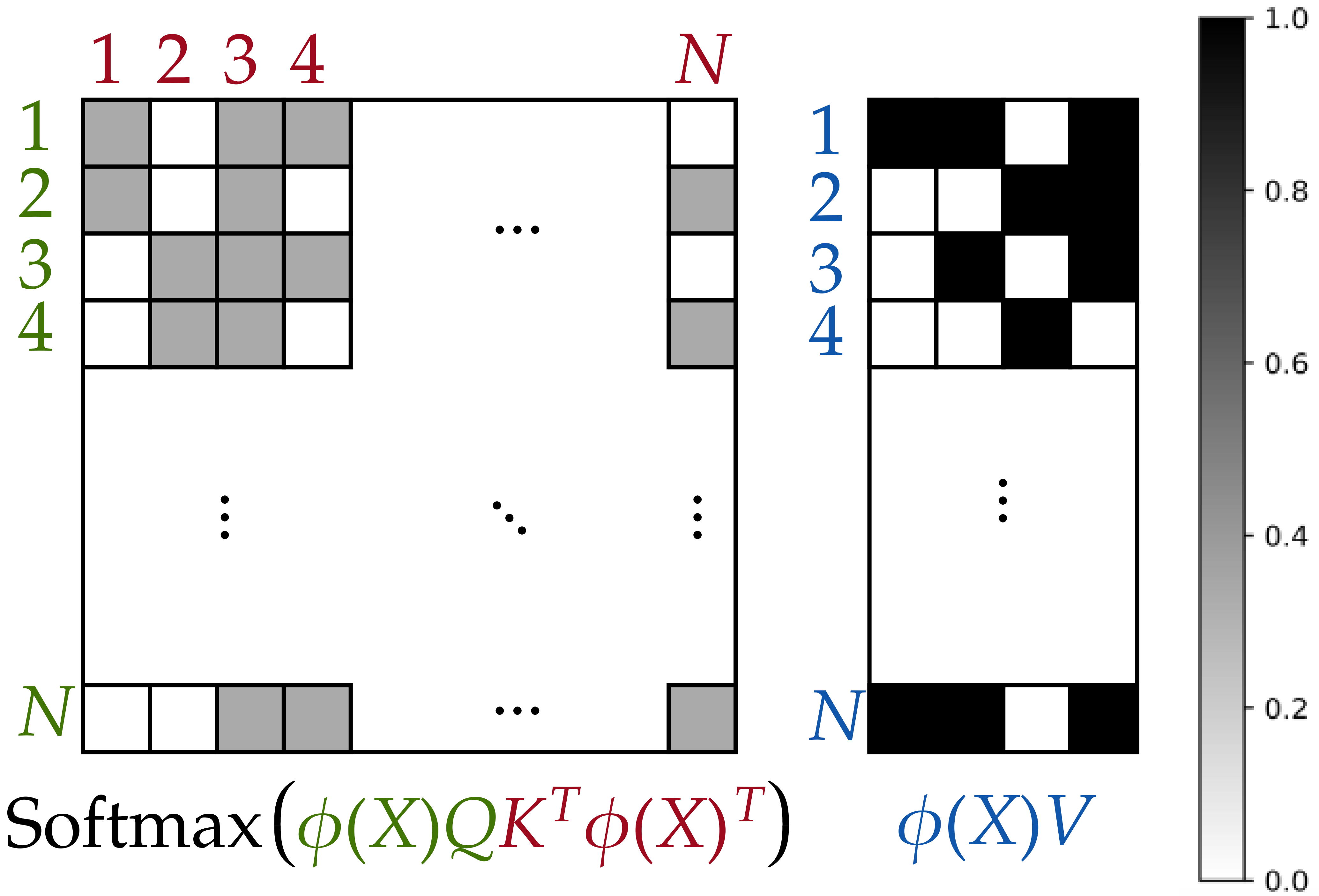


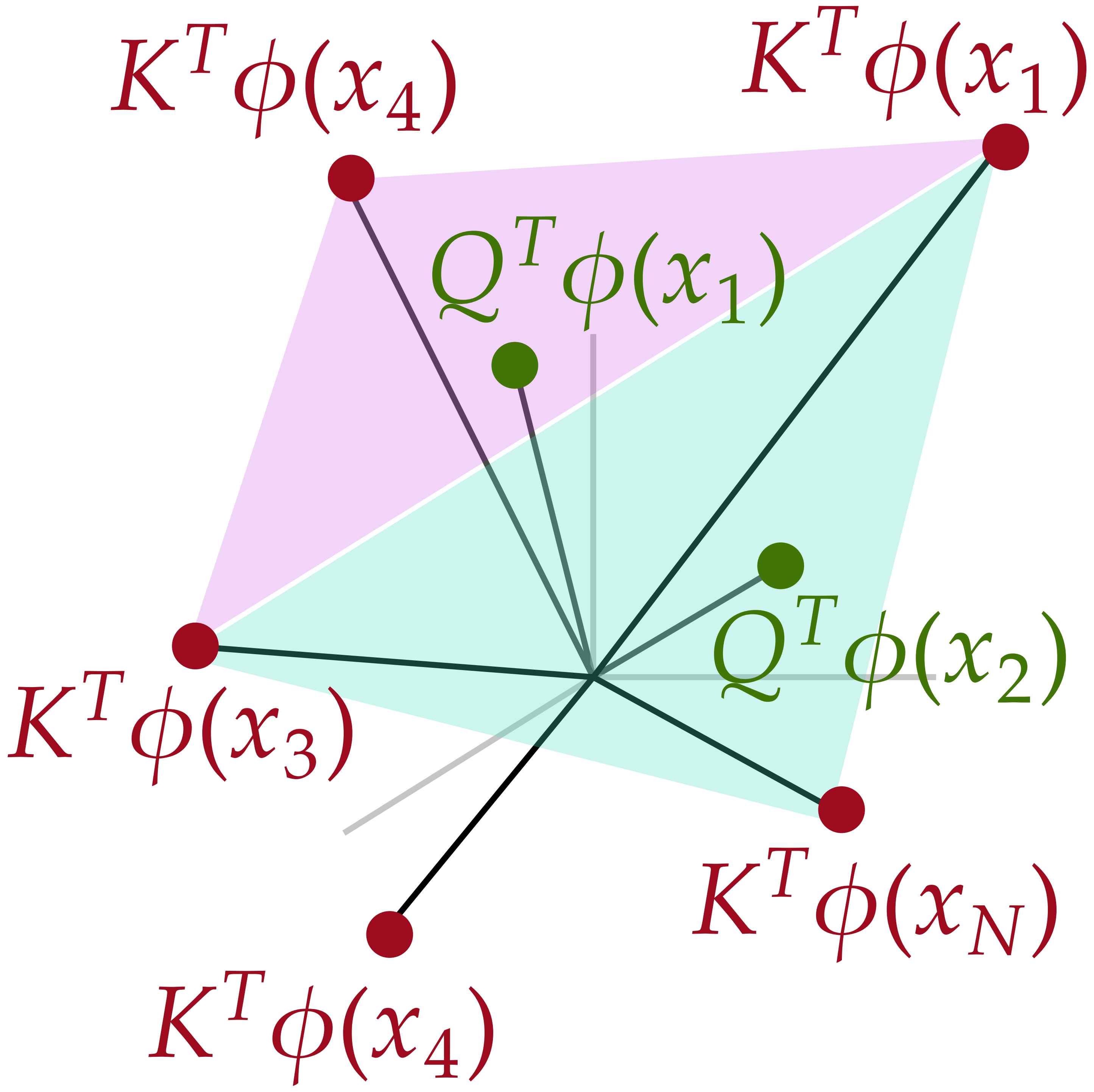
Part 1: Sparse averaging

The positive result: proof by picture

Theorem: For all q, there exists a self-attention unit f with embedding dimension $m = O(d + q\log N)$ that approximates qSA at all X with $\log(N)$ -bit precision* arithmetic.



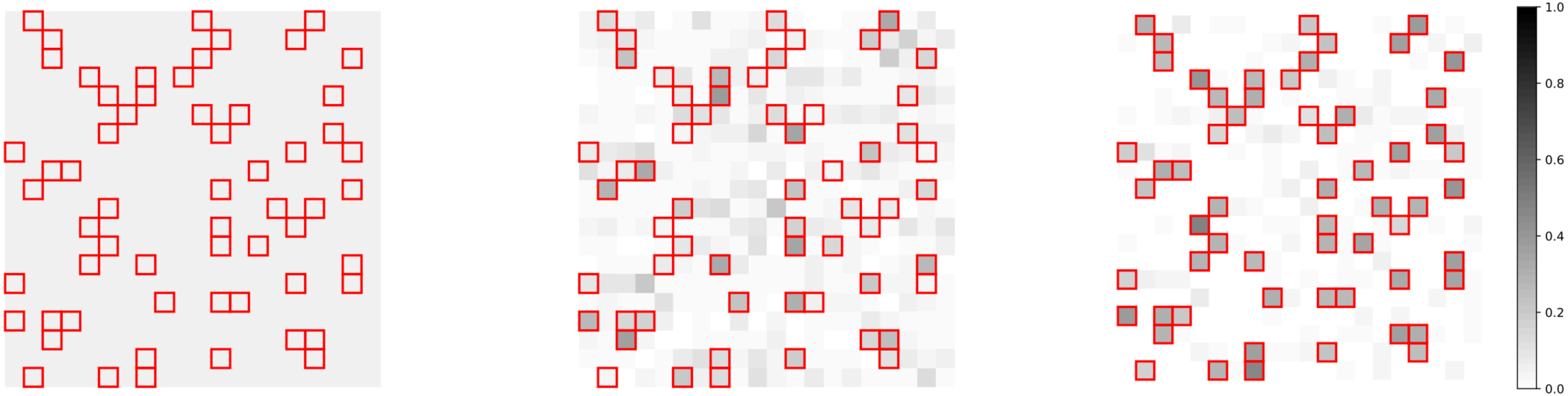




Part 1: Sparse averaging

The positive result: proof by picture

Theorem: For all q, there exists a self-attention unit f with embedding dimension $m = O(d + q \log N)$ that approximates qSA at all X with $\log(N)$-bit precision* arithmetic.



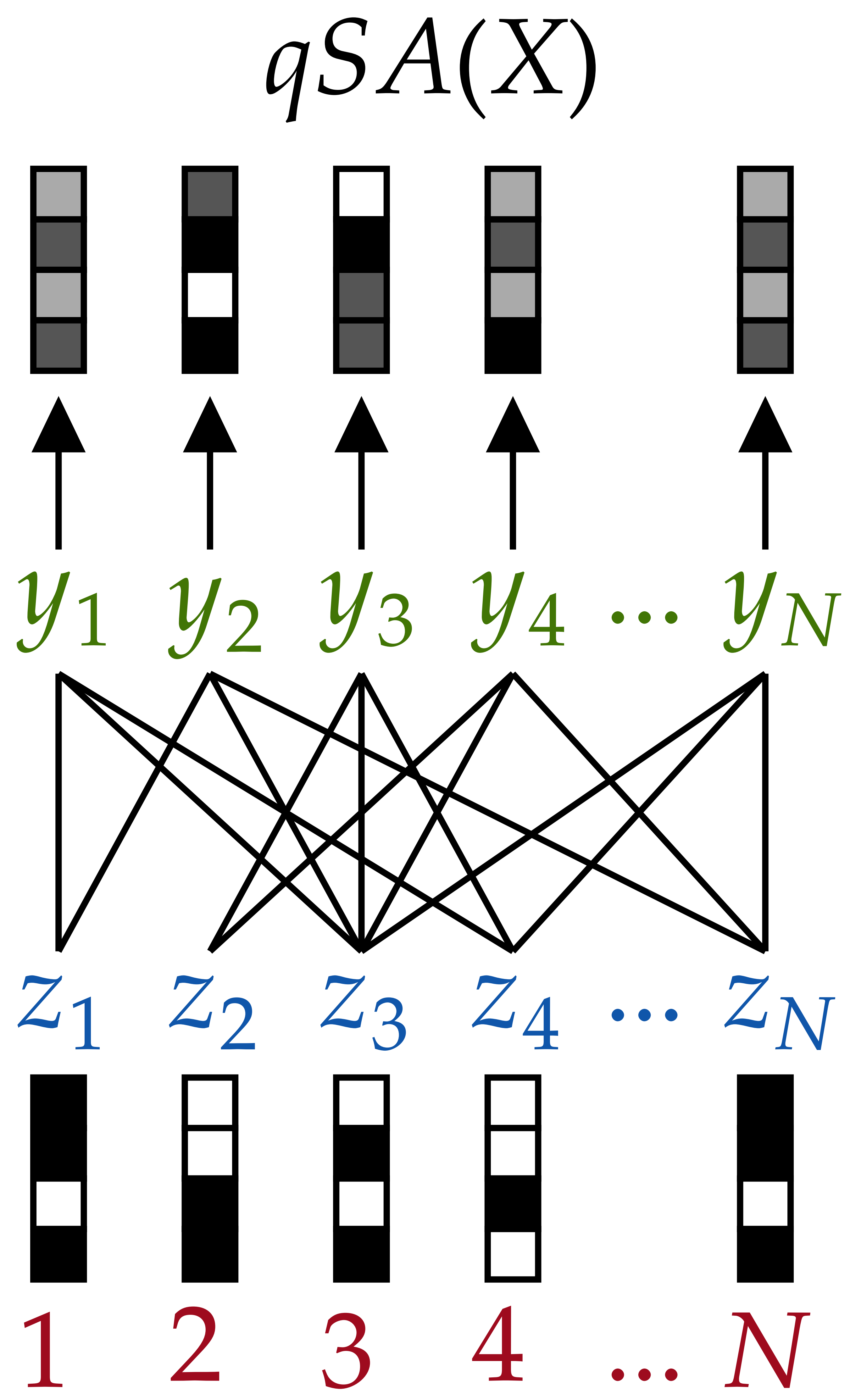
(a) $T = 0$. (b) $T = 1000$. (c) $T = 40000$.

Part 1: Sparse averaging

The negative result

Theorem: Any self-attention unit f that approximates qSA with $\log(N)$-bit precision arithmetic requires embedding dimension $m \geq q/\log N$.

Proof by communication complexity…



Part 1: Sparse averaging

An aside on communication complexity

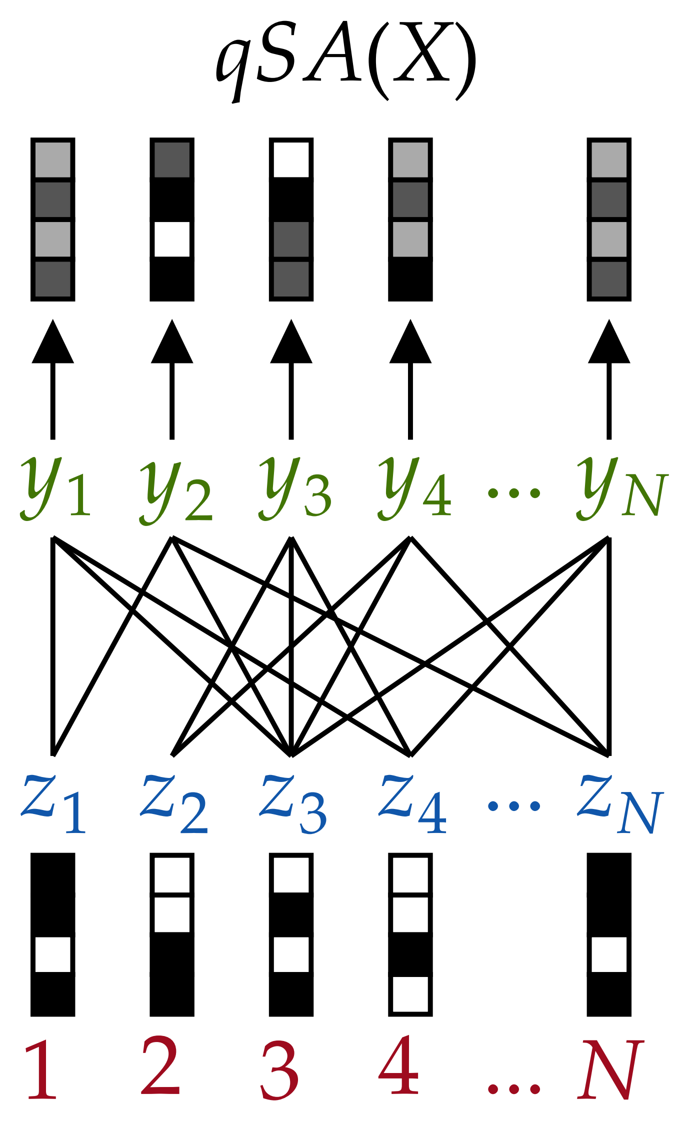
- Suppose Alice has $a \in \{0,1\}^n$ and Bob has $b \in \{0,1\}^n$ and they want to compute $\mathrm{DISJ}(a, b) = \max_i a_i b_i$.
- Unlimited computation, bounded communication:
  - Alice and Bob take turns sending single bits of information to one another.
- What is the minimum rounds of communication?
  - $\leq n$ (Alice sends all bits to Bob)
  - $\geq n$ (rank of characteristic matrix)

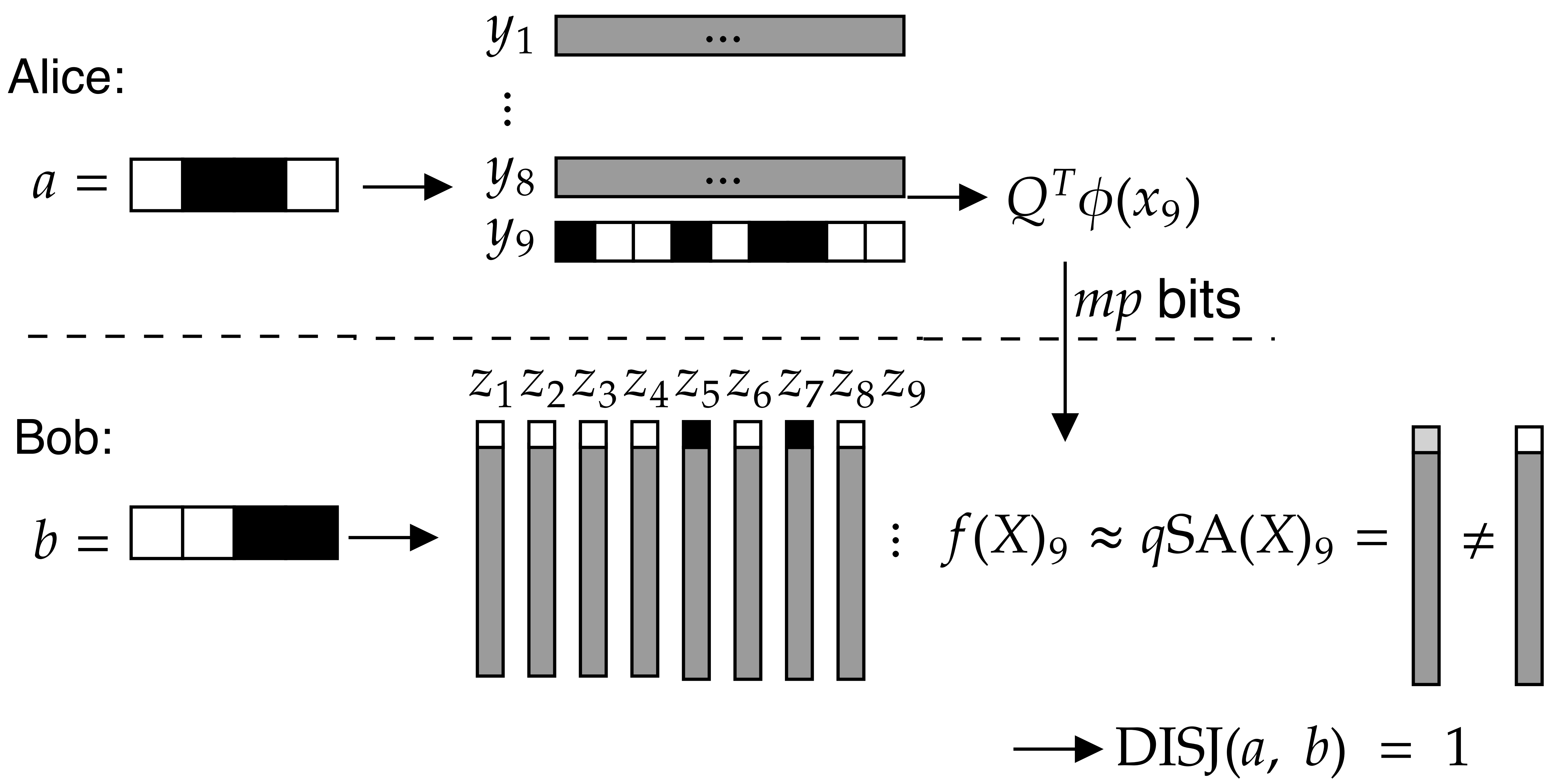
Part 1: Sparse averaging

The negative result: proof

Theorem: Any self-attention unit f that approximates qSA with $\log(N)$-bit precision arithmetic requires embedding dimension $m \geq q/\log N$.

- Create an $m \log N$-bit protocol for $\mathrm{DISJ}(a, b)$ with $n = q$, assuming the existence of f.
- Alice encodes her input in subset $y_{2q+1} = \{2i + a_i - 1 : i \in [q]\}$.
- Bob encodes his input as $z_{2i-1} = 2a_i - 1,\ z_{2i} = -1$. All other values set arbitrarily.
- Alice sends Bob her $m \log N$-bit query encoding $Q(x_{2q+1})$.
- Bob computes $f(X)$ and returns 1 iff $f(X)_{2q+1} \neq -1$.
- By CC bound, $m \log N \geq q$.





Part 1: Sparse averaging

The task

Input: $X = ((y_1, z_1), \ldots, (y_N, z_N))$ for $y_i \in \binom{[N]}{q}$ and $z_i \in \mathbb{R}^d$.

$$qSA(X)_i = \frac{1}{q} \sum_{j \in y_i} z_i$$

Results

1. Inefficient representation with FNNs or RNNs.
   - Any FNN requires width $\Omega(Nd)$.
   - Any RNN requires $\Omega(N)$-bit hidden state.
2. There exists a single unit of self attention that approximates $qSA(X)$ iff embedding dimension $m \gtrsim q$.

Part 2: Pair and triple finding

The tasks

Input: $X = (x_1, \ldots, z_N) \in [M]^N$.

$\text{Match2}(X)_i = 1\{\exists j : x_i + x_j = 0\}$

$\text{Match3}(X)_i = 1\{\exists j_1, j_2 : x_i + x_{j_1} + x_{j_2} = 0\}$

Results

1. Efficient representation of Match2 with self-attention unit.
2. No efficient representation of Match3 with multi-headed self-attention.

Part 2: Pair and triple finding

Result #1

$$\text{Match2}(X)_i = 1\{\exists j : x_i + x_j = 0\}$$

Theorem: There exists self-attention unit f with input MLPs and embedding dimension $m = O(1)$ such that $f(X) = \text{Match2}(X)$.

Proof Idea

- Choose embeddings:
  $Q(x_i) = c(\cos(2\pi x_i/M), \sin(2\pi x_i/M))$
  $K(x_i) = (\cos(2\pi x_i/M), -\sin(2\pi x_i/M))$
- Then:
  $(Q(X)K(X)^T)_{i,j} = c\cos(2\pi(x_i + x_j)/M)$
- For sufficiently large c:
  $\text{softmax}(Q(X)K(X)^T)_{i,j} \approx 0$ iff $x_i + x_j \neq 0$.
- Caveat: need blank “<STOP>" token at the end.

Part 2: Pair and triple finding

Result #2

$$\text{Match3}(X)_i = 1\{\,\exists j_1, j_2 : x_i + x_{j_1} + x_{j_2} = 0\}$$

Theorem: Any H-headed self-attention with input and output MLPs and embedding dimension m and $O(\log N)$-bit precision arithmetic approximating Match3 has $mH = \Omega(N/\log N)$.

Proof Idea

- Similar communication complexity proof.
- Embed $\text{DISJ}(a, b)$ for $n = (N-1)/2$, where Alice knows $x_1, x_2 \ldots, x_{(N-1)/2}$ and Bob knows $x_1, x_{(N+1)/2}, \ldots, x_N$.
- $\text{DISJ}(a, b) = 1$ iff triple $x_1 + x_i + x_{i+(N-1)/2} = 0$.
- Alice sends Bob $O(mH \log N)$ bits from partially computed attention units.

Part 2: Pair and triple finding

The tasks

Input: $X = (x_1, \ldots, z_N) \in [M]^N$.

$\mathrm{Match2}(X)_i = 1\{\, \exists j : x_i + x_j = 0\}$

$\mathrm{Match3}(X)_i = 1\{\, \exists j_1, j_2 : x_i + x_{j_1} + x_{j_2} = 0\}$

Results

1. Efficient representation of $\mathrm{Match2}$ with self-attention unit.
2. No efficient representation of $\mathrm{Match3}$ with multi-headed self-attention.
3. Efficient representation of $\mathrm{Match3}$ under “third-order tensor attention” generalization.
4. Efficient representation of “assisted” $\mathrm{Match3}$ with standard transformer.

Part 2: Pair and triple finding

The tasks

Input: $X = (x_1, \ldots, z_N) \in [M]^N$.

$\text{Match2}(X)_i = 1\{\exists j : x_i + x_j = 0\}$

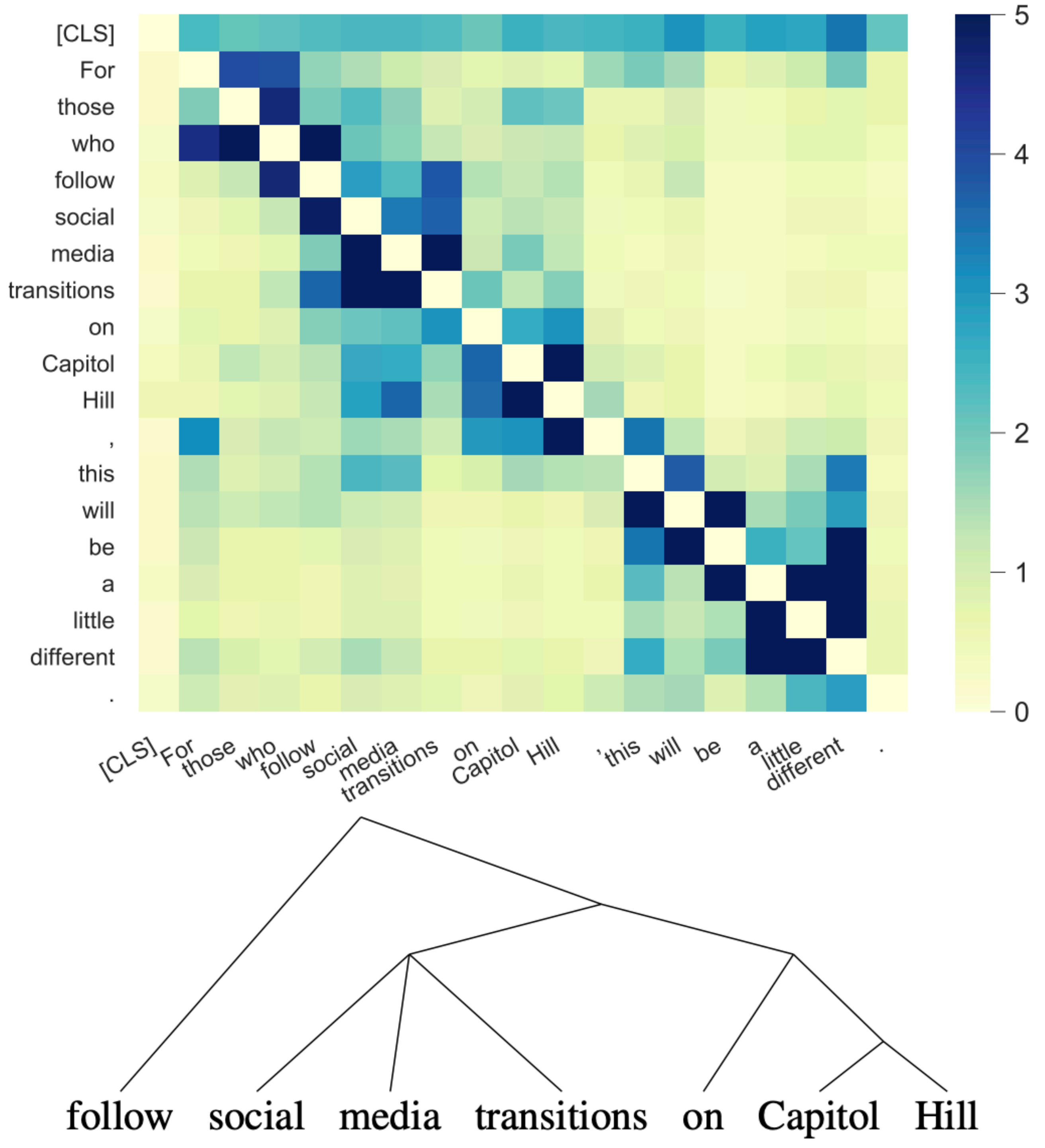
$\text{Match3}(X)_i = 1\{\exists j_1, j_2 : x_i + x_{j_1} + x_{j_2} = 0\}$

[Future] Results

Conjecture: Any D**-depth** H-headed transformer with embedding dimension m and $O(\log N)$-bit precision arithmetic approximating Match3 has $mHD = \Omega(N/\log N)$.

Future work and open questions

- Can more advanced communication complexity and distributed computing techniques be used to resolve the conjecture?
- How apt is the "sparse pairwise connectedness" framework for understanding language?
- Are there practical "intrinsically three-wise" learning tasks where modern transformers fail?



Thank you