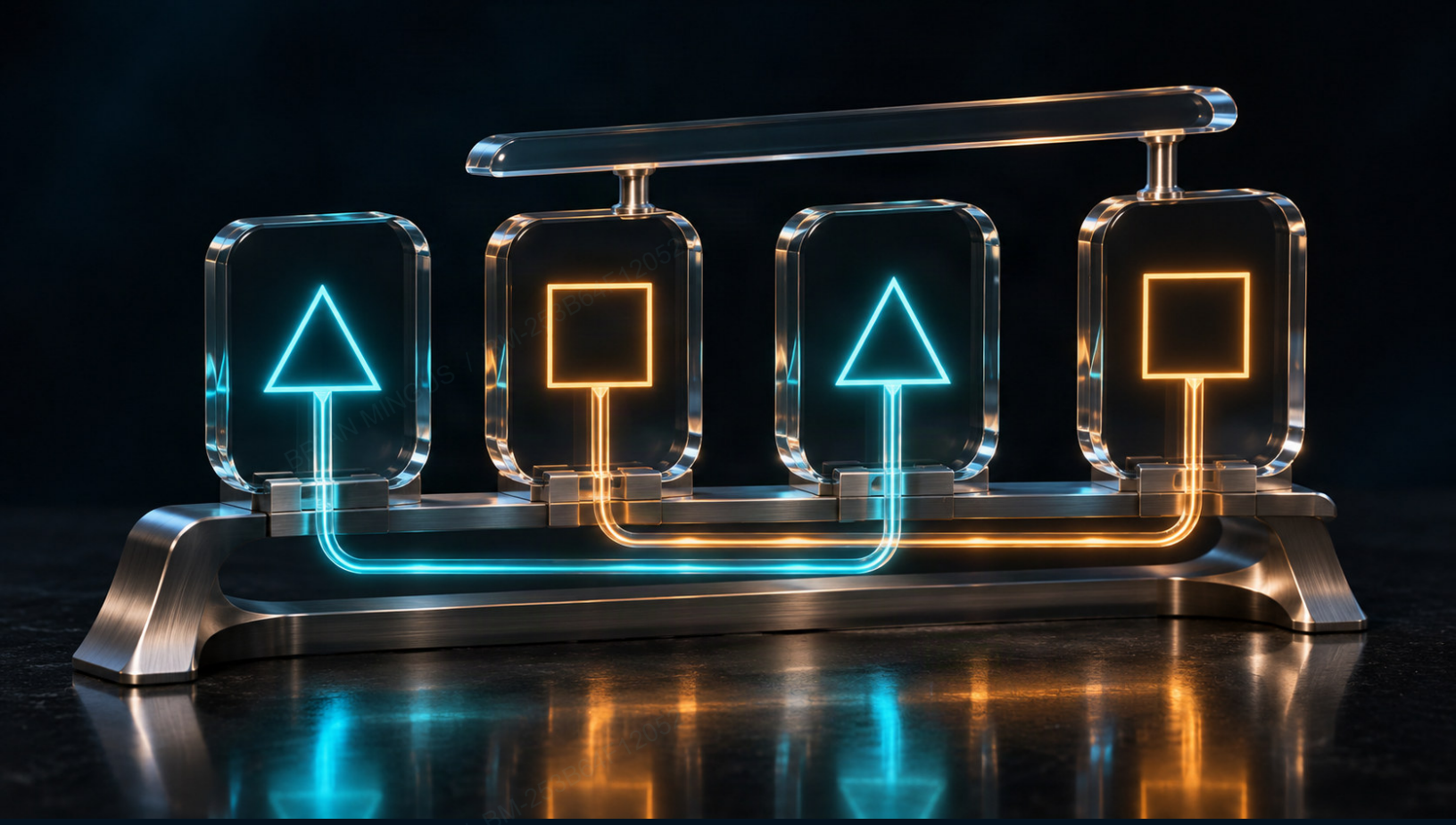

INFORMATION THEORY / TRANSFORMER CIRCUITS

From Shannon's Bigrams to Induction Heads

Brian Mingus

Cognatory, LLC

November 20, 2026

How a causal attention circuit recovers the empirical distribution of a word's observed successors.

brianmingus.com

Research portfolio edition · The complete manuscript follows.

AI-generated conceptual cover illustration; scientific figures are unchanged.

BRIAN MINGUS RESEARCH COLLECTION / BM-253B64F12052

Cognatory Technical White Papers

Cognatory, LLC *November 20, 2026* *Portfolio*

From Shannon's Bigrams to Induction Heads

By BRIAN MINGUS*

INTRODUCTION

THE words "the head and in frontal attack" open a passage Claude Shannon composed with a book and a rule for choosing its next word. Having written a word, he found another occurrence in the book and copied the word following it. Repeating the procedure preserved something of the sound of English even as the argument wandered. Shannon called this his *second-order word approximation*. It appears in his classified memorandum of 1945 and in his published paper of 1948.[1]

Suppose that the book is set aside and the writer consults the passage already written. At a second occurrence of a word A, the word B that followed its first occurrence is now a candidate for the next word. This is the elementary induction pattern $A\,B\,\ldots\,A \longrightarrow B$. An association acquired within a passage is available for use later in the same passage. The vocabulary and the rule for consulting it can both be fixed in advance.

There is a small difficulty in realizing this rule in a causal transformer. The position holding an earlier A could not, when it was read, contain information about the B still to come. The difficulty disappears if the pair is stored at B. One head carries the identity of the preceding word forward by one place. A second head can then find A and read B at the same position. This arrangement was identified in trained two-layer attention models by Elhage and colleagues.[2]

For a word that has appeared before, perfectly selective attention gives the empirical distribution of its successors. At finite selectivity, positions with other keys also receive weight. The size of this effect depends on how many earlier positions compete with each matching occurrence.

* Cognatory, LLC. brian@cognatory.com.

[1] C. E. Shannon, "A Mathematical Theory of Cryptography," Bell Telephone Laboratories, Sept. 1, 1945, §3, example 6. "A Mathematical Theory of Communication," *Bell System Technical Journal*, 27, July 1948, pp. 388–389. Markov's 1913 study of *Eugene Onegin* counted vowel and consonant transitions. See the translation in *Science in Context*, 19, 2006, pp. 591–600.

[2] N. Elhage et al., "A Mathematical Framework for Transformer Circuits," *Transformer Circuits Thread*, 2021, section on induction heads.

BRIAN MINGUS RESEARCH COLLECTION / BM-253B64F12052

1. The Word Source

Let $D = (d_1, \dots, d_L)$ be a text over a finite vocabulary. For words a and b, denote by $C_D(a,b)$ the number of positions $i < L$ at which $d_i = a$ and $d_{i+1} = b$. Write $C_D(a) = \sum_b C_D(a,b)$. Choosing uniformly among occurrences of a with a recorded successor gives

$$P_D(b \mid a) = \frac{C_D(a,b)}{C_D(a)}, \qquad C_D(a) > 0. \tag{1}$$

Shannon approximated this sampling law by opening a book at random and scanning to a match.

For a fixed D, drawing a word from (1) and using it as the next query yields a first-order Markov generator. The change to the prediction rule is the substitution $D = (x_1, \dots, x_t)$, $a = x_t$. Only completed pairs enter the counts. The final x_t can already be the successor of x_{t-1}, although its own successor has yet to be observed.

The distinction already appears in the passages ABA and ACA, where A, B, C are distinct. The first passage gives all its weight to B and the second to C. A fixed transition row for A cannot give both answers. Each query consists of one word. The evidence for its continuation can lie anywhere in the visible prefix.

2. Realization by Attention

Write $e(b)$ for the vector with a one in coordinate b and zeros elsewhere. Regard each word as one token. In an ideal previous-token head at position $j \geq 2$, all attention goes to position $j - 1$. The head writes $e(x_{j-1})$ into a separate part of the representation at j. For the later induction head, take the query, key and value to be

$$q = e(x_t), \qquad k_j = e(x_{j-1}), \qquad v_j = e(x_j). \tag{2}$$

Shifting the key assigns the pair (x_{j-1}, x_j) to its second position. The query can now attend directly to the word whose identity it should copy, as shown in Fig. 1. Position 1 has no preceding word and is omitted.

Let $\beta \geq 0$ determine the sharpness of a match, absorbing the usual dot-product scale into this constant. For $t \geq 2$ the head's output is

$$\alpha_j = \frac{\exp(\beta\, q^\top k_j)}{\sum_{\ell=2}^{t} \exp(\beta\, q^\top k_\ell)}, \qquad o_t = \sum_{j=2}^{t} \alpha_j v_j. \tag{3}$$

Here $o_t(b)$ is the fraction of attention delivered to copies of b. In a trained transformer each head writes into the residual stream. The full model combines these contributions to obtain its next-token probabilities.[3]

[3] C. Olsson et al., "In-context Learning and Induction Heads," 2022, especially the distinction between prefix matching and copying.

BRIAN MINGUS RESEARCH COLLECTION / BM-253B64F12052

3. The Copy Distribution

There are $n = t - 1$ completed pairs. Suppress the subscript D, let $a = x_t$, and put $m = C(a)$. Let $U(b)$ count the occurrences of b in positions $2, \ldots, t$, so that $U(b) - C(a,b)$ counts copies of b whose preceding word differs from a.

Proposition. For one-hot keys and values, the copy weight at finite sharpness is

$$o_t(b) = \frac{e^{\beta} C(a,b) + U(b) - C(a,b)}{e^{\beta} m + n - m}. \tag{4}$$

For $m > 0$, its limit as β tends to infinity is $C(a,b)/m$.

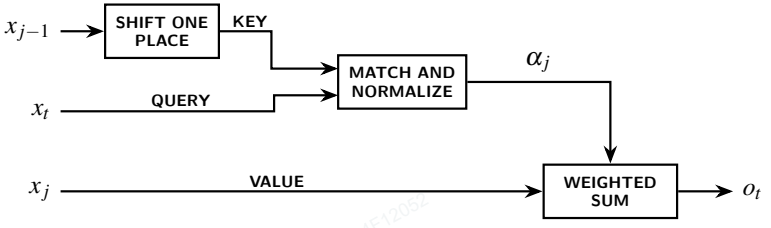

Fig. 1.—The path of one observed pair through the two heads. The weights are normalized over positions and the values summed at the current word.

Proof. The dot product in (2) and (3) is one at the m positions whose keys match a and zero elsewhere. Each matching position contributes e^{β} to the denominator of (3) and every other position contributes one. Summing those same weights over positions whose value is b gives the numerator of (4). Division by e^{β} gives the limit.

The weight assigned to all positions with a different key is

$$\delta = \frac{n - m}{m e^{\beta} + n - m}.$$

For $0 < m < n$, (4) is a mixture of the matching successors with weight $1 - \delta$ and the other successors with weight δ. Thus δ bounds the total variation distance from (1). The bound is attained when the two groups have disjoint successor vocabularies. For nine tenths of the weight or more to fall on matches, one needs $m e^{\beta} \geq 9(n - m)$.

If $m = 0$, (1) supplies no estimate and (3) gives $U(b)/n$, since all positions tie. If $m = n$, every position matches and (1) is recovered at every sharpness. For $t = 1$ there is no completed pair and the copy rule supplies no evidence.

Direct attention (3) and the count formula (4) were evaluated independently on all 3,279 strings of lengths one through seven over three words. At sharpness $0, \ln 3, 5, \infty$, the largest absolute difference in any coordinate was 5.55×10^{-16}.[4]

[4] Run `induction.py --check` for the enumeration and `induction.py` for Table I. The program uses NumPy and double-precision arithmetic.

BRIAN MINGUS RESEARCH COLLECTION / BM-253B64F12052

4. An Example

Consider the following text, with one word per position.

PAPER CRANE PAPER PLANE PAPER CRANE PAPER

The last *paper* matches the keys at positions 2, 4, and 6. Their values are *crane*, *plane*, and *crane*. The three remaining positions with keys all have value *paper*. Here $m = 3$, $n = 6$, and $\delta = 1/(1+e^{\beta})$. The weights are $\frac{2}{3}(1-\delta)$, $\frac{1}{3}(1-\delta)$, and δ, respectively.

TABLE I. Copy weights at the last word

Sharpness β	Crane	Plane	Paper
0	.3333	.1667	.5000
$\ln 3$	.5000	.2500	.2500
5	.6622	.3311	.0067
∞	.6667	.3333	.0000

Changing the first *crane* to *plane* interchanges the weights of these two words at every sharpness. The edit changes one of the three observed successors of *paper*.

A stricter test holds all word counts fixed. Compare *paper crane plane paper* with *paper plane crane paper*. The final query is the same, and each of the three words occurs once among the eligible values. Only one key matches. In the first passage it points to *crane*, and in the second to *plane*. At $\beta = \ln 3$, the weights of *crane* and *plane* are $3/5$ and $1/5$ in the first passage and $1/5$ and $3/5$ in the second. A predictor using only word frequencies must give the same answer on the two passages.

Edelman and colleagues observed a learned counterpart in transformers trained on sequences from different Markov chains. In the simple circuit studied here the prompt supplies the transitions. The one-place shift puts each successor where its predecessor can be recognized.[5]

[5] E. Edelman, N. Tsilivis, B. L. Edelman, E. Malach and S. Goel, "The Evolution of Statistical Induction Heads: In-Context Learning Markov Chains," *Advances in Neural Information Processing Systems*, 37, 2024.

BRIAN MINGUS RESEARCH COLLECTION / BM-253B64F12052