# Sequence-JOIN: A Research Proposal

Patrick Perrine

February 12, 2024

This is to formally document and properly describe a research direction from an informal discussion held in the evening of February 5, 2024 to our research group. This presents the concept of a new algorithm for the Neuroidal model to memorize *sequences* of information, known as *Sequence-JOIN* or *SJOIN* for short.

The inspiration for SJOIN came about when reading about the "Basic Mechanism" from Valiant [2017]. The Basic Mechanism formally describes the relationship between two memories $A$ and $B$, in that firing $A$ results in the subsequent firing of $B$, or $A \to B$. This is an already well-understood process in our previous work implementing JOIN and developing QJOIN, as in Perrine [2023]. Yet the more specific detail of its description in Valiant [2017] was more effective in inspiring future work. Note that the Basic Mechanism is equivalent to that of *Willshaw networks*, which prompts literature review into that area of previous work, beginning with Willshaw [1971].

To clarify, this is a new direction altogether in answering a different research question from previous efforts. The primary question to answer under this guidence would be: *How many sequences, and of what length, can the Neuroidal model memorize?* This question is distinct from the original topic of: *What is the general capacity of the Neuroidal model, with respect to unsupervised JOINs?* as with Chowdhury [2023] and Perrine [2023]

## 1 Algorithmic Formulations

For reference, here is the exact formulation of the JOIN algorithm that we are most immediately familiar with:

**Algorithm 1:** One-Step JOIN Algorithm for Shared Representations

**Input:** The Neuroidal model $G$, and memories $A$ and $B$ to be joined.
**Output:** An updated $G$, along with the new memory, $C$.

```
Algorithm OneStepSharedJOIN(G = (V, E), A, B):
    for all neurons i in A ∪ B do
        f_i ← 1                              // Fire all neurons in A ∪ B
    C ← ∅
    UpdateNeuroids(G)                        // Update G to determine any threshold transitions
    for each neuron i in V do
        if q_i == 2 then
            C ← C ∪ i                        // Collect the transitioned neurons for C
    for all neurons i in V do
        f_i ← 0                              // Reset the states for all neurons and synapses
        q_i ← 1
    for all synapses {j, i} in E do
        qq_ji ← 1
    return G, C
```

Now, here is the proposed new algorithm, SJOIN:

---

**Algorithm 2:** Sequence-JOIN for $l$-sized Sequences

**Input:** The Neuroidal model $G$, initial memory $A$, and the sequence length $l$.
**Output:** An updated $G$, along with the vector of total, chained memories $L$.
**Algorithm** `SequenceJOIN(`$G = (V, E), A, l$`)`**:**

```
L ← (A)
for m = 1, 2, ... l do
    for all neurons i in L_m do
        f_i ← 1                          // Fire the current memory in vector L
    B_{m+1} ← ∅
    UpdateNeuroids(G)
    for each neuron i in V do
        if q_i == 2 then
            B_{m+1} ← B_{m+1} ∪ i
    for all neurons i in V do
        f_i ← 0
        q_i ← 1
    for all synapses {j, i} in E do
        qq_{ji} ← 1
    L ← (L, B_{m+1})                     // Append the new memory to L
return G, L
```

---

We can analogize a call to SJOIN to represent the start of an "event" that the model will be remembering each "action" observed throughout the experience for $l$ discrete time steps. This new algorithm is the simplest known method for incorporating sequence learning within the Neuroidal model. It is also not to be confused with the LINK algorithm, which is regarding an *association* of two pre-existing memories between a layer of relay neurons.

## 2 Problems in Theory and Practice

The few changes made from Algorithm 1 to Algorithm 2 are simple enough to implement, provided that we can establish enough firing connections from one arbitrary memory to result in the firing of a new, $r$-sized memory. This will likely be a difficult task, in that it does not seem that we can simply assume this to hold true in a $\mathcal{G}(n, p)$ graph with realistic values for $n, d, k$ and $r$. We can refer to this problem as the "cold-start for SJOIN" for the time being.

Another difficulty is that this procedure introduces a completely new empirical parameter, $l$. The choice for $l$ should not be arbitrary, and should benefit from theoretical exploration similar to that for $r$ in Valiant [2005]. If we can better understand how to connect one memory to fire a new memory of proper size, then the choice for $l$ should become easier to understand.

It is recommended to experiment with the public code for the Neuroidal model to see how we can solve the cold-start problem, establish some formal constraints to determine $l$, and perform a literature review within computational neuroscience to justify that SJOIN appears biologically plausible in some context.

## 3 Conclusion

Implementing SJOIN is certainly possible, and would be required to closely interpret Valiant [2017] from an empirical context and build more directly from those results. It seems that SJOIN is the most reasonable concept for incorporating sequence modeling within the Neuroidal model, and therefore should hold some plausibility for realistic memory formation across time steps. These efforts would assist in building out the "logic alphabet" for the Neuroidal model, as more deeply described in the earlier work of Valiant [1994].

While we understand how operations such as $(A \cap B) \to C$ or $(A \cup B) \to C$ hold in an empirical context, we don't deeply understand the simpler case of $A \to B$ in a general sense. By solving the cold-start problem and implementing SJOIN, we will have a better understanding for how best to apply the framework from Chowdhury [2023] to help analyze the general capacity of the Neuroidal model in a rigorous manner, and help answer a new question altogether with a similar rigor.

## References

Leslie G. Valiant. Capacity of Neural Networks for Lifelong Learning of Composable Tasks. In *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*, pages 367–378, Berkeley, 2017. IEEE.

Patrick R Perrine. Neural Tabula Rasa: Foundations for Realistic Memories and Learning. Master's thesis, California Polytechnic State University, San Luis Obispo, June 2023.

D. Willshaw. *Models of Distributive Associative Memory.* PhD thesis, University of Edinburgh, 1971.

Chandradeep Chowdhury. Foundations Of Memory Capacity In Models Of Neural Cognition. Master's thesis, California Polytechnic State University, San Luis Obispo, December 2023.

Leslie G Valiant. Memorization and Association on a Realistic Neural Model. *Neural Computation*, 17(3):527–555, 2005.

Leslie G. Valiant. *Circuits of the Mind.* New York, NY, 1994.