James E. Dobson (June 2023). "Memorization and Memory Devices in Early Machine Learning." *Interfaces: Essays and Reviews on Computing and Culture Vol. 4, Charles Babbage Institute, University of Minnesota*, 40-49.

Memorization and Memory Devices in Early Machine Learning

James E. Dobson

Abstract: *A pressing concern for contemporary large language and computer vision models is the degree to which they memorize training data. Memory and memorization were essential to the development of early machine learning, especially as these techniques were intended to assist research into brain models and perceptual systems, but their operation was contested. Early "learning machines" were developed as analog alternatives to general-purpose digital computers and required storage of learned data. As in the present, the role of memory and memorization in these systems was contested and the available analog devices and architectures implemented different theories of learning and memory.*

In the present moment, there are numerous discussions and debates about the function and even the possibility of memorization in artificial neural networks, especially in large language models (Tirumala et. al., 2022). A model that has memorized content from its training data is particularly problematic, especially when these models are used for generative tasks. Desirable outputs from generative models are those that closely resemble but do not exactly match inputs. Corporations developing and releasing these new technologies may make themselves vulnerable to plagiarism or theft of intellectual property charges when an output image matches those found in training data. Exceptional performance on natural language processing benchmarks or highly accurate responses to questions from academic and industry tests and exams could be explained by the inclusion of these objects in the training data. "Leaked" private information is also a major concern for text generative models and evidence of such information would create similar liability issues (Carlini et. al., 2021). While deep learning models do not record strings of text or patches of images within the major architectural components—their weights, specialized layers, or attention heads—information from the network can be reconstructed that can reveal sources used as training inputs. This behavior is known as memorization. Memorization is frequently understood to signify a failure of information generalization. Deep neural networks are designed to recognize patterns, latent or explicit, and generalize from the representations of these patterns found within the network—this is why they are called models. Concerns about the leaking of private information are serious but are not the only issues connected with memorization in machine learning; memorization of training data is especially a problem for the testing and evaluation of models. Neural networks are not information storage and retrieval systems; their power and performance are the result of their exposure to many samples from which they learn to generalize. There are different theories of "information retention" in neural networks and the material history of the early implementations of machine learning provides evidence for the ongoing slipperiness of the concept of memory in machine learning.

The concept of memory was used in multiple distinct ways in machine learning discourse during the late 1950s and early 1960s. The interest in developing memory systems during that historical moment was tied up in the relays between three overlapping issues: the status of machine learning systems as brain models, and related, the issue of perception and memory as mutually implicated, and finally the belief that specialized learning machines would be faster than conventional computers. The machines that gave machine learning its name were originally developed as an alternative to general-purpose digital computers. These analog machines needed to sense and store information acquired from input data. The various memory mechanisms proposed during this era functioned like semi-permanent non-volatile storage for these learning machines. They were also the weights used to learn

the criteria for classification of input data. They thus played something of a double role in these systems. If the weights were the "programming" for these self-organized systems, then they function as a record of that programming. Serving as both data and instructions, these weights enable what we now call inference on the learned model, which is to say the classification of previously unseen inputs. Memory was not only the persistence of information within the model; it was also used to refer to the nature of the representations stored as information within the weights. Like the contemporary concern with memorization, an exact memory of inputs would mean that the model would likely fail to generalize, which is to say that it was not learning.

In Frank Rosenblatt's April 1957 funding proposal for the research project known as "Project PARA" (Perceiving and Recognizing Automaton) that would eventually result in the creation of the Mark I mechanical perceptron, Rosenblatt described his recently articulated perceptron rule as not just a method for determining decision boundaries between linearly separable data but also as a way of conceptualizing memory: "The system will employ a new theory of memory storage (the theory of *statistical separability*), which permits the recognition of complex patterns with an efficiency far greater than that attainable by existing computers" (Rosenblatt, 1957). As a brain model—this was the motivating research paradigm that Rosenblatt would make clear throughout his unfortunately short life—research into machine learning and the perceptron was concerned with using these simulated neural networks to understand more about perception and brain function. While visual perception dominated early research, this area could not be unlinked from a concern with understanding how visual inputs were stored and how memories of previously perceived patterns were compared with new stimuli.

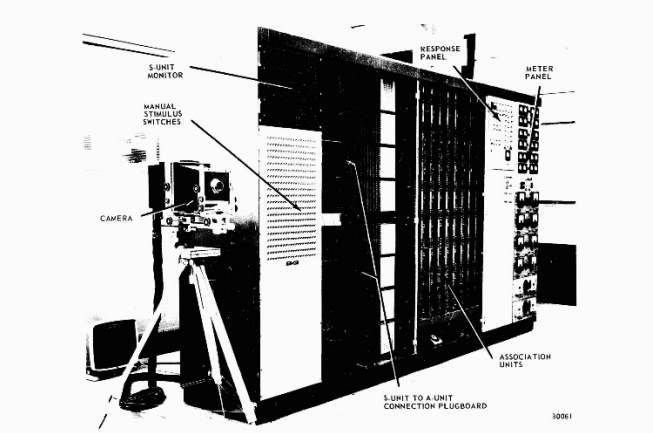

Figure 1: The Mark I Perceptron (Hay, et. al., 1960).

The "Project PARA" proposal outlines Rosenblatt's architecture. The system would be composed of three layers: the sensory or "S-System," an association or "A-System," and finally the response or "R-System." This architecture was imagined as a mechanical device and Rosenblatt anticipated this material manifestation of his design in all three layers. The "S-System," he wrote, should be imagined as "set of points in a TV raster, or as a set of photocells" and the "R-System" as "type-bars or signal lights" that might communicate output by "printing or displaying an output signal." The "A-System" would be the heart, or rather brain, of the perceptron by passing input from the sensors to the response unit by operating on the inputs in combination with pre-determined threshold value. The output from the multiple A-units, Rosenblatt explained, "will vary with its history, and acts as a counter, or register for

the memory-function of the system" (Rosenblatt, 1957). References to the material origins of machine learning are scattered throughout the terminology of this field. The weights that are learned from samples of training data are called weights because these were weighted connections between mechanical devices. The A-System provided the Perceptron's "memory function," but what it was "remembering" within these weights would be the subject of some debate.

There were a number of other early analog "learning machines" that confronted the same problems encountered by Rosenblatt. After being exposed to the Perceptron while working as a consultant in the U.S., Augusto Gamba, a physicist at the University of Genoa in Italy created his own device known as the PAPA (derived from the Italian rendering of Automatic Programmer and Analyzer of Probabilities). Like Rosenblatt's Perceptron, the PAPA combined memory and the statistical method for determining decision-making criteria:

> A set of photocells (A-units) receive the image of the pattern to be shown as filtered by a random mask on top of each photocell. According to whether the total amount of light is greater or smaller than the amount of light falling on a reference cell with an attenuator, the photocell will fire a "yes" or "no" answer into the "brain" part of the PAPA. The latter is simply a memory storing the "yes" and "no" frequencies of excitation of each A-unit for each class of patterns shown, together with a computing part that "multiplies" or "adds logarithms" in order to evaluate the probability that an unknown pattern belongs to a given class (Borsellino and Gamba, 1961).

Gamba's PAPA borrows the name "A-unit" from Rosenblatt's idiosyncratic nomenclature (one of the reasons the PAPA has become known as a "Gamba perceptron") for the Perceptron's second layer, its hidden layer, although in Gamba's architecture, the device's "memory" is not found in the association layer but in the final "brain" unit.

The relation between the machine's accumulated weights to the input data was an open problem and several different theories were used to explain and interpret the meaning of these values. For some historians of machine learning, the simplified mathematical model of a neuron proposed by Warren S. McCulloch and Walter Pitts has been assumed to be the major inspiration and basis for many working on the first neural networks (McCulloch and Pitts, 1943). While these McCulloch-Pitts neurons (as they are called) were incredibly influential, it was another theorical account that yoked together a model of perception and memory that would influence the architecture of the most important early neural networks. This was the decidedly non-mathematical work of Donald O. Hebb, a Canadian psychologist. Hebb's *The Organization of Behavior*, proposes a theory that seeks to reconcile what otherwise appeared as two distinct accounts of memory by answering the question of "How are we to provide for perceptual generalization *and* the stability of memory, in terms of what the neuron does and what happens at the synapse?" (Hebb, 1949). Perceptual generalization is the idea that people can learn to generalize from just a few examples of a wide range of objects. As Hebb puts it, "Man sees a square as a square, whatever its size, and in almost any setting" (Hebb, 1949). The stability of memory was rooted in evidence of a persistent connection or association between particular stimuli and a set of neurons. Hebb theorized a solution to this impasse with the idea of locating (in terms of neurons) independent patterns of excitation. This idea was of obvious utility to machine learning researchers wanting to develop techniques to recognize objects like letters no matter where they appeared, for example, shifted to the left or the right, when projected on a two-dimensional set of sensors called the "retina."

In an article appearing in 1958, Rosenblatt examined one theory of perception and memory that suggested that "if one understood the code or 'wiring diagram' of the nervous system, one should, in principle, be able to discover exactly what an organism remembers by reconstructing the original sensory patterns from the 'memory traces' which they have left, much as we might develop a photographic negative, or translate the pattern of electrical charges in the 'memory' of a digital computer" (Rosenblatt, 1958). Instead of memorizing inputs, Rosenblatt explained, the Perceptron implemented Hebb's theory of learning and separated learned patterns from their exact inputs. "The important feature of this approach," Rosenblatt wrote, "is that there is never any simple mapping of the stimulus into memory, according to some code which would permit its later reconstruction" (Rosenblatt, 1958). In these relatively simple machines and simulated networks, the association units might record the history of inputs as a collective representation, but they could not reproduce individual memorized inputs. For Rosenblatt, this was a sign of the success of the Perceptron; it demonstrated the practicality of Hebb's theory by implementing a memory system in the form of weights that could be used for distinguishing between classes of data without memorizing distinct inputs used to train the network. This was also Rosenblatt's grounds for differentiating the Perceptron from mere pattern matching: techniques developed contemporaneously with the Perceptron implemented databases of templates and accomplished pattern matching by memorizing and matching input samples to entries in a database (Dobson 2023).

Research on analog memory units connected two of the major sites in the development of machine learning: Rosenblatt's lab at Cornell University in Ithaca, New York and Stanford Research Institute at Stanford University in California (Stanford University would shortly divest itself of the laboratory, which would then become known as SRI International). While Rosenblatt's Mark I Perceptron is the best known of the early machines of machine learning, SRI had developed its own series of devices, the MINOS and later the MINOS II. While SRI's first projects implemented the Perceptron, researchers would later develop an alternative learning rule. SRI's MINOS project was a platform for evaluating different sensing and preprocessing techniques. George Nagy, a Hungarian-born computer scientist, worked with Rosenblatt at Cornell while a graduate student in electrical engineering; memory devices for neural networks became the subject of his dissertation and related research. Nagy worked with others in Rosenblatt's Cognitive Systems Research Program (CSRP) group to design and construct a second-generation device called the Tobermory.

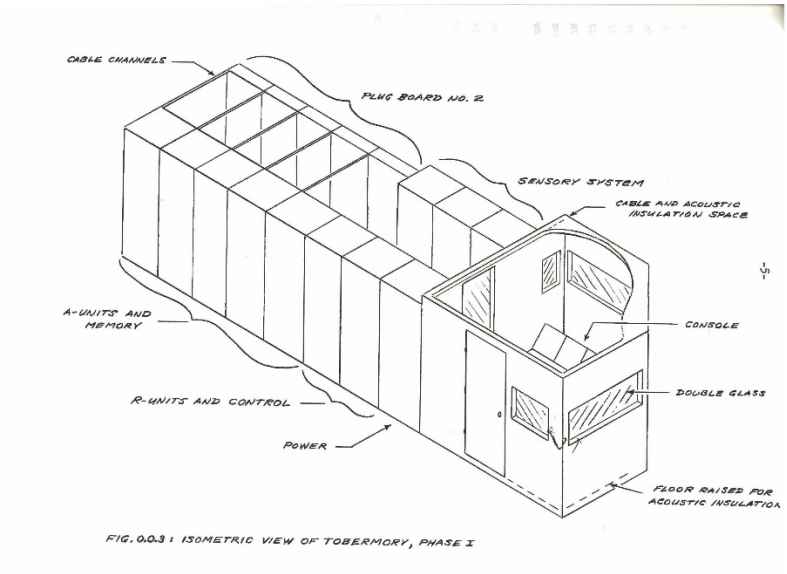

Figure 2: The Tobermory (Nagy, 1963b).

Figure 3: Tobermory Components (Rosenblatt, 1962).

The Tobermory took its name from a short story by Saki (H. H. Monroe) that featured a talking cat. As its name suggests, it would be a “phonoperceptron” and designed for audio input. Nagy’s dissertation, defended in 1962, was titled “Analogue Memory Mechanisms for Neural Nets” and examined different possible designs for analog memory devices. Some of the existing options examined by Nagy included more experimental electro-chemical devices such as electrolytic integrators and

solions and novel but difficult to use at scale film-based photochromic devices using slide projectors. Nagy settled on what was known as the "magnetostrictive read-out integrator," a device suggested by SRI's Charles A. Rosen. This was the tape-wound magnetic core memory device employed by the MINOS II and initially designed by SRI staff member Harold S. Crafts (Brain et. al., 1962). It also had the advantage of sharing features with the core memory used in conventional digital computers. The labor-intensive production of these memory devices, as Daniela K. Rosner et. al. argue, is one of several important sites of "hidden, feminized work" involved in the creation of mid-century computing (Rosner et. al., 2018). Addressing his selection of a tape-wound device for the Tobermory, Nagy wrote: "The chief virtue of the electromechanical integrator consists of its inherent stability. The 'weight' of a given connection is represented by a mechanical displacement, hence it is not subject to variation due to ambient changes or fluctuations in power supply level" (Nagy, 1962). Many existing analog alternatives, as Nagy notes in his survey, were subject to rapid decay, error, and sometimes were difficult to reinitialize or to erase previously stored values.

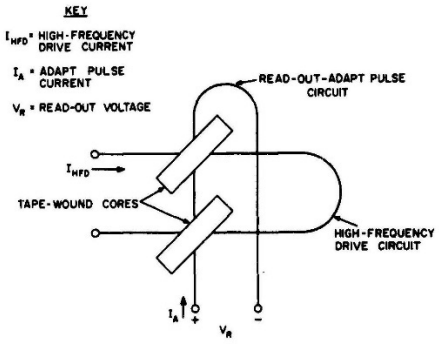

FIG. 9 BASIC TAPE-WOUND CORE-PAIR WEIGHT CIRCUIT

Figure 4: Schematic of Tape-Wound Core Memory for MINOS II (Brain et. al., 1962).

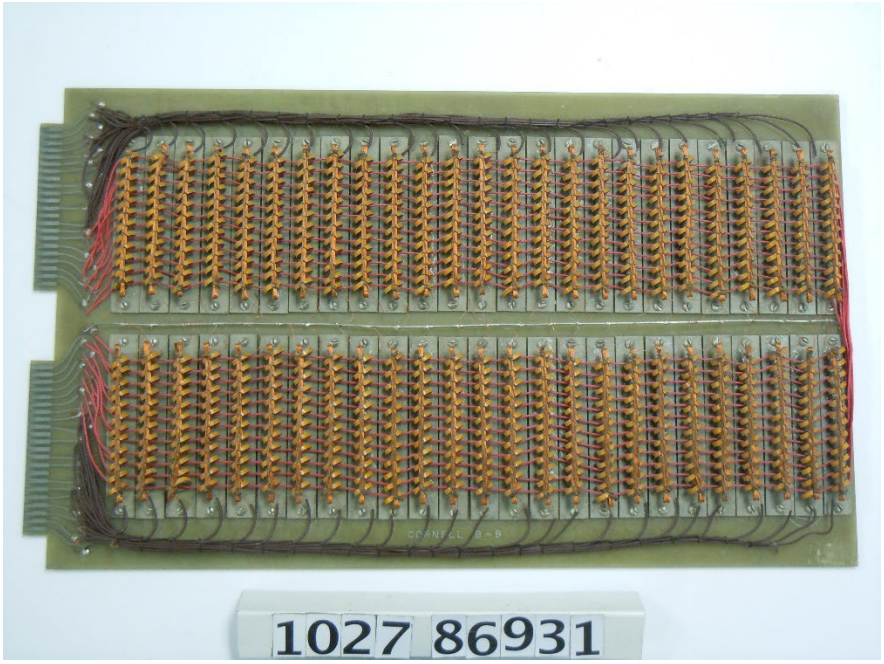

Figure 5: Tobermory Perceptron analog core memory. Courtesy of the Computer History Museum.

Despite the ongoing research and development of analog learning machines with memory devices during this period, many researchers were simultaneously implementing neural networks as simulated machines on conventional digital computers. In their justification for building a learning machine, the SRI MINOS team explained what they saw as the deficiency of digital computers: “Their major function in the present line of research is to simulate the performance of machine concepts which might be mechanized in some form which would be efficient (smaller, faster cheaper, etc.). The general-purpose digital machine thus appears as a research tool rather than as a final device for pattern recognition” (Brain et. al., 1960). In these simulations, the weights were stored in regular core memory during training and evaluation and persisted in various offline storage systems. The simulation of learning machines was necessary at the beginning of machine learning while engineers worked to construct analog machines and find appropriate memory devices, but this paradigm stuck as digital computers increased in speed and became easier to program and use. The appeal quickly became apparent to researchers. In an article summarizing his research into analog memory devices, Nagy speculated that advancements in digital computers might soon render analog memory obsolete. “In principle,” he wrote, “any pattern recognition machine using weighted connections may be simulated on a binary machine of sufficiently large capacity” (Nagy, 1963a). Specialized hardware for machine learning, although now fully digital and instrumented with layers of software, returned in the late 1980s and early 1990s during the high-performance massively parallel computer boom. Today, costly clusters of high-density graphical processing units (GPUs) and tensor processing units (TPUs) are being deployed to train very large models although these also execute software simulated learning machines.

Early machine learning was primarily directed toward the discrimination and classification of visual data. These models worked with highly simplified representations of images. They were not trained to generate new images. Today’s deep learning models in computer vision and the extremely popular Transformer-based large language models are now routinely used in generative applications. The size of these models combined with these new uses (themselves a function of model size), has prompted a reconsideration of the memory issue. The assumption that patterns of activation generalize,

as Hebb theorized in biological models, seems to be under pressure when applied to understanding the operation of artificial neural networks with billions or more parameters. There is strong evidence that large language models are memorizing examples from their training and that this behavior is more likely in large models (Carlini 2021). The retention of this information suggests that these patterns can be mapped. Research into the interpretability of deep learning models has discovered some of these patterns and demonstrated that sets of neurons can be edited to alter the model’s predictions (Meng et. al., 2022). This line of inquiry returns us to lingering important questions about the relation between learning and memory, the differences between generalization and memorization, and the location of memory in neural networks that were also present at the founding of the field of machine learning.

Bibliography

Borsellino, A., and A. Gamba (1961). "An Outline of a Mathematical Theory of PAPA," *Del Nuovo Cimento* 20, no. 2, 221–231. https://doi.org/10.1007/BF02822644.

Brain, Alfred E., Harold S. Crafts, George E. Forsen, Donald J. Hall, and Jack W. Machanik (1962). "Graphical Data Processing Research Study and Experimental Investigation." 40001-PM-60-91.91(600). Menlo Park, CA: Stanford Research Institute.

Carlini, Nicholas, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, et al. (2021). "Extracting Training Data from Large Language Models." In *Proceedings of the 30th USENIX Security Symposium*. 2633–2650.

Dobson, James E. (2023). *The Birth of Computer Vision.* University of Minnesota Press.
Hay, John C., Ben E. Lynch, David R. Smith (1960). "Mark I Perceptron Operators' Manual (Project Para)" VG-1195-G-5. Cornell Aeronautical Laboratory.

Hebb, Donald O. (1949). *The Organization of Behavior: A Neuropsychological Theory.* John Wiley and Sons.

McCulloch, Warren S., and Walter Pitts (1943). "A Logical Calculus of the Ideas Immanent in Nervous Activity." *Bulletin of Mathematical Biophysics* 5, 115–33.

Meng, Kevin, David Bau, Alex Andonian, and Yonatan Belinkov (2022). "Locating and Editing Factual Associations in GPT." *Advances in Neural Information Processing Systems*, 35, 17359-17372.

Nagy, George (1962). "Analogue Memory Mechanisms for Neural Nets." PhD diss. Cornell University.

Nagy, George (1963a). "A Survey of Analog Memory Devices." *IEEE Transactions on Electronic Computers* EC-12, no. 4: 388–93. https://doi.org/10.1109/PGEC.1963.263470.

Nagy, George (1963b). "System and Circuit Designs for the Tobermory Perceptron," Cognitive Research Program. Report No. 5. Ithaca, NY: Cornell University.

Rosenblatt, Frank (1962). "A Description of the Tobermory Perceptron." Cognitive Research Program. Report No. 4. Collected Technical Papers, Vol. 2. Edited by Frank Rosenblatt. Ithaca, NY: Cornell University.

Rosenblatt, Frank (1957). "The Perceptron: A Perceiving and Recognizing Automaton (Project PARA)." Report 85-460-1. Cornell Aeronautical Laboratory.

Rosenblatt, Frank (1958). "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain." *Psychological Review* 65, no. 6: 386–408. https://doi.org/10.1037/h0042519.

Rosner, Daniela K., Samantha Shorey, Brock R. Craft, and Helen Remnick (2018). "Making Core Memory: Design Inquiry into Gendered Legacies of Engineering and Craftwork." In *Proceedings of the 2018 CHI*

Conference on Human Factors in Computing Systems (CHI '18). ACM. https://doi.org/10.1145/3173574.3174105.

Tirumala, Kushal, Aram Markosyan, Luke Zettlemoyer, and Armen Aghajanyan (2022). “Memorization without Overfitting: Analyzing the Training Dynamics of Large Language Models.” In *Advances in Neural Information Processing Systems* 35. Edited by S. Koyejo et. al. 38274-38290. Vancouver, Canada: Curran Associates.

James E. Dobson (June 2023). “Memorization and Memory Devices in Early Machine Learning.” *Interfaces: Essays and Reviews on Computing and Culture Vol. 4, Charles Babbage Institute, University of Minnesota*, 40-49.

About the author: James E. Dobson is assistant professor of English and creative writing and director of the Institute for Writing and Rhetoric at Dartmouth College. He is the author of *Critical Digital Humanities: The Search for a Methodology* (University of Illinois Press, 2019) and *The Birth of Computer Vision* (University of Minnesota Press, 2023).