

# The Role of Machine Learning and Neural Networks Across the Decades

Paolo Didier Alfano

*Università degli Studi di Torino*

## Abstract

Artificial intelligence and machine learning based on neural networks have seen an ever-increasing diffusion in recent years. In this short essay, we will illustrate the birth and development of artificial neural networks by analyzing three vitally important papers on the subject. As we will see, these papers provide a fundamental contribution to computer science, not only in the field of artificial neural networks.

## 1 | Introduction

In recent years, neural networks, particularly thanks to the developments in deep learning techniques, have been attracting much attention. To realize this fact, it is enough to type the words *artificial neural network* into any search engine. In the news section, it will be possible to find articles from the world's most important newspapers of recent publication. It would be difficult to find a field of study to which the term *pervasive* can be applied as effectively. Recently, we have seen neural network-based algorithms perform activities considered achievable exclusively by humans: driving vehicles, painting, writing poems. These exceptional developments are partly the result of increased computational power, but are primarily the consequence of surprisingly ancient solid theoretical foundations. In the following, we will see how the context in which neural networks are born and develop has had a strong impact on the research and studies conducted. The path we will follow consists of three parts:

1. In the first part, we will study the origins of neural networks. We will see, thanks to the article by Warren McCulloch and Walter Pitts, the first study on artificial neural networks.
2. Next, we will look at the study that lays a theoretical foundation for neural networks, by Stephen Kleene.
3. Finally, we will see how, in an already mature context, the concept of the modern neural network based on automata was developed. For this last part, we will refer to a work by Marvin Minsky.

## 2 | The Origins: McCulloch and Pitts

The history of neural networks begins when Warren McCulloch meets Walter Pitts. Their first meeting dates back to 1942, in Chicago. These two personalities remained connected until their respective deaths, both occurring in 1969, a few months apart from each other.<sup>1</sup> In 1943, thanks to McCulloch's interest in the propagation of electrical signals in the cerebral cortex and Walter Pitts' early talent in logical-mathematical studies, the paper theorizing artificial neural networks for the first time was published[1].

The purpose of the paper is clear from the summary: due to the *all-or-none* nature of nerve activity, it can be represented using propositional logic.

Initially, some fundamental information regarding the biological neuron is provided: structure, propagation of the electrical signal, and particularly the concept of *state*.

Indeed, a neuron can be, at any given moment in time, either *excited* or *at rest*. The intuitive idea is that a neuron changes its state from "at rest" to "excited" if it receives a sufficient number of electrical stimuli from neighboring neurons. This number is called the *threshold value*. After a few milliseconds in an excited state, during which it may produce an electrical signal, the neuron returns to its resting state.

Connecting multiple neurons together, we obtain a *neural network*.

As we mentioned, when a neuron is excited, it can emit a signal. Neuronal signals are of two types:

- Excitatory: sent to another neuron, they contribute to reaching the threshold value of the recipient.
- Inhibitory: sent to another neuron, they prevent it from transitioning from the resting state to the excited state.

The introductory part concludes by analyzing a problem: a signal propagating from one end of the neural network to the other is destined to extinguish once it has traversed it. The solution to this problem is to introduce cycles that allow the signal to persist over time within the network. Therefore, the paper by McCulloch and Pitts first studies networks without cycles. In the last part, it deals with cyclic networks.

In this regard, the article introduces the concept of *order*: in a network, the order is the number of neurons to remove to make it acyclic. We note, in particular, that zero-order networks are already cycle-free.

Immediately after, artificial neural networks are introduced with some assumptions: neuron activity is Boolean; to excite a neuron, a certain number of excitatory signals are needed simultaneously; a single inhibitory signal to a neuron

---

<sup>1</sup>In particular, the life of Walter Pitts, rich in notable events, would deserve a separate essay. For completeness, we report an article that accurately describes Pitts' life, starting from his correspondence with McCulloch.

prevents its excitation; the structure does not change over time. The mathematical representation of McCulloch and Pitts is based on the symbolism of Carnap[2] and Russell[3]. As also noted by Stephen Kleene, this notation is less readable than that used in subsequent articles. The  $m$  neurons belonging to a certain network  $\mathcal{N}$  are represented using the notation

$$c_1, \dots, c_m$$

In particular, the neurons are divided into two groups:

- *Peripheral afferents*: the neurons that do not receive input from other neurons; they are the ones from which the initial signal propagates. We assume that the signal activating the peripheral afferents comes from the outside.
- All other neurons. Kleene refers to these neurons as *inner neurons*. They are excited over time thanks to the propagation of the signal produced by the peripheral afferents.

Furthermore, the notation  $N_i(t)$  indicates that neuron  $i$  is excited at time  $t$ . With  $N_i$ , we indicate the actions of neuron  $i$  over time.

The concept of *solution* is then introduced. A solution is a set of logical sentences of the following form: "neuron  $n$  is excited at a certain time  $z_1$  if and only if the logical combination of inputs along with the signal propagation over time allows its excitation." We report a generic logical sentence  $S_i$  belonging to the set of solutions:

$$S_i : N_{p+1}(z_1) \equiv Pr_i(N_1, N_2, \dots, N_p, z_1) \quad (1)$$

Where  $z_1$  is the time,  $Pr_i$  is the  $i$ -th predicate belonging to the set of solutions, and  $N_1, \dots, N_p$  are the peripheral afferents of the neural network.

Through the concepts just seen, the article introduces a new class of logical expressions called *temporal propositional expressions* or *TPE*. The propositions belonging to this logic have a single free variable identifiable with discrete time and are recursively defined as follows:

1. Every predicate with a single argument is a TPE.
2. The operations of disjunction  $\vee$ , conjunction  $\wedge$ , and negation  $\neg$  between TPEs produce a TPE. Furthermore, a TPE in which we appropriately replace the temporal value with its predecessors generates a TPE.
3. Nothing else is a TPE.

After introducing TPEs, McCulloch and Pitts proceed to prove the most important theorem about acyclic networks:

**Theorem 1.** *Every TPE can be implemented by a zero-order network.*

This allows us to assert that given a logical expression, it is always possible to represent it using a cycle-free neural network. Later, the article presents six additional theorems on acyclic networks. For example, they prove the complement of the theorem just stated: "given a zero-order network, such network can be solved by a TPE." They also demonstrate how raising the excitation threshold or explicitly inhibiting a neuron are equivalent processes. . .

Subsequently, the article proceeds to prove some properties of cyclic networks. The theorems related to cyclic networks, although of great interest, will not be presented here as they would require the introduction of a much more complex and unclear notation.<sup>2</sup>.

The article concludes with an assessment of the consequences of the theorems proved. Here are the two most important ones:

- The model of artificial neuron adapts well to neuronal activities and more generally to all inherently propositional activities.
- The presented model has strong causality but is strongly non-reciprocal in certain aspects. In fact, while the *all-or-none* nature of neural networks allows for precisely determining the evolution of a network state from the previous one, the reverse operation is not possible. This leads to an incomplete knowledge of the world.

An important point to note, which goes beyond the conclusions of McCulloch and Pitts, is that although the term *automaton* does not appear, many believe that the results obtained in this article lay the groundwork for finite state automaton theory. It would be more than ten years before the publication of articles by George Mealy[4] and Edward Moore[5], which would define automaton theory. However, we can assert that these articles and subsequent studies on automata derive from this initial formalization.

We conclude the analysis of the article with some considerations on the historical context in which the article was born. First and foremost, it is historically significant to note the influence of the work of McCulloch and Pitts on another prominent figure of the time, John von Neumann. In 1945, von Neumann presented the article [6] which gave rise to the well-known *von Neumann architecture*. It is interesting to note that the *E elements* introduced in von Neumann's article to describe circuits are strongly based on McCulloch and Pitts' neurons. Von Neumann's idea was to use the model just described by replacing neurons with vacuum tubes.

The strong influence of McCulloch and Pitts' model on von Neumann's model is due to their meeting at Princeton University in December 1943. During this occasion, Pitts presented the model developed with McCulloch, greatly impressing von Neumann.

---

<sup>2</sup>Even Stephen Kleene admits to having understood very little about the part on cyclic networks, as evidenced by his significant comment:

"[...] we found the part of their paper dealing with arbitrary net obscure"

Another important consideration concerns the timing of McCulloch and Pitts' work being published. One might think that the birth of the article is the result of a series of coincidences, including the fortuitous meeting between the two authors. However, we believe that the timing of the article's writing was not accidental.

In the years preceding the publication of the article, Nicolas Rashevsky, a Russian-born American mathematician and theoretical physicist, moved to Chicago. Rashevsky was one of the pioneers of mathematical biophysics and in the 1930s developed an initial model of neural networks on which McCulloch and Pitts based their propositional model.

During that period, Rashevsky was among the first to formulate a theory that unified various fields of study in the scientific domain. In this sense, he later introduced the concept of *organic sets*[7], with which he sought to represent an organism from different perspectives: biological, physical, and mathematical.

We believe that this idea of *unification* may have strongly influenced the development of the article just analyzed, and given the conjunction between biology and logic, we can rightfully place McCulloch and Pitts' article within this "unifying current". In support of this thesis, we recall that Pitts worked in Rashevsky's group from the late 1930s to the early 1940s, being impressed by his application of the physical-mathematical method to biology. Further supporting this, we note that McCulloch and Pitts' article was published in the Bulletin of Mathematical Biophysics, founded in 1939 by Rashevsky.

### 3 | A Solid Theory: Kleene

From the model of McCulloch and Pitts, many studies on neural networks have emerged. Among these, a particularly important role is played by Stephen Kleene[8]. Published in 1956, Kleene's work aims to understand what can be represented by a neural network. This question finds a precise and comprehensive answer in Kleene's writing:

All and only events belonging to a certain class of *regular events* are representable.

As mentioned earlier, studies related to neural networks have led to extremely interesting developments for computer science as a whole. The regular events defined by Kleene correspond—with some differences—to today's regular expressions.

The article is structured into three parts: the first part revisits the concepts illustrated by McCulloch and Pitts. Subsequently, regular events are introduced, demonstrating some of their properties. In the final part, there is a brief mention of a representation of networks using automata.

As mentioned earlier, in the first part of Kleene's article, he revisits what was discussed by McCulloch and Pitts. In particular, he makes a few additional

assumptions and reintroduces the proposition  $N_i(t)$  to indicate that neuron  $i$  is excited at time  $t$ . To indicate its quiet state, Kleene uses the notation  $\overline{N_i(t)^c}$  [8]. Given the propositional nature of the expressions used, Kleene introduces *truth tables* that represent the state of each neuron over a certain interval of time. Specifically, the size of these tables is  $k \times l$ , where  $k$  is the number of neurons in the network and  $l$  corresponds to the length in units of time of the interval considered, proceeding backwards from the present moment  $p$ . Note that the length  $l$  is not necessarily fixed.

Following this, Kleene introduces the following terms:

- *kl-formula*: a logical formula representable using  $k \times l$  tables.
- *event*: a term that can indicate the  $k \times l$  table or the statements representable by the table.

We present some events in the form of statements and one event in the form of a table:

1.  $N_1$  is excited at time  $p$ .
2.  $N_2$  is not excited at time  $p$  and  $N_1$  is excited at time  $p - 1$ .
3.  $N_2$  has been excited at some time.

t	$N_1$	$N_2$
p	1	0
p-1	1	1
p-2	0	1
$\vdots$	$\vdots$	$\vdots$

Table 1: A table with  $k = 2$  and unspecified  $l$

With this tabular event, we have represented a network with two neurons. Its interpretation is straightforward: at the present time  $p$ , only neuron  $N_1$  is excited; at time  $p - 1$ , both are excited, and so on...

As mentioned, the duration  $l$  of events is not fixed a priori. Referring back to the three non-tabular statements seen earlier, we note that the third has an unspecified temporal duration. Additionally, even the event in Table 1 has an unspecified duration.

For this reason, the concept of a *defined event* is introduced. A defined event is an event of *fixed duration*  $l$ , starting at time  $p - l + 1$  and ending at the present time  $p$ . Each defined event can be represented by a logical formula and consequently by a table. For example, if we were to limit the duration of the event in Table 1 to the last three time units, we would obtain a defined event representable by the logical formula

$$N_1(p) \wedge \overline{N_2(p)} \wedge N_1(p-1) \wedge N_2(p-1) \wedge \overline{N_1(p-2)} \wedge N_2(p-2)$$

Thanks to what has been said so far, the article proceeds to prove the following theorem:

**Theorem 2.** *For any formula-kl  $F$  and a defined event  $E$  on  $k$  neurons, there exists a cycle-free neural network corresponding to  $F$  that represents  $E$ .*

Following this, a series of corollaries and lemmas are proven. Through these, the article proceeds to demonstrate the following theorem:

**Theorem 3.** *Given a neural network  $\mathcal{N}$ , the state of a neuron at time  $p$  in the network is equivalent to the occurrence of a corresponding defined event.*

These proofs are followed by two additional corollaries and a section regarding indefinite events that we cannot present here.

The most interesting part of Kleene's article, however, concerns *regular events*. To discuss regular events, Kleene introduces the concept of *sets of tables* and defines operations on these sets. Assuming that a certain number of tables belong to a set of tables, consider two sets  $E$  and  $F$ . On these sets of tables, we can define the following operations:

- Sum or disjunction  $E \vee F$ : the set to which all tables that belonged to  $E$  or  $F$  belong
- Product or conjunction  $EF$ : the set to which all tables obtained by concatenating each table of  $E$  with each table of  $F$  belong. If a table in  $E$  had  $l_1$  rows and a table in  $F$  had  $l_2$  rows, the corresponding product table in  $EF$  will have  $l_1 + l_2$  rows.

We notice that the two operations just defined are associative and iterable. Indeed, Kleene decides to use a concise notation to represent them. For example,  $E^0 F$  represents  $F$ ,  $E^1$  represents  $E$ ,  $E^3$  represents  $EEE$ , and so on. Let's look at the last operation defined on sets of tables:

- Iteration  $E * F$ : the set to which all tables of the infinite sum belong

$$F \vee EF \vee EEF \vee \dots$$

in other words, all tables of the form

$$\sum_{n=0}^{\infty} E^n F$$

Finally, Kleene defines the *unit set* as the set containing exactly one table.

Thanks to the preceding definitions, we can say that:

- The *regular set* (of tables) corresponds to the smallest class of sets of tables containing the unit set, the empty set, and closed under the operations of sum, product, and iteration.
- A certain event is said to be *regular* if there exists a regular set of tables that describes it. By this we mean that an event occurs if its input is described by one of the tables in the set.
- The representation of regular sets on which we apply one or more times the operations of sum, product, and iteration is called a *regular expression*.

We note a simple consequence of the preceding definitions: every finite set of tables is obviously regular.

Following this, a series of simple algebraic transformations for regular expressions of obvious significance are proposed <sup>3</sup>:

$$\begin{array}{llll}
E \vee E = E & E \vee F = F \vee E & (E \vee F) \vee G = E \vee (F \vee G) \\
(EF)G = E(GF) & (E * F)G = E * (FG) & (E \vee F)G = EG \vee FG \\
& & \vdots
\end{array}$$

Introducing the necessary terminology and operations, the article proceeds to demonstrate the two most important theorems:

**Theorem 4.** *For every regular event, there exists a neural network that represents it through the excitation of a certain internal neuron at time  $p$ , when the neurons of the network at initial time 1 are in a suitable state.*

**Theorem 5.** *For every event constructed from a regular event by applying the operations  $\wedge$ ,  $\vee$ , or  $-$ , there exists a neural network that represents it through the excitation of a certain internal neuron at time  $p$ , when the neurons of the network at initial time 1 are in a suitable state.*

These two results are very important for the development of neural networks because if the purpose of the article was to determine what can be represented by a neural network, the answer is now clear and concise: neural networks can represent all and only regular events.

The concluding part of the article is dedicated to a brief study on automata and primitive recursion. Two theorems are presented: the first connects automata, neural networks, and regular events; the second links regular events to primitive recursion.

**Theorem 6.** *In every finite state automaton initialized at time 1, the event represented by a certain existing state at time  $p$  is regular.*

**Theorem 7.** *For every regular event  $E$  involving input neurons  $n_1, \dots, n_k$ , the predicate  $E(p)$  is primitive recursive in the predicates  $N_1(t), \dots, N_k(t)$ .*

---

<sup>3</sup>The entire set of algebraic transformations can be found on page 24 of the article



A treatment of primitive recursion is beyond the scope of this brief essay; further clarifications on this are available in another article by Kleene [9]. We conclude the analysis of the article by reflecting on the historical moment in which it was published. We cannot definitively justify Kleene’s interest in neural networks. The only certain fact available to us is that in 1951, Kleene spent the summer at the research and development institute (RAND), receiving the institute’s support in writing the article just studied. However, the work done by Kleene, like that of McCulloch and Pitts, does not come without precedent. The years preceding the publication saw great excitement in the field of computability. In 1936, the Turing Machine model was proposed, and throughout the 1930s, there were several attempts to formalize the notion of computability. Furthermore, from the late 1930s to the 1950s, various studies led to significant developments in recursion theory. Kleene himself, through his studies, contributes to laying the groundwork which, combined with the studies of McCulloch and Pitts, almost naturally leads to the publication of the article just studied.

## 4 | Neural Networks and Automata: Minsky

As we have observed, the two previous articles lay the groundwork for the theory of neural networks. In addition to providing valuable theoretical elements, they outline the use of automata to represent neural networks.

The representation of neural networks through automata will increasingly gain ground in the years to come, which is why we present a particularly relevant work [10] by Marvin Lee Minsky. Minsky’s work represents the culmination of concepts developed over the previous twenty-five years, particularly advancing a model based on automata of undeniable practical utility.

The article is structured into two parts: the first introduces neurons in the form of automata used to construct networks of increasing complexity. The second part demonstrates the equivalence between neural networks and finite state machines, along with some considerations regarding the expressive power of the latter. The initial concepts naturally focus on neurons and connections of inhibitory/excitatory types. These components are illustrated in Figure 1.

As in the previous articles, Minsky adheres to the classical model where the only delay in the network is an internal delay within neurons between receiving input and producing output. For completeness, we note that this choice differs from other models [11][12], where components dedicated to generating delay are proposed.

The remainder of the first part is dedicated to presenting various types of automata. We highlight a couple of particularly interesting ones:

- Memory component: This allows us to remember that a certain cell was active in the previous state. To achieve this, we simply use a feedback

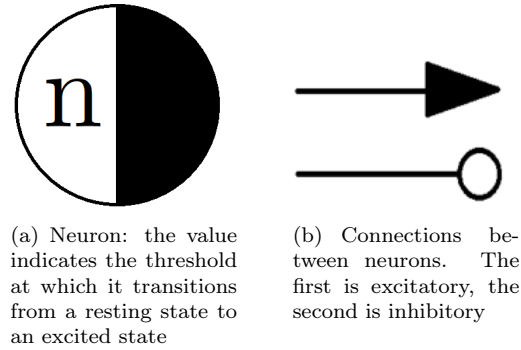


Figure 1: Basic components of a neural network in the form of automata

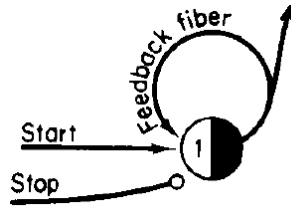


Figure 2: Memory component

loop, as shown in Figure 2.

When the neuron receives the start signal, since its threshold is 1, it transitions from a resting state to an excited state. Being excited, it sends a signal to the rest of the network and also to itself. In the subsequent time steps, the neuron will remain excited, continuing to send signals through the feedback loop. . . Therefore, in the following time steps, the neuron will remain excited until it receives the inhibitory stop impulse.

Note that feedback loops give rise to cycles. As mentioned in the article by McCulloch and Pitts, this enhances expressiveness and prevents the neural network from "turning off" after propagating a signal.

- Counter component: It is easy to understand that if we were able to "count," we could perform much more complex operations. We show two examples of counters in Figure 3.

The first one is a simple counter capable of counting up to two. By initially exciting the top neuron with an input, it remains excited thereafter due to the feedback loop  $x$ . Upon receiving a second signal, the bottom neuron also transitions from its quiescent state and produces an output to indicate that we have effectively counted up to two. Additionally, when the bottom neuron is excited, it sends a signal to inhibit both neurons, resetting the network to its initial state.

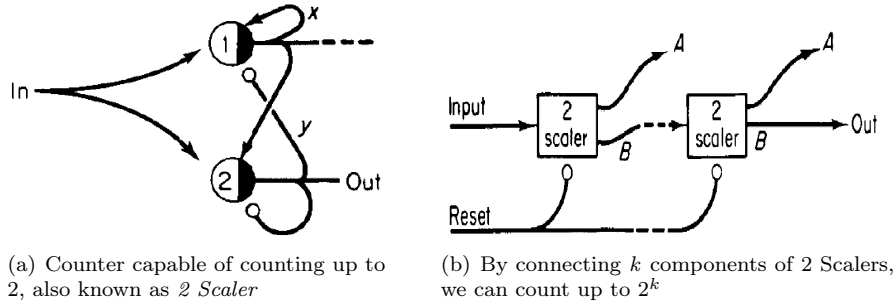


Figure 3: Counter components

Suppose we encapsulate the described component and use it in series  $k$  times as shown on the right side of Figure 3, we achieve a counter capable of counting up to  $2^k$ .

The second part of the article introduces some theoretical elements and revisits the studies of Mealy and Moore to demonstrate the following theorem:

**Theorem 8.** *Every finite state automaton is equivalent to, and can be simulated by, some neural network.*

At this point, Minsky proceeds to study some issues regarding the expressiveness of automata, aiming to identify a minimal set of components capable of representing every neural network. This set is referred to as the *universal basis*. Initially, the author attempts to construct a universal basis using only AND and OR nerve cells<sup>4</sup>.

Unfortunately, these components are not sufficient, and to understand why, we must consider the issue of *monotonicity*. A component is said to be *monotonic* if it is not possible to decrease the output size by increasing the input size.

Both the AND and OR components are monotonic. This is easily understood: consider a network  $R$  with a certain number of inputs and outputs. By replicating it and adding an excitatory signal, the number of output signals must be at least equal to the number of output signals from  $R$ .

It is easy to understand that to obtain a non-monotonic network, it is necessary to use inhibitory signals as well. There are some components called *non-monotonic* that can decrease the output size while increasing the input size, all of which make use of inhibitory signals. Examples of non-monotonic components are shown in Figure 4. The last non-monotonic component is particularly important as it constitutes a universal basis on its own.

<sup>4</sup>The structure of such nerve cells is quite intuitive: an AND nerve cell receives two or more excitatory signals, and its excitation threshold is always equal to the number of signals it receives. An OR nerve cell receives two or more excitatory signals, and its excitation threshold is always one.

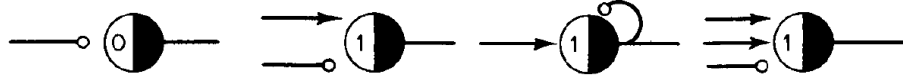


Figure 4: Non-monotonic components

To conclude, we also consider the historical context in which Minsky’s article was written. Certainly, Kleene’s research discussed in Chapter 3 may have influenced his work, but we believe that the greater influence came from other studies.

An author frequently cited by Minsky in the article is John von Neumann. We believe that the development of automata capable of performing complex operations such as arithmetic derives from von Neumann’s machine model. Moreover, as discussed in Chapter 2, there is a strong connection between von Neumann’s model and neural networks. Given the close relationship between neural networks and automata, we believe that Minsky’s work may have been influenced by von Neumann’s studies.

Furthermore, when Minsky moved to the Massachusetts Institute of Technology in 1958, we believe he was guided in writing the article by the aforementioned studies of Mealy [4] and particularly those of Moore [5].

Between Moore’s research in 1956 and the publication of Minsky’s article in 1967, automata theory had time to spread and influence the studies of many, including Minsky.

## 5 | Conclusions

In the preceding pages, we have traced the origins of neural networks. We examined the model of McCulloch and Pitts and how the concept of neurons influenced various fields of computer science seemingly unrelated to neural networks: computability theory, automata theory, and computing systems.

Next, we explored Kleene’s studies and their theoretical contributions combined with the valuable invention of regular expressions.

Finally, we studied the work of Minsky, who made a significant contribution to the representation of neural networks.

At this point, considering the influence of the McCulloch and Pitts model on von Neumann’s studies, it would be interesting to consider what direction computer science would have taken without their work.

Nevertheless, we can assert that the history of neural networks has brought with it a surprising amount of new ideas and concepts, playing a

## | References

- [1] Warren S. McCulloch e Walter Pitts *A Logical Calculus of Ideas Immanent in Nervous Activity*, Bulletin of Mathematical Biophysics, volume 5, 1943.
- [2] Carnap Rudolf *The Logical Syntax of Language*, Open Court Publishing, 1937.
- [3] Bertrand Russell e Alfred N. Whitehead *Principia Mathematica*, Cambridge University Press, 1925.
- [4] George H. Mealy *A Method for Synthesizing Sequential Circuits*, Bell System Technical Journal: 1045–1079, 1955.
- [5] Edward F. Moore *Gedanken-experiments on Sequential Machines*, Automata Studies, Annals of Math, Princeton University Press: 129–153, 1956
- [6] John von Neumann *First Draft of a Report on the EDVAC*, 1945
- [7] Nicolas Rashevsky *Organismic Sets and Biological Epimorphism*, Bulletin of Mathematical Biophysics, volume 29, 1967.
- [8] Stephen C. Kleene *Representation of events in nerve nets and finite automata*, Princeton University Press, 1956.
- [9] Stephen C. Kleene *Introduction to metamathematics*, North Holland Publishing Co. , 1952.
- [10] Marvin L. Minsky *Computation: Finite and Infinite Machines*: 32-66, Prentice Hall, 1967.
- [11] Arthur W. Burks e Hao Wang *The Logic of automata* JACM 4: 193-218 e 279-297, 1957.
- [12] Irving M. Copi, Calvin C. Elgot e Jesse B. Wright *Realization of events by logical nets*, JACM 5, num. 2: 181-196, 1958.