

Hopfield Network의 기억력 감퇴

문승현 · 도현진
서울과학고등학교

Memory Degeneration of Hopfield Network

Seunghyun Moon · Hyunjin Do
Seoul Science High School

국문 초록

Hopfield Network는 Ising Model을 기반으로 하여 뇌의 구조를 형상화한 네트워크 중 하나로, 노드의 상태를 정보로 저장한다. 사람의 뇌는 나이를 들어가며 기억력 감퇴 현상이 발생하고, 일부 사람에 대해서는 치매가 발병하기도 한다. 이러한 과정은 뉴런 사이의 연결이나 뉴런에 손상이 발생하며 생기는 현상임이 알려져 있다.

본 연구에서는 Hopfield Network를 구성하는 변수 중 일부를 사람의 뇌에서의 기억력 감퇴 현상의 원인과 같은 방법으로 손상하여(Edge-Defected Brain Model과 Node-Defected Brain Model) 네트워크가 저장하고 있던 정보가 소실되는 과정을 정량적으로 분석하였다. 두 개의 모델 모두에서 Defection에 따른 기억력 감퇴 현상은 Logistic Curve가 나타남을 관찰할 수 있었다.

주제어: Hopfield Network, Monte Carlo Simulation, Alzheimer's Disease, Defection of Network, Logistic Curve

Abstract

Hopfield Network is a neural network based on Ising model, which saves each state of nodes. Memory degeneration occurs in human brain due to aging or Alzheimer's disease. The cause of memory degeneration is known as defection of connections between neurons or death of neurons.

We defected Hopfield Network, presenting defection of connections as Edge-Defection and death of neurons as Node-Defection. The memory degeneration of Hopfield Network on defection is analyzed quantitatively. In both model, we could investigate Logistic Curve.

Keywords: Hopfield Network, Monte Carlo Simulation, Alzheimer's Disease, Defection of Network, Logistic Curve

I. 서론(Introduction)

1.1. 필요성과 목적

Hopfield Network는 뇌의 기억을 형상화하는 구조로 Ising Model에 기초를 둔 이론이다. 이 네트워크는 매우 단순하지만 높은 정확도를 가지는 기억을 형상화한다. 특히 실제 사람의 뇌가 가지고 있는 뉴런과 뉴런들 사이의 연결을 형상화한 것과 비슷한 모습을 가지고 있어 뇌에서 일어나는 과정을 그대로 그려보는 데에 적합하다.

사람의 뇌에서는 노화나 치매 등의 사유로 기억력 감퇴 현상이 발생한다. 이때 노화나 치매는 많은 연구에서 뉴런 사이 연결의 파괴나 뉴런의 파괴와 밀접한 관련이 있음을 알려준다 [5-8]. 본 연구에서는 이러한 뉴런 사이의 연결과 뉴런의 파괴를 각각 Edge-Defected Brain Model과 Node-Defected Brain Model로 나누어 분석함으로써 기억력 감퇴 현상이 나타나는 과정을 정량적으로 분석하고자 한다. Hopfield Network의 사용을 통하여 실제 뇌에서의 모습과 비슷한 방향으로 변화하는 과정을 구현할 수 있으리라 기대하며, 더 나아가 노화와 치매에서의 차이로 제시되는 치매에서는 뉴런의 파괴가 훨씬 더 많이 일어난다는 이론을 본 연구에서 검증해 보고자 한다.

1.2. 선행연구 및 관련 이론

1.2.1. Hopfield Network

Hopfield Network는 집합적 계산 능력을 가지고 있는 많은 수의 two-state variable로 이루어진 neural network이다. 이 네트워크는 일종의 물리적 시스템으로 에너지가 최소가 되는 방향으로 시간이 지남에 따라 상태가 수렴하게 된다. 이때의 에너지는 0 또는 1의 값을 가지고 있는 각 노드의 값과 계 자체의 해밀토니안 행렬에 의해 정해진다. 이러한 경향성은 Ising 모델에서의 해를 구하는 과정과 직접적인 유사성을 가지고 있다. 이런 네트워크의 고유 상태는 뇌의 기억을 나타낼 수 있는 모델로 평가받는다. [1, 2] 이러한 hopfield network는 associative memory의 일종으로 손상된 정보가 주어진 상황에서 손상된 정보를 저장된 정보 중 하나로 복원해 낼 수 있다 [3].

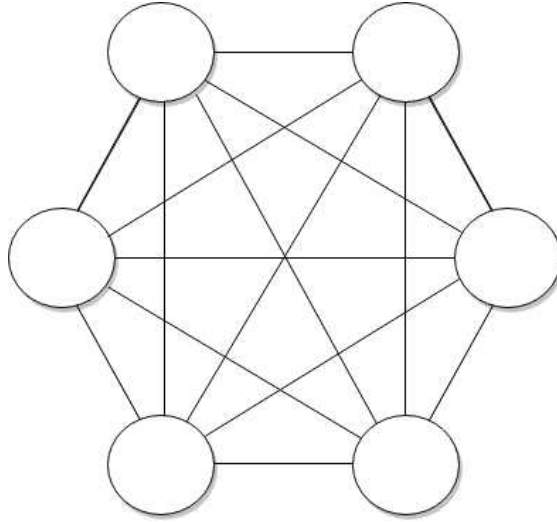


Fig 1. 노드의 개수가 6개인 Hopfield Network의 일부를 형상화한 다이어그램

Hopfield Network에서 사용하는 에너지는 아래 수식과 같은 형태를 가지고 있다.

$$E(V) = -\frac{1}{2} \sum_{i,j} T_{ij} V_i V_j + \sum_i U_i V_i$$

여기서 V 는 노드의 값을 표현한 벡터로, $V_i = 0 \text{ or } 1$ ($i = 1, 2, \dots, N$)으로 주어진다. T 는 노드들 사이의 상호작용을 표현한 값으로, T_{ij} 는 i 번째 노드와 j 번째 노드의 상호작용을 나타내며 $T_{ii} = 0$ 이다. 마지막으로 U_i 는 i 번째 노드에 걸려있는 위치에너지 장을 나타낸다. [4]

Hopfield Network의 목적은 N 개의 노드로 이루어진 시스템에 크기가 N 인 일차원 벡터 n 개를 저장하는 것으로, 저장된 벡터와 비슷한 벡터가 초기 상태로 주어진 상황에서 시간이 지나면 저장된 벡터로 수렴하여야 한다. John Hopfield는 1982년에 아래 수식으로 T 와 U 를 정한다면 잘 수렴하는 계를 만들 수 있음을 실험적으로 밝혀내었다.

$$\begin{aligned} T_{ij} &= \frac{1}{N} \sum_r (2V_i^{(r)} - 1)(2V_j^{(r)} - 1), (i \neq j) \\ T_{ii} &= 0 \\ U_i &= 0 \end{aligned}$$

여기서 $V^{(r)}$ 은 저장하고자 하는 n 개의 벡터 중 r 번째 벡터를 뜻한다. [2]

본 네트워크는 1982년의 연구 결과로부터 T 와 U 를 정하는 다양한 새로운 방법들이 제시되고 발전되어 왔고, 어떠한 방식으로 T 와 U 를 정하는지가 Hopfield Network가 얼마나 잘 정보를 저장할 수 있는지를 결정하게 된다. 이를 위해서 최근에는 아래와 같이 에너지를 아래와 같이 더욱 간단하게 정의하게 되었다.

$$E = \sum_{i=1}^n \sum_{j=1}^n J_{ij} V_i V_j$$

여기서 V 벡터는 지금까지와 달리 각 성분이 1 or -1의 값을 가지고 있게 되고, $n \times n$ 행렬 J 하나가 T 와 U 의 역할을 모두 수행하도록 정의되어 있고, 단순한 내적 두 번만으로 정리가 되기에 어렵지 않게 계산할 수 있다. 또, J 행렬에 변화를 주면 에너지의 변화 양상이 바뀌어 새로운 정보를 저장하거나 정보를 삭제할 수 있게 된다.

1.2.2. Dementia & Alzheimer's Disease

치매는 사람의 뇌에서 뉴런 사이 연결의 파괴나 뉴런의 파괴와 밀접한 관련이 있음이 알려져 있다 [5-7]. 그러나 노화로 인한 뇌의 기능 저하 또한 뉴런 사이 연결의 파괴나 뉴런의 파괴로 인해 발생한다. 여기서 노화로 인한 기능 저하는 비교적 뉴런의 파괴가 적게 일어나고 증상이 덜함을 확인할 수 있었으며, 13세부터 101세 사람의 뇌를 분석한 결과 치매의 발병은 뉴런의 파괴가 유의미하게 많이, 그리고 훨씬 더 넓은 부위에서 무작위적으로 관찰할 수 있게 하였다 [8].

치매가 발병하는 과정 중 뉴런 사이 연결의 파괴와 뉴런의 파괴에 큰 영향을 주는 질병은 알츠하이머 질병으로, 두 병명이 혼용될 정도로 많은 치매 환자들이 알츠하이머 질병을 앓는다. 알츠하이머 질병의 근본적인 원인이 정확히 밝혀지지는 않았다. 따라서 치매와 알츠하이머 질병을 분석하기 위해서는 뉴런 사이 연결의 파괴와 뉴런의 파괴를 분석하는 것으로 충분하고 더 많은 분석을 수행하기 어렵다.

1.2.3. Logistic Curve

Logistic Curve는 아래 함수로 정의되는 함수와 비슷한 형태의 그래프를 뜻하여 S자 모양을 가지고 있다.

$$f(x) = \frac{1}{1 + e^{-x}}$$

음의 무한대 방향에서 0으로 수렴하고 양의 무한대 방향에서 1로 수렴하는 본 그래프는 다양한 분야에서 응용되는데, 특히 인구 증가의 모델 등의 성장을 나타내는 다양한 모델에서 전체적인 모습을 묘사하는 데에 적극적으로 사용된다 [9].

1.2.4. Percolation Theory

Percolation theory는 노드와 그 노드들의 연결들로 구성된 네트워크의 모습을 분석하는 이론으로, 기하학적인 상 전이에 대하여 분석한다 [10]. 노드들의 연결이 깨져 나가는 과정에서 네트워크를 통과하기 쉬운 정도를 측정한 transport property를 입자들이 채워진 정도인 Filler content에 대하여 그래프를 그렸을 때, Logistic Curve를 찾아볼 수 있다 [11]. 그리고 이러한 과정에서 transport property가 급격하게 변화하는 시점의 filler content를 percolation threshold라고 칭한다.

II. 연구 방법 및 이론(Method & Theory)

2.1. 연구 방법

2.1.1. Brain Model

Hopfield Network는 1.2.1.에서 소개된 수식으로 정의가 된다. 여기서 마지막에 소개된 수식에서 J 행렬을 조정하는 방식으로 뇌를 모델링하도록 한다. 여기서 에너지를 계산하기 위해 위의 식을 아래와 같이 변형하기도 한다.

$$E = \sum_{i=1}^n H_i V_i$$
$$H_i = \sum_{j=1}^n J_{ij} V_j$$

당연하게도 같은 값으로 정해지지만, H 벡터는 각 위치에 다른 노드들과 J 행렬이 걸어준 장으로 생각을 할 수 있다. 이렇게 세워진 모델은 손상된 정보를 기존의 정보로 되돌릴 수 있게 된다. 이때 되돌리는 과정은 해당 손상된 정보와 가장 비슷한 정보 중 에너지 함수의 국소적인 최솟값을 찾는 과정이다. 이는 Monte Carlo 시뮬레이션을 이용하여 계산할 수 있다. 조금 더 구체적인 과정은 아래 Fig 2.와 같다.

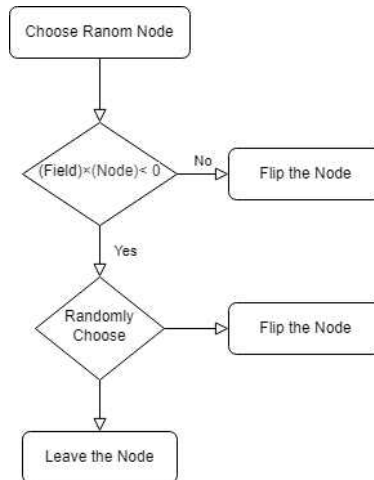


Fig 2. Monte Carlo 시뮬레이션을 위해서 진행하는 업데이트 과정의 Flow Chart

위 Flow Chart는 근처의 국소적으로 안정한 평형점으로 이동하기 위해서 진행되는 한 번의 스텝으로, 해당 스텝을 많은 횟수 반복하면 근처의 국소적인 곳에 도달하여 수렴하게 된다. 먼저 수많은 노드 중 하나의 노드를 무작위로 선택한 뒤, 해당 노드와 H 벡터의 해당 성분의 부호를 비교한다. 둘을 곱했을 때 양수라면 음수가 되었을 때 에너지가 더욱 감소할 것이므로 노드의 값을 뒤집어주는 것이 에너지를 감소하도록 하는 것이다. 그러나 둘을 곱한 값이 이미 음수인 시점에 확률적으로 노드의 값을 뒤집는다. 이것은 어떤 시점에 아주 국소적으로 에너지가 최저인 시점에 도달하더라도 이를 지나갈 수 있게 해주는 장치로, 더 정확하고 빠른 수렴을 위해 필요하다. 여기서 랜덤하게 고르는 기준은 $\exp(-\beta \Delta E)$ 의 확률로 노드를 뒤집는 것이다. 이때의 β 는 이 계의 온도를 의미하고 있으며 어느 정도로 국소적인 부분을 무시할 것인지 정해주는 기준이 되어 실험적으로 값이 정해지게 된다.

2.1.2. Healthy Brain Model

우선 전통적으로 연구된 Hopfield Network를 건강한 상태의 뇌로 가정하여 연구를 진행하였다. Hopfield Network는 주어진 정보들을 이용하여 즉각적으로 해당 정보들이 저장된 J 행렬을 계산해 낼 수 있고, 2.1.1.에서 소개한 방식을 이용하여 손상된 정보들을 수렴시키며 확인해볼 수 있다. 이때 가장 기초적인 식을 사용했는데, 이는 다른 방식으로 연구된 수많은 J 행렬들은 데이터의 크기가 매우 큰 경우에 대해서만 제대로 작동하는 것이 검증되었기 때문에 사용하는 것은 적합하지 않았다.

2.1.3. Edge-Defected Brain Model

Edge-Defection은 우리 뇌가 충격을 받았을 때 뉴런과 뉴런 사이의 시냅스가 깨지는 경우를 형상화한 것과 같다. 이를 만드는 방법은 J 행렬에서 임의의 값을 0으로 만드는 것이다. 이를 뉴런에서의 그림으로 형상화하면 아래와 같다.

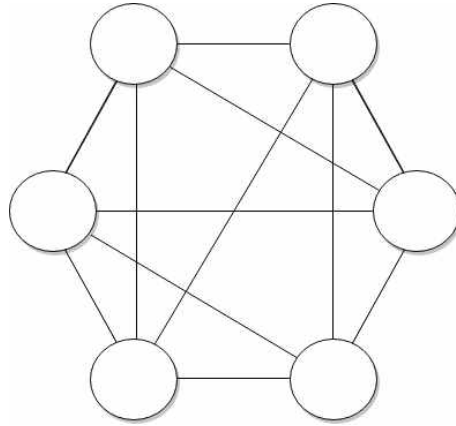


Fig 3. Hopfield Network에서 Edge-Defection이 일어난 경우의 다이어그램. 노드들 사이의 몇 개의 연결고리가 사라짐을 확인할 수 있다. 이러한 Defection은 알츠하이머의 발생이나 자연적인 노화에 따라 자연스레 발생하기도 한다.

이때 랜덤한 숫자가 아니라 0으로 만드는 이유는 충격이 가해지거나 비정상적으로 뇌가 발달하는 경우 특정 방향으로 연결이 진행되는 것보다 아예 연결이 사라지는 것이 훨씬 합당하다고 바라봤기 때문이다. 본 연구는 추가로 실제로 적용되는 인공지능 모델에게 영향을 줄 수 있다. 일부 인공지능 모델은 Over fitting의 방지를 위해 Dropout이라는 알고리즘을 사용하는데, 이는 본 연구에서 진행할 Edge-Defection과 매우 유사한 구조를 가진다. 또, Ising Model 등의 자석을 형상화한 모델에서는 노드들 사이의 거리가 주요한 요소로 작용하는데, 이는 이미 Defection이 특정한 형태를 보이고 이루어진 것과 매우 비슷하다.

2.1.4. Node-Defected Brain Model

Node-Defection은 우리 뇌가 충격을 받았을 때 특정 뉴런이 완전히 죽어버리는 것을 형상화한 것과 같다. 이를 만드는 방법은 J 행렬에서 죽을 노드와 연결된 모든 값을 0으로 만드는 것이다. 이를 그림으로 나타내면 Fig 4.와 같이 나타난다.

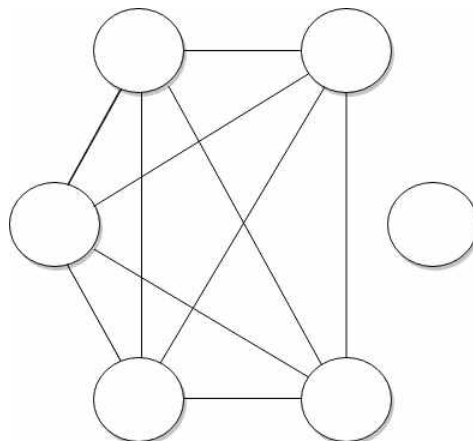


Fig 4. Hopfield Network에서 Node-Defection이 일어난 경우의 다이어그램. 우측 말단의 노드와 다른 노드들 사이의 연결고리가 모두 사라짐을 확인할 수 있다. 이러한 Defection은 노드의 죽음을 의미하며 알츠하이머의 발생이나 자연적인 노화에 따라 자연스레 발생하고, 실제 알츠하이머의 가장 주요한 원인 중 하나로 제시된다.

이는 해당 뉴런이 더 이상 타 뉴런에 아무런 영향도 주지 못하고, 그저 값을 랜덤하게 정해주는 뉴런이 되었음을 형상화한다. 본 연구는 2.1.3.의 Edge-Defected Brain Model과 유사하게 Dropout, Ising Model과의 밀접한 관련을 유사한 방식으로 가지고 있다. 다만, 타 분야의 경우 특정 노드가 아예 사라지는 것을 연구하지는 않아 어느 정도의 차이점이 있다고 생각할 수 있다. 그러나 Ising Model에서는 Defection이 발생하여 일부 원자가 있어야 되는 부분에서 사라진 것으로 생각한다면 매우 유사한 것을 알 수 있다.

2.2. Evaluation of Memory Degeneration

위 2.1.에서는 어떠한 Brain Model을 사용할 것인지 소개하였다. 이제 Brain Model이 정해지면 해당 Brain Model에서 어느 정도의 Memory Degeneration이 일어났는지 평가하는 과정을 알아보자. 우선 2.1.1.의 과정을 통하여 각 J 행렬에 대해 주어진 데이터와 인접한 국소적인 에너지가 최소인 위치를 찾아낼 수 있다. 이렇게 찾아내는 과정은 각각 저장된 정보가 수정된 다음에 적용돼야 하는 것이다. 조금 더 구체적인 과정으로는 J 행렬을 저장할 데이터를 이용하여 2.1.1.에서의 과정으로 얻어낸 뒤, 각 J 행렬에 대해 데이터를 입력하는 과정을 반복한다. 데이터 입력은 주어진 저장된 데이터들을 특정 개수의 노드만큼 손상한 뒤, 모델에 입력하여 어디로 수렴하는지 관찰한다. 이때 수렴한 위치가 손상 전의 위치와 어느 정도만큼의 오차를 가지는지 기록하여 모든 데이터에 대해 평균을 계산함으로써 해당 J 행렬이 가지는 Memory Degeneration을 계산할 수 있게 된다. 그러나 본 방법에서 데이터가 손상되는 방향이 무작위적으로 정해질뿐더러, 저장될 데이터와 수렴하는 과정까지도 전부 무작위성을 가지고 있어 필연적으로 Memory Degeneration은 무작위성을 가지고 있을 수밖에 없다. 따라서 반복적인 수행을 통해 통계적으로 연산을 진행해야 합리적인 Memory Degeneration을 측정해낼 수 있다.

2.3. 실험 과정

본 시뮬레이션을 위해 Python3를 이용하였고, Numpy를 이용하여 데이터를 다루었다. Python3를 이용한 이유는 본 연구를 응용하여 연구를 진행한다면 쉽게 다른 연구와 연계하여 연구를 진행할 수 있을뿐더러 호환성이 좋아 다른 연구의 결과를 본 연구에 적용하여 추가적인 연구를 진행할 수 있다는 점이 있다. 그러나 본 연구에는 유의미하게 큰 연산량이 요구되어 연산 시간이 크게 필요하였다. 따라서 Numpy 라이브러리를 이용하여 데이터를 다룸으로써 연산 속도를 매우 큰 폭으로 높임과 동시에 훨씬 더 편한 방식으로 데이터를 다룰 수 있었다.

2.3.1. Brain Model

Brain Model은 하나의 class를 만들어 다루었다. State이라는 이름을 가진 본 class는 J 행렬과 H 벡터, 그리고 각 노드의 값을 저장하고 있고, 이들을 다루어 계산하는 다양한 함수들을 가지고 있다. 이때의 함수로는 각 변수를 대입하는 함수, Monte Carlo 시뮬레이션을 통해 스텝을 진행하는 함수, Memory Degeneration을 측정하는 함수가 있다. 여기서 가장 먼저 Brain Model의 노드 크기를 200으로 정했다. Healthy Brain Model에서 유의미하게 작은 Memory Degeneration을 보여주는 동시에 노드 크기가 프로그램의 연산량에 미치는 영향이 매우 커 크게 잡기가 어려웠기 때문이다. 저장된 데이터의 크기와 노드의 크기 비율에 관한 선행연구를 근거로 20개의 데이터를 참값으로 설정하였을 때 적당한 값이 나올 수 있다[4].

2.3.2. Healthy Brain Model

1.2.1.에서 주어진 T_{ij} 와 U_i 에 관한 식을 사용하도록 한다. 해당 식을 이용해 값을 계산한 뒤 J 행렬에 대입하는 방식으로 구해낼 수 있는데, 이는 $O(n^2)$ 으로 시간복잡도가 작지 않지만 해야 하는 계산이 작고 n 자체도 크지 않아 길지 않은 시간에 Brain Model을 얻어낼 수 있다. 이렇게 얻은 옳은 값을 따로 저장해두어 아래 Defected Brain

Model을 얻을 때 이용하도록 한다.

2.3.3. Edge-Defected Brain Model

Edge-Defected Brain Model은 위 2.3.2.의 Healthy Brain Model을 변형하여 얻어진다. 2.1.3.에서 언급한 바와 같이 정해진 개수의 J 행렬의 성분들을 0으로 만드는 방식으로 Edge-Defection을 진행하게 되는데, 200개의 성분을 순차적으로 제거해 가며 총 2만 개의 성분을 제거하는 경우까지 총 101개의 경우를 실험하였다.

2.3.4. Node-Defected Brain Model

Node-Defected Brain Model은 위 2.3.2.의 Healthy Brain Model을 변형하여 얻어진다. 2.1.4.에서 언급한 바와 같이 노드를 무작위로 정하여 해당 노드와 연관된 모든 J 행렬의 성분을 0으로 만든다. 이때 처음 고르는 노드에 의해 제거되는 개수는 399개이다. 총 200개의 노드를 하나씩 순차적으로 제거해 가며 실험을 진행하였다.

2.3.5. Evaluation of Memory Degeneration

Memory Degeneration의 측정을 위해서는 Monte Carlo 시뮬레이션이 필수적이었다. 위 [그림 2]와 같은 Flow Chart를 가지는 과정을 이용하여 진행하였다. 여기서 Monte Carlo 시뮬레이션을 위해 이용하는 β 의 값은 아래 Healthy Brain Model이 잘 작동하는 값인 10을 택하였고, 해당 환경에서 시뮬레이션은 의미 있는 결과를 가짐을 검증하였다. 또, Healthy Brain Model의 작동 여부를 통해 모델이 수렴하는 스텝의 개수를 1000으로 정하였다. 평균적으로 각 노드를 5번 확인하는 것이다. 또, Memory Degeneration을 정하는 과정에서 손상하는 노드의 수를 10으로 정하여 5%의 손상이 일어나게 해 어느 정도의 복구가 가능하도록 유도하였다. 스텝의 수가 10번의 손상 횟수보다 압도적으로 많기에 수렴하는 데에 문제가 생기지 않으리라고 전망하였다.

결론적으로 본 연구에서 얻어내고자 하는 것은 Edge-Defection과 Node-Defection의 영향으로, 각각 손상 정도에 따라 Memory Degeneration을 위에서 주어진 수치에 대해 측정하였다. (Edge-Defection은 0부터 20000까지 200 간격으로, Node-Defection은 0부터 200까지 1 간격으로 측정했다) Memory Degeneration의 이론적인 최대치는 200의 절반인 100으로, 두 Defection 모두 적지 않은 비율의 정보를 손상한 채로 측정할 것이므로 100에 도달할 수 있을 것으로 바라보았다. 본 연구를 위해 작성한 프로그램은 아래 <참고 자료>에서 찾아볼 수 있으며, 누구나 실행할 수 있는 조건으로 작성되었다.

III. 결과(Data & Result)

3.1. Edge-Defected Brain Model

먼저 Edge-Defected Brain Model에서 얻은 결과는 아래 Fig 3.과 같다.

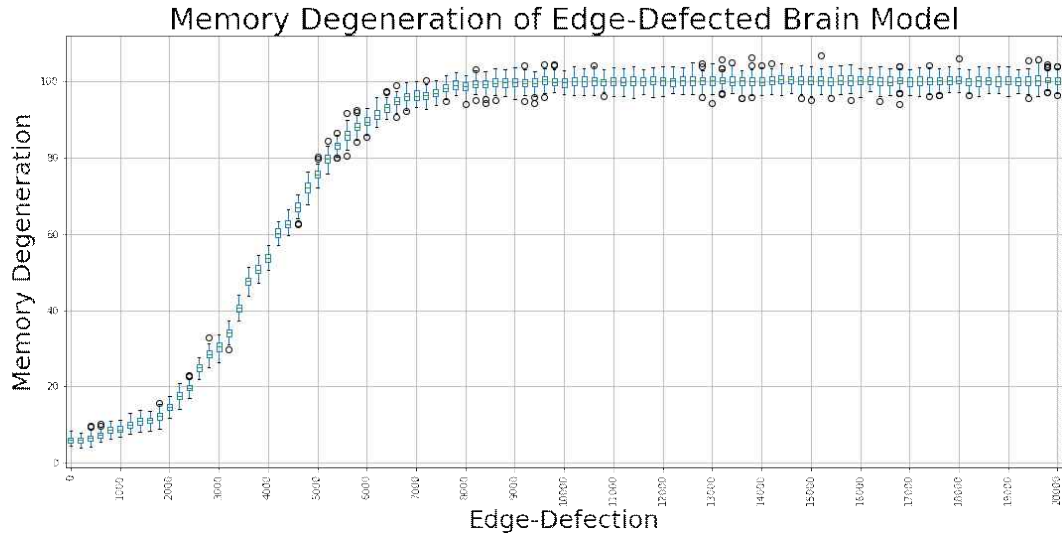


Fig 5. Edge-Defection에 따른 Memory Degeneration을 측정한 결과를 Box Plot을 이용해 그린 그래프. Memory Degeneration 함수가 무작위성을 가지고 있어 각 점에 대해 100번의 측정을 통해 통계적으로 처리되었고, 중간중간 Outlier들이 존재함도 확인할 수 있었다. 그렇지만 Memory Degeneration의 변동성이 크지 않고 전체적인 경향성의 관찰이 가능했음을 알 수 있다. Defection이 10000에 도달할 때까지 Logistic Curve가 관찰되며 그 이후부터는 이론적인 최대치인 100에 도달하여 유지됨을 관찰할 수 있었다.

위 Fig 5.에서는 Edge-Defection에 따른 Memory Degeneration을 측정하였고, 이를 Box Plot을 이용해 그래프를 그려내었다. Edge-Defection에 대해 Memory Degeneration의 값이 범위로 정해지는 것은 2.2.에서 언급했듯이 Memory Degeneration이 무작위성을 가지고 있기 때문이다. 이에 따라 100번의 반복적인 측정을 통해 통계적인 처리를 수행하였다. Fig 5.에서 관찰할 수 있듯이 Memory Degeneration의 변동성이 경향성에 지장을 줄 정도로 크지 않아 전체적인 경향성의 관찰이 가능함을 확인할 수 있다. Defection이 10000 이하인 시점에는 Logistic Curve가 관찰되고, 그 이후에는 최대치인 100에 도달한 채로 유지되는 것으로 Logistic Curve에 매우 부합하는 모습을 보여준다. 조금 더 자세한 관찰을 위해 10000 이하인 시점의 그래프만 수집한 그림은 아래 Fig 6.와 같다.

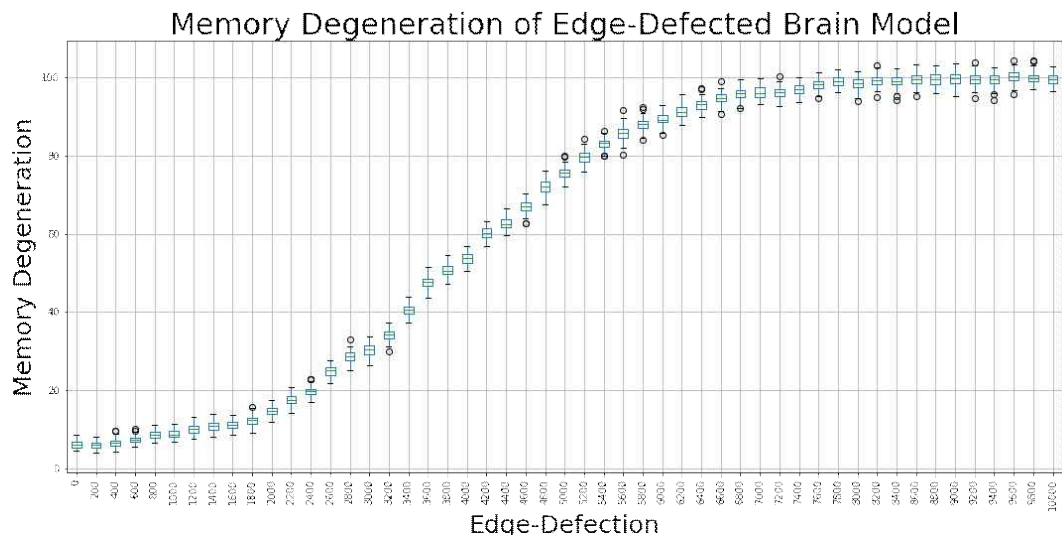


Fig 6. 조금 더 자세한 관찰을 위해 Edge-Defection을 10000까지만 잘라 관찰한 그림이다. 적극적인 성장을 겪는 4600 전후에서는 오차가 매우 작지만, 이론적 최대치인 100에 근접하는 단계에서 비교적 오차가 커짐도 확인할 수 있었다.

3.2. Node-Defected Brain Model

Node-Defected Brain Model에서 얻은 결과는 아래 Fig 5와 같다.

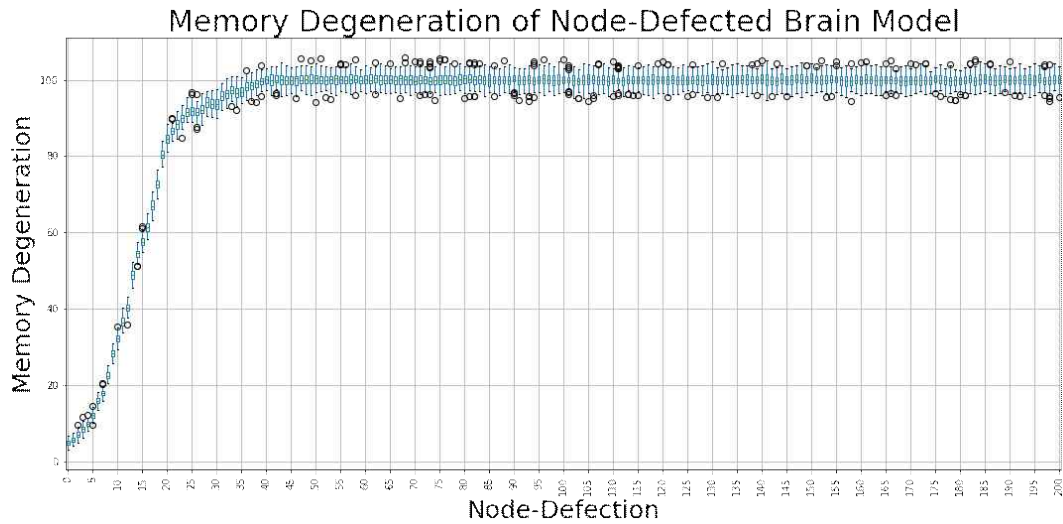


Fig 7. Node-Defection에 따른 Memory Degeneration을 측정한 결과를 Box Plot을 이용해 그린 그래프. Edge-Defection에 대한 경우와 같이 Memory Degeneration 함수를 통계적으로 처리하였다. Memory Degeneration의 변동성이 Edge-Defection에 대한 경우보다 상대적으로 컸음에도 전체적인 경향성의 관찰이 가능했음을 알 수 있다. Defection이 50에 도달할 때까지 Logistic Curve가 관찰되며 그 이후부터는 이론적인 최대치인 100에 도달하여 유지됨을 관찰할 수 있었다.

위 Fig 7.에서는 Node-Defection에 따른 Memory Degeneration을 측정하였고, 이를 Box Plot을 이용해 그래프를 그려내었다. Edge-Defection에 대한 경우와 같은 방법으로 Memory Degeneration을 통계적으로 처리하였다. Fig 7.에서 관찰할 수 있듯이 Memory Degeneration의 변동성이 Edge-Defection에서의 경우보다 상대적으로 크지만, 경향성에 지장을 줄 정도로 크지는 않아 전체적인 경향성의 관찰이 가능함을 확인할 수 있다. Defection이 50 이하인 시점에는 Logistic Curve가 관찰되고, 그 이후에는 최대치인 100에 도달한 채로 유지되는 것으로 Logistic Curve에 매우 부합하는 모습을 보여준다. 조금 더 자세한 관찰을 위해 50 이하인 시점의 그래프만 수집한 그림은 아래 Fig 8.과 같다.

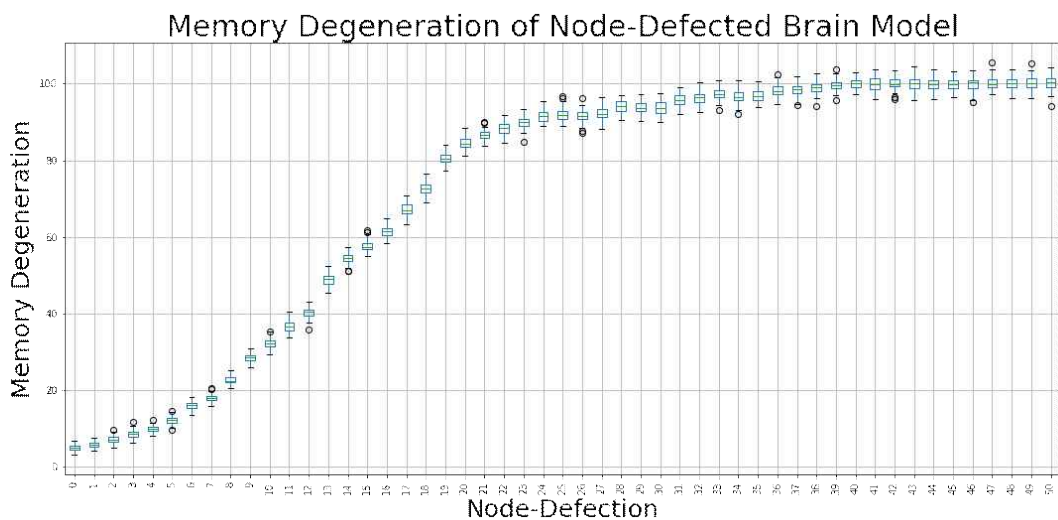


Fig 8. 조금 더 자세한 관찰을 위해 Node-Defection을 50까지만 잘라 관찰한 그림이다.

3.3. Logistic Curve of Memory Degeneration

위 3.1.과 3.2. 모두에서 Logistic Curve를 관찰할 수 있었기에 이들의 그래프를 조금 더 정량적으로 분석할 필요를 느꼈다. 이를 위해서 각각 잘린 그래프에 대해 Logistic Function을 scipy의 minimize 함수를 이용해 그래프와 가장 비슷한 Logistic Function을 얻어냈다. 여기서 사용한 Logistic Function의 꼴은 아래와 같다.

$$f(x) = \frac{A}{1 + ce^{-kx}} + C$$

Logistic Regression on Memory Degeneration of Edge-Defected Brain Model

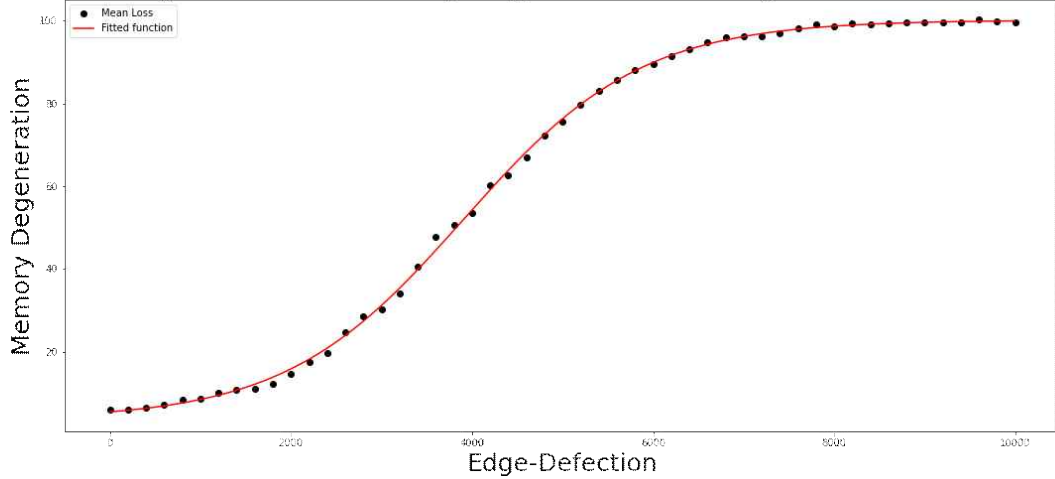


Fig 9. Edge-Defected Brain Model에서의 Memory Degeneration 그래프와 가장 잘 맞는 Logistic Function을 찾아낸 그래프를 그린 그림이다.

위 Fig 9.에서 알 수 있듯이 Edge-Defected Brain Model에서의 Memory Degeneration은 Logistic Curve와 매우 유사한 그래프를 가지고 있다. 아래 Fig 10.에서도 유사하게 Node-Defected Brain Model에서의 Memory Degeneration이 Logistic Curve를 그리고 있음을 확인할 수 있었다.

Logistic Regression on Memory Degeneration of Node-Defected Brain Model

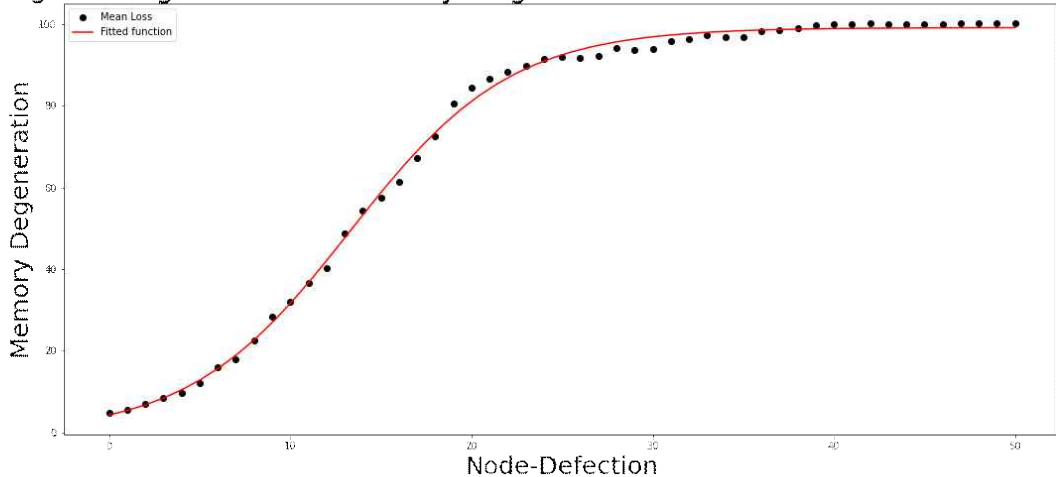


Fig 10. Node-Defected Brain Model에서의 Memory Degeneration 그래프와 가장 잘 맞는 Logistic Function을 찾아낸 그래프를 그린 그림이다.

이렇게 정리한 Fig 9.과 Fig 10.에서 그린 그래프를 그리며 얻은 수치는 아래 Table 1.에 MSE와 함께 정리하였다. A 와 C 의 값에서 봤을 때 최댓값이 100에 수렴하고 최솟값이 거의 0임을 알 수 있었다. 또, MSE의 값에서 관찰할 수 있듯이 Node-Defection이 상대적으로 Logistic Curve에서의 오차가 큰 것을 확인할 수 있었다.

여기서 percolation theory를 적용하여 생각한다면, percolation theory에서 분석하고자 하는 것과 본 연구에서 수행한 분석에는 연구하고자 하는 내용에 차이가 존재한다. 그러나 두 이론 모두 노드 사이의 연결 양상에 따라 계가

특정 성질을 가지는 양상을 분석하고, 이를 분석하는 과정에서 상전이를 찾아낼 수 있다. 그렇다면 본 연구에서 얻은 percolation threshold에 대응되는 값은 Edge-Defection에서는 3903, Node-Defection에서는 13.3이 된다. Percolation theory에서 묘사하는 바를 이용하면 해당 percolation threshold가 상전이가 일어나는 시점으로 계가 연결되는 시점이다. 본 연구에서 바라보는 치매의 발생이 이 시점에 일어난 것으로 가정한다면, edge-defection이 3903, node-defection이 13.3인 시점에 치매가 발병한 것이라 말할 수 있다. 13.3의 node-defection을 위해서는 edge-defection이 5,143개 있어야 한다. 따라서 node-defection이 발생하는 것이 edge-defection보다 치매의 발병에 악영향을 준다.

분류	A	c	k	C	MSE
Edge-Defection	96.385	52.872	1.0165×10^{-3}	3.7295	0.8020
Node-Defection	99.537	20.130	0.22559	-0.42071	1.4298

Table 1. Fig 9.과 Fig 10.을 그리며 얻은 Edge-Defection과 Node-Defection에 대한 Logistic Function의 변수들과 MSE를 정리한 표. Node-Defection이 상대적으로 더 큰 오차가 나타남을 관찰할 수 있다.

IV. 고찰(Discussion)

본 연구에서는 Hopfield Network를 Monte Carlo 시뮬레이션을 이용하여 프로그래밍하였고, 이렇게 만든 Hopfield Network에서 변수들을 적당히 바꾸어 어떻게 저장된 정보가 변화하는지 관찰하였다. 저장된 정보는 Edge-Defection과 Node-Defection이 발생함에 따라 감소하였고, 순차적으로 정보가 삭제되었다. 이러한 과정은 실험을 설계하는 과정에서 제시한 Defection의 발생 양상이 유의미하게 저장된 정보에 영향을 줄 수 있었고, 이러한 영향이 특정한 경향성을 가지고 있음을 Logistic Curve를 띄는 것을 통해서 관찰해 낼 수 있었다. 이러한 경향성은 percolation theory와 유의미한 관련이 있고, 이러한 연관성이 합리적인 치매의 발병 시점 예측에 도움을 주며, 두 Edge-Defection과 Node-Defection의 영향을 비교하는 데 필요했다.

V. 결론(Conclusion)

본 연구에서 제시한 두 가지 Defection 중 치매에서는 무작위적인 Node-Defection이 더욱 많이 발생하였음이 선행연구에서 분석되었다 [8]. 이에 따라 삭제한 총 J_{ij} 의 수에 대한 Memory Degeneration을 비교해 보았을 때, Edge-Defection에서는 25%가 파괴된 시점에 Memory Degeneration이 100에 도달하였지만, Node-Defection에서는 50개의 노드 (즉, 17,500개의 J_{ij})가 파괴된 시점에서 도달하였다. 이는 Edge-Defection이 상대적으로 Memory Degeneration에 영향을 더 많이 줄이 확인되는 것이다. 이에 따라, 비 국소적인 Edge-Defection을 발생시키는 술과 약품이 뇌에 끼치는 영향이 크다는 것과 국소적인 뇌의 Defection이 전체적인 정보에 아주 큰 영향을 주기는 어렵다는 것으로 결론지을 수 있었다.

VI. 제언(Proposal)

본 연구에서 관찰할 수 있던 가장 큰 현상은 Memory Degeneration이 Defection에 대해 Logistic Curve를 그린다 는 것이다. 이는 실험적으로 관찰된 결과로, 다양한 의미를 내포하고 있을 것으로 예상된다. 그러나 이것에 대한 정성적인, 혹은 정량적인 설명을 하는 것은 불가능했고, Logistic Curve를 그리게 되는 이유에 대해 이론적으로 접근하여 설명해내는 방법이 있을 것으로 내다본다. 또, Percolation Theory를 이용해 조금 더 심도 있게 분석을 수행하여 치매의 발병 시점이 Percolation Threshold와 관련이 있다는 가설을 뒷받침할 필요가 있다.

VII. 참고 문헌(Reference)

- [1] Little, W. A. (1974). "The Existence of Persistent States in the Brain". **Mathematical Biosciences**, 19 (1-2): 101-120. doi:10.1016/0025-5564(74)90031-5.
- [2] Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. **Proceedings of the National Academy of Sciences**, *79*(8), 2554-2558. <https://doi.org/10.1073/pnas.79.8.2554>
- [3] **Autoassociative memory**. (2022, May 26). Wikipedia. https://en.wikipedia.org/wiki/Autoassociative_memory
- [4] Bruce, A. D., Gardner, E. J., & Wallace, D. J. (1987). Dynamics and statistical mechanics of the Hopfield model. **Journal of Physics A: Mathematical and General**, *20*(10), 2909-2934. <https://doi.org/10.1088/0305-4470/20/10/035>
- [5] Gómez-Isla T, Hollister R, West H, Mui S, Growdon JH, Petersen RC, Parisi JE, Hyman BT. Neuronal loss correlates with but exceeds neurofibrillary tangles in Alzheimer's disease. *Ann Neurol*. 1997 Jan;41(1):17-24. doi: 10.1002/ana.410410106. PMID: 9005861.
- [6] Niikura T, Tajima H, Kita Y. Neuronal cell death in Alzheimer's disease and a neuroprotective factor, humanin. *Curr Neuroparmacol*. 2006 Apr;4(2):139-47. doi: 10.2174/157015906776359577. PMID: 18615127; PMCID: PMC2430668.
- [7] Wirths O, Zampar S. Neuron Loss in Alzheimer's Disease: Translation in Transgenic Mouse Models. *Int J Mol Sci*. 2020 Oct 30;21(21):8144. doi: 10.3390/ijms21218144. PMID: 33143374; PMCID: PMC7663280.
- [8] West, M. J., Coleman, P. D., Flood, D. G., & Troncoso, J. C. (1994). Differences in the pattern of hippocampal neuronal loss in normal ageing and Alzheimer's disease. **The Lancet**, *344*(8925), 769-772. [https://doi.org/10.1016/s0140-6736\(94\)92338-8](https://doi.org/10.1016/s0140-6736(94)92338-8)
- [9] McKendrick, A. G., & Pai, M. K. (1912). XLV.—The Rate of Multiplication of Micro-organisms: A Mathematical Study. **Zenodo**. <https://doi.org/10.1017/s0370164600025426>
- [10] Broadbent, S., & Hammersley, J. (1957). Percolation processes: I. Crystals and mazes. *Mathematical Proceedings of the Cambridge Philosophical Society*, 53(3), 629-641. doi:10.1017/S0305004100032680
- [11] Jiang, Q., Yang, J., Hing, P., & Ye, H. (2020). Recent advances, design guidelines, and prospects of flexible organic/inorganic thermoelectric composites. *Materials Advances*, 1(5), 1038-1054. <https://doi.org/10.1039/D0MA00278J>

보충 자료(Supplemental Information)

- [1] GitHub Repository of Programs used for this research.
<https://github.com/MoonRainy21/Memory-Degeneration-of-Hopfield-Network>