

제조현장 개발자를 위한

# AI 기반 데이터 분석 실전 강의

프롬프트 엔지니어링으로 배우는 제조 현장 데이터 분석

2026.02.23-2026.02.24 (2일, 16시간)

Prepared by Daekyeong Kim

Ph.D.

Copyright 2022. Daekyeong all rights reserved



## 2. 교육 시간표

### 2일차 (8시간)

시간	세션	내용	방식
09:00-09:30	오리엔테이션	1일차 복습 및 2일차 개요	강의 / 실습
09:30-10:30	현장 데이터 분석 기초	데이터 분석 신경망 구조 개요 / 인공지능망과 퍼셉 트론의 원리와 구현	강의
10:30-12:00	“	데이터 정제 및 이상치 제 거	실습
12:00-13:00	점심시간		
13:00-14:30	AI 학습 데이터 변환	AI 학습 데이터 변환	실습
14:30-14:45	“	불량률 예측 및 원인 분석	
14:45-16:15	“	종합 실습 및 케이스 스터 디	실습
16:15-17:30	“	종합 실습 및 케이스 스터 디	실습
17:30-18:00	마무리	Q&A 및 마무리	-

# 『Day 2』

## AI 활용 데이터 변환 및 분석

- Part 1. 1일차 복습 및 2일차 개요
- Part 2. 현장 데이터 분석 기초
- Part 3. AI 학습 데이터 변환





# 1. 1일차 핵심 내용 복습



## 1일차 핵심 내용 요약

- 생성형 AI의 개념과 활용 분야
- 효과적인 프롬프트 작성 원칙 (구체성, 맥락, 예시)
- ChatGPT 등 기본 사용법
- 제조 현장 데이터 분석의 중요성



# 1. 1일차 핵심 내용 복습



## | 학습자 참여

- "1일차에서 가장 인상 깊었던 내용은?" (간단한 발표)
- "실제 업무에 적용해본 경험?" (경험 공유)



# 1. 1일차 핵심 내용 복습

## 실습

- 실습 1-1: 데이터 구조 파악 (15분)
- 실습 1-2: 기초 통계량 계산 (15분)
- 실습 1-3: 불량률 분석 (15분)
- 실습 1-4: 데이터 시각화 요청 (15분)



## 2. 2일차 학습 목표 및 데이터 소개



### | 실습 데이터

- 파일명: defect\_prediction\_data.csv
- 내용: 생산 공정 1000건의 데이터
- 변수: 생산번호, 온도, 압력, 속도, 진동, 설비연식, 원료등급, 교대조, 요일, 불량여부
- 목표: 불량 발생 패턴 분석 및 원인 규명



## 2. 2일차 학습 목표 및 데이터 소개



### ■ 학습 로드맵

- 데이터 탐색 → 데이터 정제 → 데이터 변환 → 불량 예측 → 종합 분석



# 『Day 2』

## AI 활용 데이터 변환 및 분석

- Part 1. 1일차 복습 및 2일차 개요
- Part 2. 현장 데이터 분석 기초
- Part 3. AI 학습 데이터 변환





# 1. 데이터 분석 신경망 구조

## 개요

- 제조 공정 불량(불량=0/1) 예측을 위한 “데이터 분석 + 신경망(Neural Network) 구조”의 설계 아웃라인
- 현재 데이터 컬럼: 온도, 압력, 속도, 진동, 설비연식 + 범주형(원료등급/교대조/요일), 타겟=불량



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약



에이다 러브레이스 통찰력

"... 해석엔진은 숫자 이외의 것도 처리할 수 있을 것이다. ... 예를 들어 화음과 음조를 해석엔진의 표기에 맞출 수 있다면, 해석엔진은 꽤 복잡한 곡을 작곡할 수도 있다." [Ada1843]

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1969년  
퍼셉트론의 한계 제시



마빈 민스키, 시모어 페퍼트

19C-20C 초반  
생체 신경망



이반 파블로프

1956년  
퍼셉트론



프랭크 로젠블랫

요한 카를 프리드리히 가우스(Johann Carl Friedrich Gauss)

선형 회귀 분석에서 가중치를 사용하여 오류를 최소화하는 개념을 도입했습니다.

1843

19세기  
초



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

폴 브로카

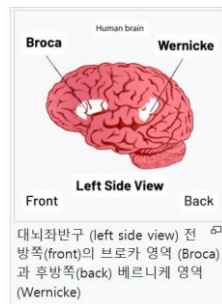


폴 피에르 브로카

폴 브로카(Paul Broca)

실어증을 앓는 뇌 손상 환자를 연구하는 과정에서 인지 기능을 담당하는 특정 영역이 왼쪽 뇌에 있다는 것을 처음 발견

브로카 영역(Broca's area)



대뇌좌반구 (left side view) 전  
방쪽(front)의 브로카 영역 (Broca)  
과 후방쪽(back) 베르니케 영역  
(Wernicke)

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1969년  
퍼셉트론의 한계 제시



마빈 민스키, 시모어 페퍼트

19C-20C 초반  
생체 신경망



이반 파블로프

1956년  
퍼셉트론



프랭크 로젠블랫

1861



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약



카밀로 골지  
이탈리아 해부학자

카밀로 골지(Camillo Golgi) 골지 염색기법을 개발하여 단일 뉴런의 복잡한 구조를 형태학적으로 연구

골지 염색기법을 적극 활용한 신경생물학자 산티아고 라몬 이 카할(Santiago Ramon y Cajal)

‘뇌의 기능적 단위는 뉴런’

‘뇌의 작용은 신경세포를 차례로 옮겨 가는 신호 흐름’

뉴런주의(neuron doctrine) 주장

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1969년  
퍼셉트론의 한계 제시



마빈 민스키, 시모어 페퍼트

19C-20C 초반  
생체 신경망



이반 파블로프

1956년  
퍼셉트론



프랭크 로젠블랫

1890

산티아고 라몬 이 카  
할

신경과학자





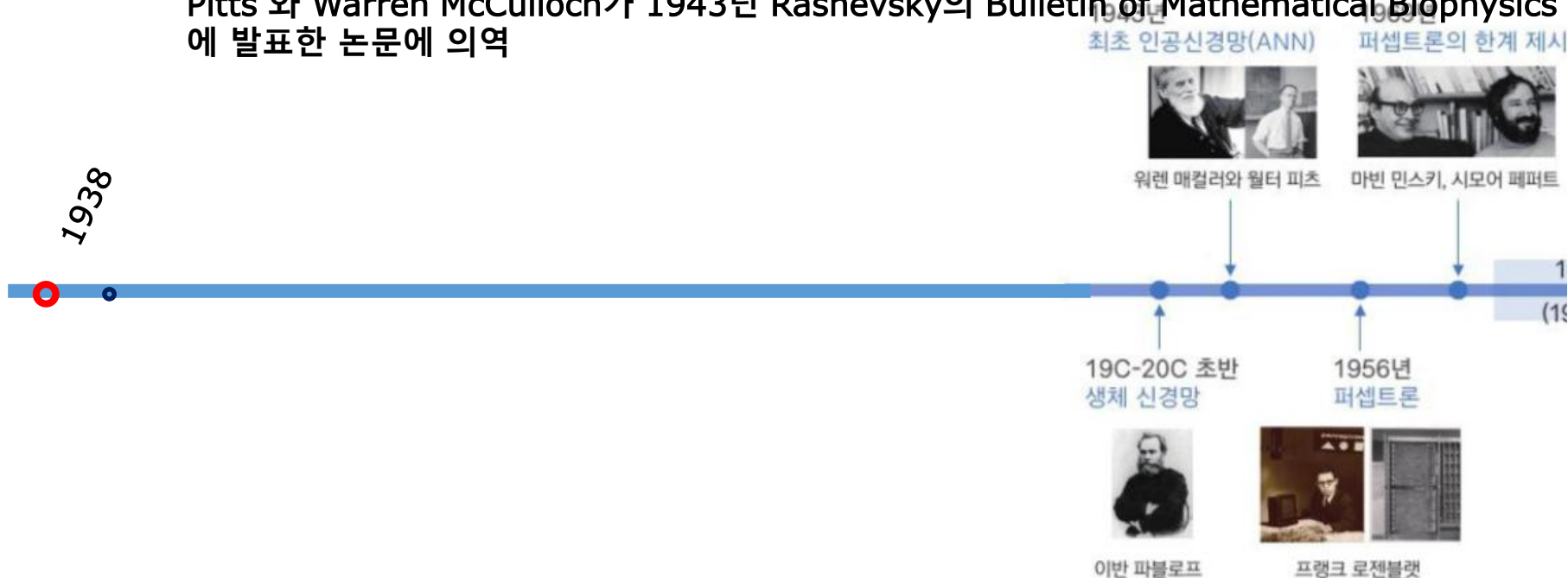
# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

니콜라스 라세프스키(Nicolas Rashevsky)

Mathematical Biophysics

최초로 뉴런과 뉴런들로 구성된 신경구조의 작동을 수학적 모델로 구현. 그의 학생 Walter Pitts 와 Warren McCulloch가 1943년 Rashevsky의 Bulletin of Mathematical Biophysics에 발표한 논문에 의역

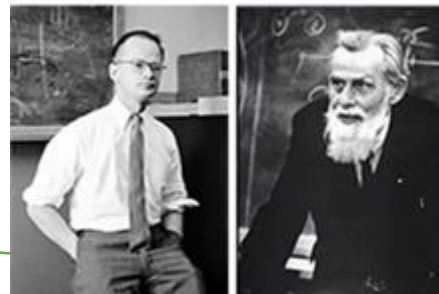




# 1. 데이터 분석 신경망 구조

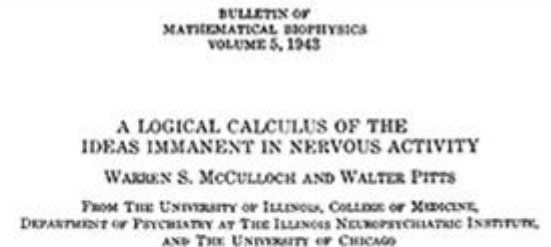
## 신경망 연구의 역사 요약

- 워런 맥컬록(Warren McCulloch), 월터 피츠(Walter Pitts) 최초의 신경망을 개발
- "A Logical calculus of ideas immanent in nervous activity"
- 뉴런의 작동 방식을 AND, NOT, OR 같은 디지털 신호로 재현할 수 있는 '맥컬록-피츠 뉴런(MCP뉴런)' 이라는 최초의 모델인 신경 시스템 모델 발표
- 워런 맥컬록
- 뇌의 뉴런 모델(Model of Neurons of Brain)
- 신경망은 on, off 상태의 '이진 스위칭' 소자가 복잡하게 연결된 네트워크라는 가설을 세움
- 뉴런이 서로 연결된 망에서도 일정한 수학적 계산과 학습이 가능하다는 사실 증명



논리학자 월터피츠  
(Walter Pitts)

신경외과의 워런 맥컬록  
(Warren McCulloch)



Because of the "all-or-none" character of nervous activity, neural events and the relations among them can be treated by means of propositional logic. It is found that the behavior of every net can be described in these terms, with the addition of more complicated logical means for nets containing circles; and that for any logical expression satisfying certain conditions, one can find a net behaving in the fashion it describes. It is shown that many particular choices among possible neurophysiological assumptions are equivalent, in the sense that for every net behaving under one assumption, there exists another net which behaves under the other and gives the same results, although perhaps not in the same time. Various applications of the calculus are discussed.



# 1. 데이터 분석 신경망 구조

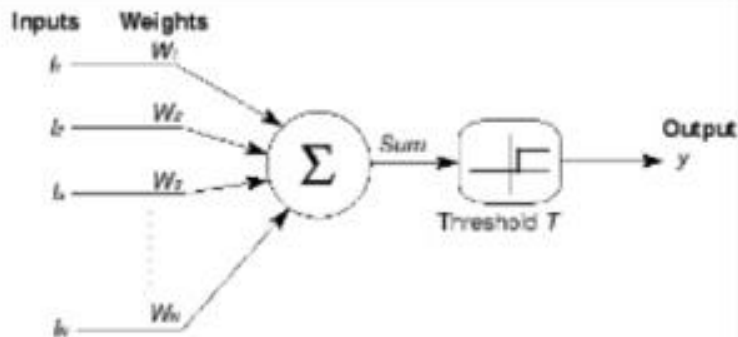
## 신경망 연구의 역사 요약

- 도날드 헵(Donald Hebb) 학습 알고리즘
- 헤비안 학습(Hebbian Learning) 하나의 규칙을 따라 뉴런 사이의 연결 강도가 수정되는 모델 시연 성공



### 헵의 학습 (Hebb's Learning)

- 생물학적 신경망에서 뉴런 사이에서 신호전달 시 반복적 또는 지속적으로 신호가 자극됨. 이 결과로 뉴런 A에서 뉴런 B로 가는 경로인 시냅스 연결이 강화된다.
- Hebb은 이것이 신경 레벨의 일종의 학습/기억의 과정이라고 보고 인공신경망에 가중치 개념으로 도입했다.
- 이것이 헵의 규칙 또는 헵의 학습이라 한다.



1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1949





# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- 앨런 튜링(Alan Turing)
- "Computing Machinery and Intelligence"
- 1937년 '보편기계(universal machine)' 개념 제안 후, 인공지능 이슈 역사상 최초로 제기
  - 튜링 기계
  - 기계학습
  - 유전 알고리즘
  - 강화 학습 소개

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1950

• 1950년 Alan Turing이 Computing Machinery and Intelligence  
논문에 발표



A. M. Turing (1950) Computing Machinery and Intelligence, *Mind* 49: 433-460.

### COMPUTING MACHINERY AND INTELLIGENCE

By A. M. Turing

#### I. The Imitation Game

I propose to consider the question, "Can machines think?" with definitions of the meaning of the words "machine" and "think" are to be found by examining how they are commonly used it is difficult to escape the conclusion that the meaning and the answer to the question, "Can machines think?" is to be sought in a statistical survey such as a Gallup poll. But this is absurd. Instead of attempting such a definition I shall replace the question by another, which is closely related to it and is expressed in relatively unambiguous words.

The new form of the problem can be described in terms of a game which we call the "imitation game." It is played with three people, a man (A), a woman (B), and an interrogator (C) who may be of either sex. The interrogator stays in a room apart from the other two. The object of the game for the interrogator is to determine which of the other two is the man and which is the woman. He knows them by labels X and Y, and at the end of the game he says either "X is A and Y is B" or "X is B and Y is A." The interrogator is allowed to put questions to A and B thus:

C: Will X please tell me the length of his or her hair?

Now suppose X is actually A, then A must answer. It is A's object in the game to try and cause C to make the wrong identification. His answer might therefore be:

"My hair is shingled, and the longest strands are about nine inches long."

In order that tones of voice may not help the interrogator the answers should be written, or better still, typed. The ideal arrangement is to have a teleprinter communicating



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- 마빈 민스키(Marvin Minsky), 딘 에드먼즈(Dean Edmonds)
- 최초의 신경망 컴퓨터 제작
- IBM
- 1952년에는 Arthur Samuel에 의해 인간과 체커스게임을 독립적으로 수행할 수 있는 프로그램을 설계하며, 기계의 상호작용이 독립적으로 이루어지게 된 상징적인 계기가 되었다. [

A. L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers," in IBM Journal of Research and Development, vol. 3, no. 3, pp. 210-229, July 1959, doi: 10.1147/rd.33.0210.

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1951

1952



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- 다트머쓰회의에서 AI 개념 정립
- Dartmouth College에서 마빈 민스키(Marvin Minsky), 클로드 새넌(Claude Shannon), 나다니엘 로체스터(Nathaniel Rochester) 등이 워크숍 조직
- 로체스터 IBM 연구팀
- 인공지능 체커 프로그램 선보임 >> 2016년 구글의 인공지능 알파고와 이세돌로 이어짐

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1955

1956

1956 Dartmouth AI Project



('1956)

인공지능 용어 등장

1955년 John McCarthy가 워크숍 아이디어 제안 "A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence"

John McCarthy ('1927 ~ '2011)

- 1956년 John McCarthy가 Dartmouth 대학 Conference에서 최초로 인공지능 용어를 사용

# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- 퍼셉트론(perceptron)은 인공신경망의 한 종류로서, 1957년에 코넬 항공 연구소(Cornell Aeronautical Lab)의 프랭크 로젠블라트 (Frank Rosenblatt)에 의해 고안되었다. 반복적으로 가중치를 학습 "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain" by Frank Rosenblatt (1958)

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

19C-20C 초반  
생체 신경망



이반 파블로프

1956년  
퍼셉트론



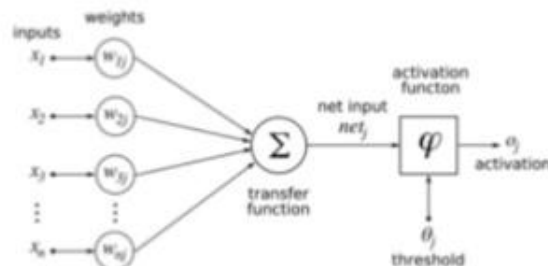
프랭크 로젠블라트



Frank Rosenblatt ('1928 ~ '1971)

- 1958년 인공신경망(ANN: Artificial Neural Network)을 실제로 구현한 **Perceptron** 이론을 발표
  - 맥컬록과 피츠의 인공신경망 이론에 헵(Hebb)의 생물학적 신경망의 학습효과를 인공신경망에 적용
  - 400여 개 빛 감지 센서와 512개의 증폭기, 40개 제어 부품으로 8개의 뉴런(Neuron, 신경세포)을 시뮬레이션 함.
- The perceptron: A probabilistic model for information storage and organization in the brain
  - 단층 퍼셉트론 (Single-Layer Perceptron)

McCulloch-Pitts TLU 이론 + Hebb의 가중치 이론으로 입력층과 출력층만으로 구성



- 활성화 함수 (Activation Function)
    - ✓ 일정크기(Threshold  $\theta$ )이상의 신호가 올 경우 값을 전달
    - ✓ 계단함수 대신 Sigmoid 함수나 tanh 함수 또는 ReLU 함수를 사용.
  - AND / OR Perception 예
    - ✓ AND :  $y = 0.6x_1 + 0.6x_2$  이고  $\theta = 1$
    - ✓ OR :  $y = 1.2x_1 + 1.2x_2$  이고  $\theta = 1$
- 적당한 가중치를 알아내는 학습이 필요함.



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- 존 맥카시
- LISP(List Processing) 개발
- 최초의 인공지능 시스템 Advice Taker 설계

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1958



John McCarthy ('1927 ~ '2011)

- 1956년 John McCarthy가 Dartmouth 대학 Conference에서 최초로 인공지능 용어를 사용
- 1958년 LISP 프로그래밍 언어 개발 (인공지능용 프로그래밍 언어)
- 인공지능 연구에 지대한 영향을 미침



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

Arthur Samuel ('1901 ~ '1990)



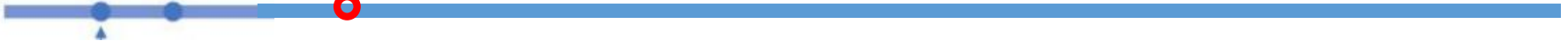
1959년에는 프로그래머 Arthur Samuel이 체스 자동 대국 프로그램을 제작하며 기계가 기계의 설계자에게 가르침을 받으며 종국에는 설계자의 실력보다 나아지는 것을 '머신러닝'이라 부르며 머신러닝의 개념이 도입되게 되었다.

1943년  
최초 인공



워런

1959





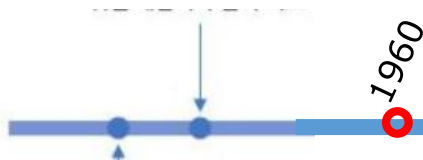
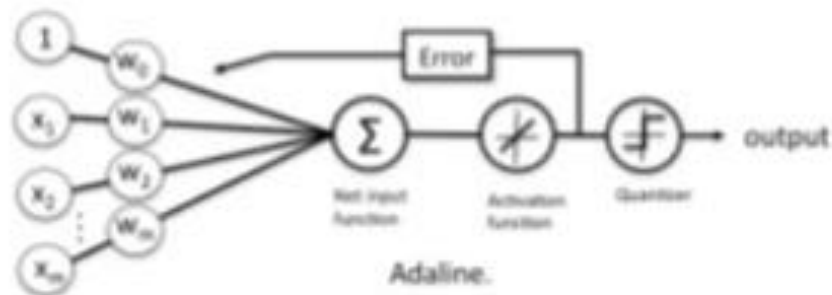
# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- LMS(Least Mean Squares) 알고리즘 인 adaline과 학습규칙 소개

### ? Widrow-Hoff Learning Rule

- 1960년 Bernard Widrow는 Perceptron과 유사한 ADALINE (Adaptive Linear Neuron) 단층 신경망 모델과 위드로-호프 학습규칙(Widrow-Hoff Learning Rule)을 공개
- Widrow-Hoff Learning Rule (Delta Rule)  
단층 신경망에서 적당한 가중치를 알아내기 위해 출력층의 출력 값의 오차에 비례해 가중치를 조절하는 것.
- 델타 규칙은 신경망의 오차가 충분히 줄어들 때 까지 학습 데이터를 반복해서 재 학습해야 한다.
- 델타 규칙은 후에 역전파의 기본이론이 된다.





# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

### • 다층 퍼셉트론 출현

- 1969년 Minsky와 Papert 의 『Perceptrons』는 퍼셉트론의 한계 지적. XOR 문제도 해결 못하는 선형 분류기에 불과함 입증 → 신경망 연구 퇴조. 인공지능 겨울에 일조
- 1969년 위치 호(Yu-Chi Ho)와 브라이손(Arthur E. Bryson)에 의해 Back-propagation Algorithm이 고안

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1968

1969





# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

60년대 말, 70년대 초에 인공지능 분야에서 발생한 일련의 비관적인 사건들이 일시적인 인공지능 침체기를 불러오다.



1<sup>st</sup> AI Winter

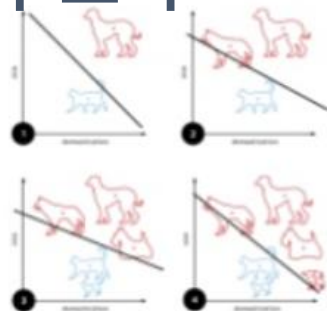
60년대 말 ~ 70년대

1943년  
최초 인공신경망(ANN)



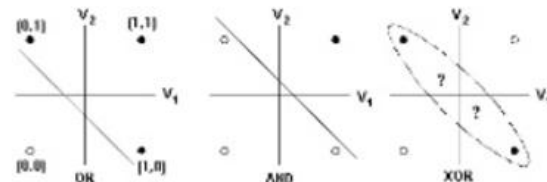
워렌 매컬러와 월터 피츠

1970



### Perceptron 이론의 한계 발견

- ✓ Perceptron은 학습이 진행될수록 선형 분리(linear boundary)를 업데이트하면서 학습
- ✓ 하지만, 간단한 XOR문제에는 적용할 수 없는 한계가 발견 → 인공지능의 외면



### 프로젝트 실패와 연구비 중단

#### 【 미국 】



- 57년 소련에서 최초로 인공위성 스푸트니크 발사 후 러시아어 자동 기계 번역 프로젝트 진행하였으나 66년 연구 중단 (약 2천만 달러 사용)
- 69년 이후 국방 연구 예산을 실무적인 분야에 투자함으로 인공지능 연구자금 확보 어려움

#### 【 영국 】



- 71년 James Lighthill 경의 영국 인공지능 현황 분석 보고서  
"AI work within the UK was unproductive"
- 인공지능 기술이 현실적인 대규모 문제를 풀기에는 역부족이다.  
→ 연구비 중단

Kunihiko Fukushima의 합성곱 신경망, Kunihiko Fukushima가 1970년대에 개발한 신경망 모델은 현대 합성곱 신경망(CNN)의 전신으로 볼 수 있는 "Neocognitron" 개발.



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- 하버드대의 폴 워보스(Paul Werbos)는 다층 퍼셉트론 환경에서 학습을 가능하게 하는 Back-propagation Algorithm으로 박사학위 논문을 씀

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1974

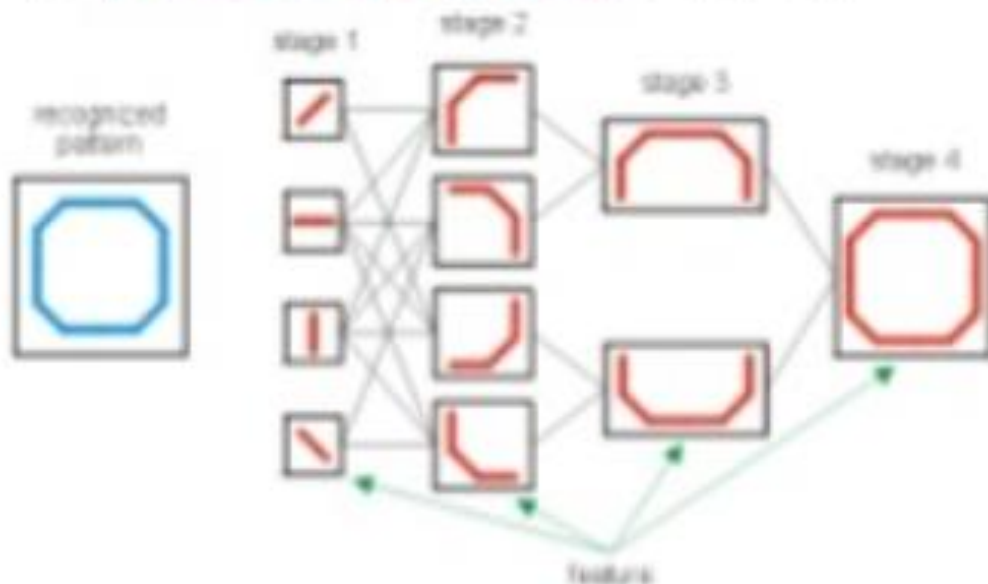


# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

### 네오코그니트론(neocognitron) 모델 ('1979)

- 후쿠시마 쿠니히코가 발표 (일본)
- 최초로 신경생리학이 융합된 인공신경망  
(뇌가 시각정보를 처리하는 구조를 기반으로 모델 작성)
- 현재 **Convolution 신경망(CNN)**과 매우 유사



1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1979



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- 홉필드(Hopfield) <Neural networks and physical systems with emergent computational abilities>  
홉필드 신경망(Hopfield network theory)
- 코호넨(Kohonen)의 자기조직 맵 이론(self-organized maps theory)
- 하버드대의 폴 워보스(Paul Werbos)는 다층 퍼셉트론 환경에서 학습을 가능하게 하는 Back-propagation Algorithm으로 박사학위 논문을 발표

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1982



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

### • 얀 르쿤(Yann LeCun)

- Backpropagation Multilayer Network(MLP)
- David Rumelhart, Geoffrey Hinton 및 Ronald Williams(1986)는 역전파를 개척하여 다층 신경망에서 가중치 학습을 실용적으로 만들었습니다. "Learning representations by back-propagating errors" by Rumelhart, Hinton, and Williams (1986) G. Hinton과 그의 동료들은 단층 Perceptron이 가지는 한계를 극복하기 위해 단층 Perceptron을 여러 겹 쌓은 다층 Perceptron 구조를 제시하게 된다.
- 루멜하트의 『Parallel Distributed Processing』은 은닉층을 가진 다층 퍼셉트론 제안 → 신경망 부활
- 다층 퍼셉트론(Multi-Layer Perceptron, 이하 MLP) 의 트레이닝 방법인 Back-Propagation (역전파, 이하 BP)
- 오류 역전파 알고리즘이라고 불리는 통계적 기법
- 동일 입력층에 대해 원하는 값이 출력되도록 개개의 시냅스의 가중치 값을 반복적으로 조정
- Palm, G. (1986). Warren McCulloch and Walter Pitts: A Logical Calculus of the Ideas Immanent in Nervous Activity. In: Palm, G., Aertsens, A. (eds) Brain Theory. Springer, Berlin, Heidelberg. [https://doi.org/10.1007/978-3-642-70911-1\\_14](https://doi.org/10.1007/978-3-642-70911-1_14)
- >> 2006, 2012 토론토대학교의 제프리 힌튼 교수팀에 의해 해결. - 딥러닝 "A fast learning algorithm for deep belief nets"

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1984

1986



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

Vanishing Gradient와 Overfitting

2nd AI Winter

80년대 후반 ~ 90년대 초

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

- 브룸헤드(Broomhead)와 로우(Lowe)
- 피드포워드 신경망 이론(layered feedforward networks theory)

얀 르쿤(Yann LeCun)

1989년 Yann LeCun의 역전파를 탑재한 합성곱 신경망  
필기 숫자에서 획기적 성능 향상(수표 자동인식 시스템)

1988

1990





# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

### Vanishing Gradient와 Overfitting

2nd AI Winter

80년대 후반 ~ 90년대 초

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

- 경사 소실(Gradient Vanishing) 문제를 처음으로 공식적으로 다룬 논문은 1991년 Sepp Hochreiter가 작성한 석사 논문입니다.
- 이 논문의 제목은 "Untersuchungen zu dynamischen neuronalen Netzen"(동적 신경망에 대한 조사)입니다. 독일어로 작성되었습니다.
- Hochreiter는 이 논문에서 깊은 신경망에서 역전파 알고리즘을 사용할 때 발생하는 경사 소실 문제를 처음으로 수학적으로 분석하고 설명했습니다.
- 그는 신경망의 층이 깊어질수록 오차의 그래디언트가 지수적으로 감소하여 입력층 근처의 가중치들이 효과적으로 학습되지 않는 현상을 보여주었습니다.
- 이 석사 논문은 당시에는 크게 주목받지 못했지만, 이후 Hochreiter와 그의 지도교수 Jürgen Schmidhuber가 1997년에 발표한 LSTM(Long Short-Term Memory) 논문의 기초가 되었습니다.

1991



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- "순환 네트워크의 경사 흐름: 장기 종속성 학습의 어려움"(Yoshua Bengio, Patrice Simard 및 Paolo Frasconi(1994))

1943년  
최초 인공신경망(ANN)



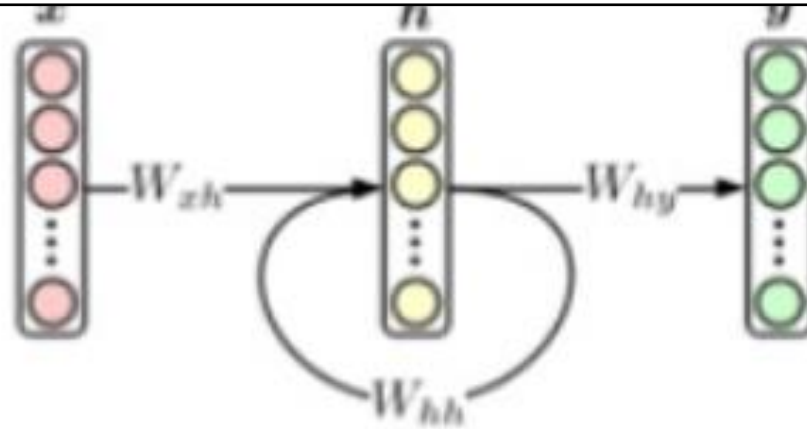
워렌 매컬러와 월터 피츠

1994

1997

- 1997년 S. Hochreiter과 J. Schmidhuber에 의해 개발된 장단기 기억 신경망(RNN의 일종, 순환신경망) 등의 매우 다양한 딥러닝 모델들이 소개

- Palm, G. (1986). Warren McCulloch and Walter Pitts: A Logical Calculus of the Ideas Immanent in Nervous Activity. In: Palm, G., Aertsen, A. (eds) Brain Theory. Springer, Berlin, Heidelberg. [https://doi.org/10.1007/978-3-642-70911-1\\_14](https://doi.org/10.1007/978-3-642-70911-1_14)
- Fukushima, K. A feature extractor for curvilinear patterns: a design suggested by the mammalian visual system. Kybernetik 7, 153-160 (1970). <https://doi.org/10.1007/BF00571695>
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735-1780.







# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- IBM Deep Blue 체스 인간에게 승리

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1997

【 Deep Blue 】



- 1996년 세계 체스 챔피언 Garry Kasparov 에게 도전했으나 1승 2무 3패로 패배
- 1997년 재도전 : 2승 3무 1패로 승리
- Champion급 체스 선수들은 약 10수 정도를 예측하는데 Deep Blue는 12수를 예측할 수 있었음.



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- Handwritten digit recognition with....



### 손글씨 문자 인식 연구 ('1998)

- CNN을 활용하여 높은 정밀도의 손글씨 문자 인식 (LeCun 팀)

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

1998

- Gradient-Based Learning Applied to Document Recognition -> LeNet-5



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- Google Multiple Language Translation

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

2000

딥러닝 빙하기 (2000~2005)

DNN 최적화 어려움  
엄청난 계산량  
머신러닝 알고리즘 발전



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- 2006, 2012 토론토대학교의 제프리 힌튼 교수팀에 의해 해결. - 딥러닝 용어 등장
- 제프리 힌튼은 "A fast learning algorithm for deep belief nets" 논문을 통해 가중치(weight)의 초깃값을 제대로 설정한다면 깊은 신경망을 통한 학습이 가능하다는 것을 밝혀냈습니다.
- Pre-training

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

2006

## "A Fast Learning Algorithm for Deep Belief Nets" by Geoffrey Hinton (2006)

### A fast learning algorithm for **deep belief nets**

Geoffrey E. Hinton and Simon Osindero  
Department of Computer Science University of Toronto  
10 Kings College Road  
Toronto, Canada M5S 3G4  
{hinton, osindero}@cs.toronto.edu

Yee-Whye Teh  
Department of Computer Science  
National University of Singapore  
3 Science Drive 3, Singapore, 117543  
tehyw@comp.nus.edu.sg

#### Abstract

We show how to use "complementary priors" to eliminate the explaining away effects that make inference difficult in densely-connected belief nets that have many hidden layers. Using complementary priors, we derive a fast, greedy algorithm that can learn deep, directed belief networks one layer at a time, provided the top two layers form an undirected associative memory. The fast, greedy algorithm is used to initialize a slower learning procedure that fine-tunes the weights using a contrastive version of the wake-sleep algorithm. After fine-tuning, a network with three hidden layers forms a very good generative model of the joint distribution of handwritten digit images and their labels. This generative model gives better digit classification than the best discriminative learning algorithms. The low-dimensional manifolds on which the digits lie are modelled by long ravines in the free-energy landscape of the top-level associative memory and it is easy to explore these ravines by using the directed connections to display what the associative memory has in mind.

#### 1 Introduction

Learning is difficult in densely-connected, directed belief nets that have many hidden layers because it is difficult to infer the conditional distribution of the hidden activities when given a data vector. Variational methods use simple approximations to the true conditional distribution, but the approximations may be poor, especially at the deepest hidden layer where the prior assumes independence. Also, variational learning still requires all of the parameters to be learned together and makes the learning time scale poorly as the number of parameters increases.

remaining hidden layers form a directed acyclic graph that converts the representations in the associative memory into observable variables such as the pixels of an image. This hybrid model has some attractive features:

1. There is a fast, greedy learning algorithm that can find a fairly good set of parameters quickly, even in deep networks with millions of parameters and many hidden layers.
2. The learning algorithm is unsupervised but can be applied to labeled data by learning a model that generates both the label and the data.
3. There is a fine-tuning algorithm that learns an excellent generative model which outperforms discriminative methods on the MNIST database of hand-written digits.
4. The generative model makes it easy to interpret the distributed representations in the deep hidden layers.
5. The inference required for forming a percept is both fast and accurate.
6. The learning algorithm is local: adjustments to a synapse strength depend only on the states of the pre-synaptic and post-synaptic neuron.
7. The communication is simple: neurons only need to communicate their stochastic binary states.

Section 2 introduces the idea of a "complementary" prior which exactly cancels the "explaining away" phenomenon that makes inference difficult in directed models. An example of a directed belief network with complementary priors is presented. Section 3 shows the equivalence between restricted Boltzmann machines and infinite directed networks with tied weights.

Section 4 introduces a fast, greedy learning algorithm for constructing multi-layer directed networks one layer at a time. Using a variational bound it shows that as each new layer is added, the overall generative model improves. The greedy algorithm bears some resemblance to boosting in its



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- ImageNet : ILSVRC
- 2009년 스탠포드 대학교의 페이페이 리(Fei-Fei Li)가 Li 1,400백만장 이상의 라벨이 붙은 이미지 데이터 베이스인 ImageNet 구축
- ImageNet을 시작으로 각종 딥러닝 성능 비교 대회에서 ImageNet의 경량 모델인 AlexNet, AlexNet을 최적화한 ZFNet, 그리고 VGGNet, GoogLeNet(InceptionNet), Skip Connection 개념으로 성능을 개선한 ResNet, 그리고 GoogLeNet을 개선한 Inception v2/v3/v4 모델이 등장하게 되었다.
- 2010년부터 ILSVRC(ImageNet Large Scale Visual Recognition Challenge)라는 이미지 인식(image recognition) 경진대회

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

2009

2010



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- Speech
- IBM 왓슨 Jeopardy 승리

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

2011

[ Watson ]



- 2011년 일반 상식 퀴즈 프로그램 Jeopardy 쇼에서 역대 최다 우승자(74회 연속)와 최고 상금 수상자 상대로 우승
- 4TB의 데이터를 오프라인으로 보유 (2억 페이지 정보)
- 다양한 비구조적인 많은 데이터를 종합적으로 처리.
- 수 많은 가설을 세우고 검증하는 일을 3초에 처리



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- 깊은 구조(deep architecture) 제프리 힌튼 팀의 AlexNet이 약 26%였던 오류율을 16%로 낮추는 압도적인 정확도로 우승을 차지하면서 딥 러닝은 부흥기를 맞게 됩니다.
- 2012년, 구글의 Jeff Dean과 Andrew Ng에 의해 인공신경망이 별다른 정보 혹은 Label(인공신경망이 가중치를 조절하게 될 참고 출력 값)없이도 고양이를 인식하게 하는 인공신경망을 설계하며, 딥러닝이라는 머신러닝의 인공신경망 분야 중에서도 더 세분화된 개념을 제시하게 된다.
- Dean, Jeffrey & Corrado, G.s & Monga, Rajat & Chen, Kai & Devin, Matthieu & Le, Quoc & Mao, Mark & Ranzato, Aurelio & Senior, Andrew & Tucker, Paul & Yang, Ke & Ng, Andrew. (2012). Large Scale Distributed Deep Networks. Advances in neural information processing systems.
- AlexNet으로 크리제프츠키, 수츠케버, 힌튼 팀이 84% 정확도로 ILSVRC에서 우승
- ImageNet Classification: CNN에 관한 Krizhevsky, Sutskever 및 Hinton(2012) 논문은 이미지 분류 작업에서 상당한 개선을 보여주었습니다.
- "ImageNet Classification with Deep Convolutional Neural Networks" by Krizhevsky, Sutskever, and Hinton (2012)

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

2012





# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- 강화학습을 위한 심층 Q-밍이 NIPS에 소개됨
- 심층 Q망을 이용하여 초창기 Atari 2600 게임을 인간 수준으로 플레이할 수 있게 됨

Google의 모회사인 Alphabet의 런던에 기반을 둔 연구 자회사인 Deepmind는 57개의 모든 Atari 2600 게임에서 표준 인간 벤치마크를 능가하는 Agent57을 만들었습니다. 이전에 우리는 회사가 2D 이미지에서 3D 모델을 렌더링할 수 있는 AI를 만드는 것을 보았습니다. 이번에 최근 논문에서 회사는 Atari 2600 게임(57개 모두)에서 인간을 이길 수 있는 능력을 가진 최초의 심층 강화 학습(RL) 에이전트인 Agent57을 만들었다고 밝혔습니다. 따라서 이름은 Agent57입니다.

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

2013

아타리 2600



개발사	아타리
종류	가정용 게임기
세대	2세대
출시일	1977년 9월 11일 1978년 1983년 5월 10일(아타리 2800) 1983년 9월
출시 가격	199 달러
판매량	3000만대 (1992년 기준) <sup>[1]</sup>
CPU	MOS 6507 @ 1.19 MHz
컨트롤러	조이스틱, 패들, 레이스 휠, 트랙볼, 키패드
사용 매체	롬 카트리지, 카세트 테이프
RAM	128바이트 RAM, 4kB ROM
온라인 서비스	게임라인(GameLine)
많이 팔린 게임	팩맨 700만대 (2006년 9월 1일 기준) <sup>[2][3]</sup>
이전 기종	퐁
후속 기종	아타리 5200





# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- 생성형 모델의 기반이 되는 AutoEncoder 모델 또한 Skip Connection의 개념이 적용되어 U-net이 개발 되었다. Liou, C.-Y., Cheng, W.-C., Liou, J.-W., & Liou, D.-R. (2014). Autoencoder for words. Neurocomputing, 139, 84–96.
- 이안 굿펠로우(Ian Goodfellow, 본명: 이안 J. 굿펠로우, Ian J. Goodfellow, 1987년 출생)가 주도한 몬트리올 대학교의 연구그룹이 GAN 소개
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... Bengio, Y. (2014). Generative adversarial nets. In Advances in neural information processing systems (pp. 2672–2680).



2014

# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

### • AlphaGo 이세돌에 도전

2014년 구글은 DeepMind를 인수하고, 2016년 AlphaGo는 이세돌 9단을 4:1로 물리친다.

#### 【 DeepMind 】



- 2011년 설립된 Startup으로 Google 인수 전 12명의 Deep Learning 전문가로 구성.
- 2014년 4억 달러 (약 4,000억원)에 구글에 인수
- 인수조건으로 구글에 인공지능윤리위원회 설치 요구
- 강화학습(Reinforcement Learning) 분야 최고의 회사
  - DQN(Deep Q-Network) 알고리즘 점수를 얻는 방법과 조이스틱을 조작하는 요령만 교육 받음.



<https://youtu.be/Q70ulPJW3Gk>

#### 【 AlphaGo 】



- 알고리즘 버전 13 : 2015년 10월 유럽 바둑 챔피언 Fan Hui 2단을 5:0으로 승리
- 알고리즘 버전 18 : 2016년 3월 이세돌 9단을 4:1로 승리
- 5개월 간 100만 번 이상의 대국. 인간 기준으로 매일 한 번씩 대국을 한다면 2,700년을 훈련한 것과 같다.
- 두 대국의 컴퓨터 규모는 CPU 1,202개, GPU 176개로 동일
- 2017년 5월 : 커제 9단을 3:0으로 승리

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

2016

# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

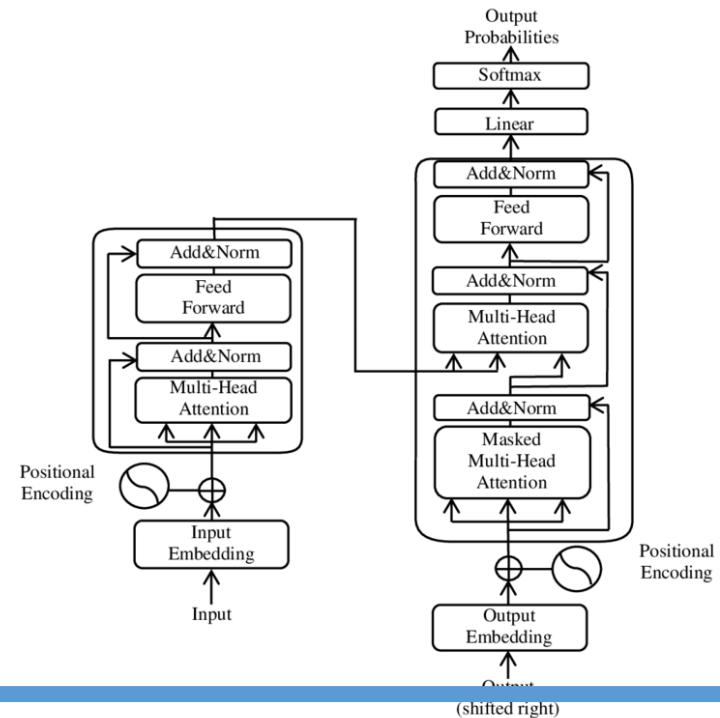
- Attention Is All You Need
- Vaswani 등은 Transformer 모델을 도입하여 NLP 작업에 혁명을 일으켰습니다.
- 트랜스포머(transformer)는 어텐션 메커니즘
- Vaswani, A., Shazeer, N.M., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., & Polosukhin, I. (2017). Attention is All you Need. Neural Information Processing Systems.

1943년  
최초 인공신경망(ANN)



워렌 매컬러와 월터 피츠

2017





# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약

- 2020년 6월 오픈AI가 GPT-3를 첫 출시 후
- 구글 2021년 5월 LLM '람다'(LaMDA)
- 네이버가 '하이퍼클로바'를 2021년 5
- 화웨이가 2021년 5월 '판구
- 바이두는 2021년 7월 '어니3'
- 카카오 코(Ko)-GPT 2021년 11
- LG그룹 연구조직 LG AI연구원이 '엑사원' 공개한 시점은 2021년 12월

• 오픈AI(Open AI)가 한국시간으로  
2022년 12월 1일 챗GPT(Chat GPT)  
출시

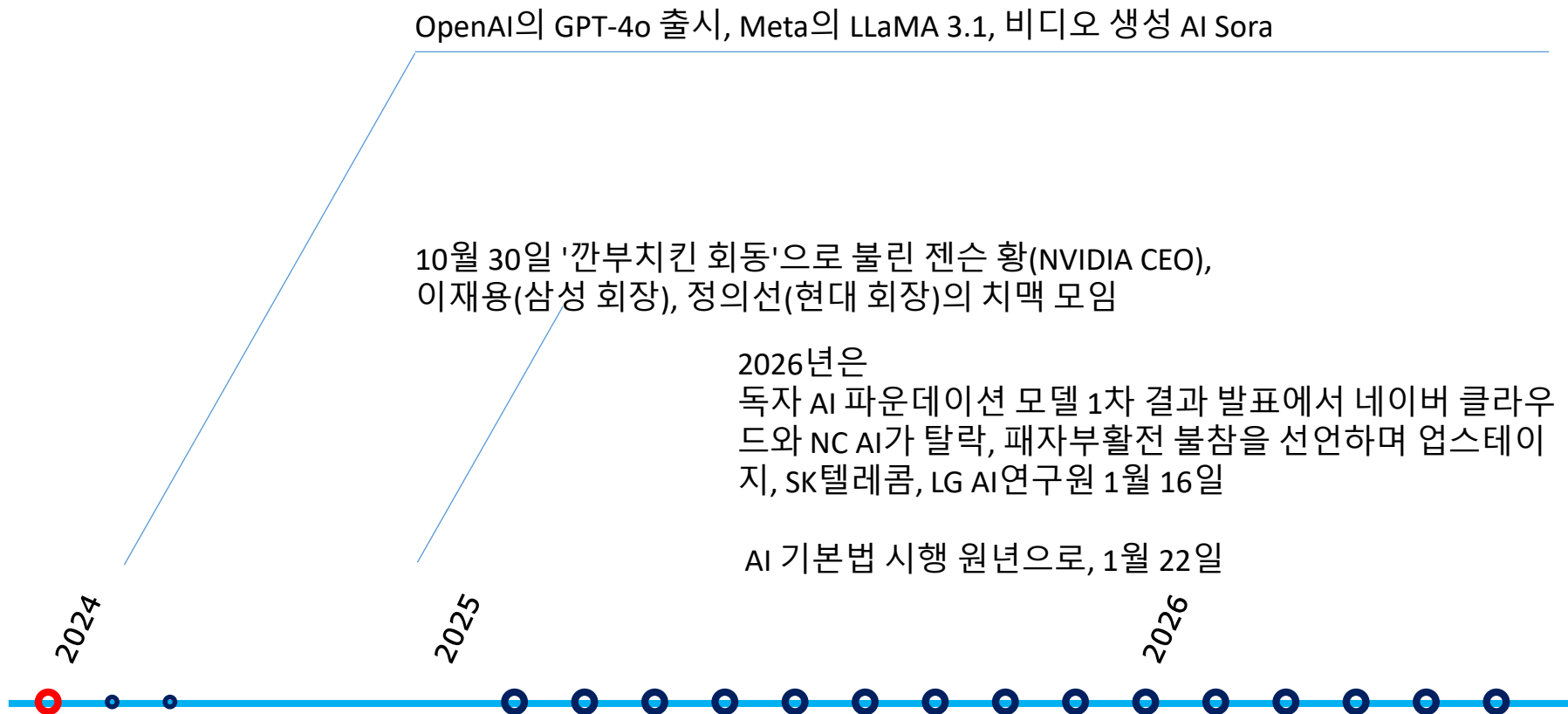


- 구글은 2월 '바드', 5월 '팜2'(PaLM) 그리고 2024년 2월, '제미나이(Gemini)
- 오픈AI 3월 GPT-4
- 바이두 '플라토3'(3월) ▲아마존 '타이탄'(4월) ▲메타 '라마2'(7월) ▲애플 '에이잭스'(7월) ▲텐센트 '훈위안'(9월)
- LG AI연구원의 '엑사원 2.0' 7월 ▲네이버 '하이퍼클로바X'(8월) ▲엔씨소프트 '바르코'(8월) ▲SKT '에이닷'(9월) ▲KT '믿음



# 1. 데이터 분석 신경망 구조

## 신경망 연구의 역사 요약





# 1. 데이터 분석 신경망 구조

## 신경망 모델 종류

- 딥러닝 (Deep Learning)
  - CNN: 이미지 인식 (불량 검사 자동화)
  - RNN: 시계열 패턴 학습
  - Transformer: 복잡한 패턴 인식
- AutoML (자동 머신러닝)
  - 알고리즘 자동 선택
  - 하이퍼파라미터 자동 최적화
- 시계열 예측
  - ARIMA: 전통적 시계열 분석
  - Prophet: 페이스북 개발 라이브러리



# 1. 데이터 분석 신경망 구조

## 퍼셉트론

- 퍼셉트론은 TLU(Threshold Logic Unit)이라는 형태의 뉴런을 기반으로 하며, 아래의 그림과 같이 입력과 출력이 어떤 숫자고 각각의 입력에 각각 고유한 가중치(  $W$  , weight)가 곱해진다.

$$z = w_1x_1 + w_2x_2 + \cdots + w_nx_n = \mathbf{W}^T \cdot \mathbf{X}$$

- 그런 다음 계산된 합  $z$  에 계단 함수(step function)를 적용하여 결과  $h$  를 출력한다.

$$\begin{aligned} h_w(\mathbf{X}) &= \text{step}(z) \\ &= \text{step}(\mathbf{W}^T \cdot \mathbf{X}) \end{aligned}$$



# 1. 데이터 분석 신경망 구조



## CNN (Convolutional Neural Network)

- 개념:

- 이미지 처리에 특화된 딥러닝
- 합성곱 층으로 특징 추출
- 제조 현장 비전 검사에 활용

- 제조업 적용 사례:

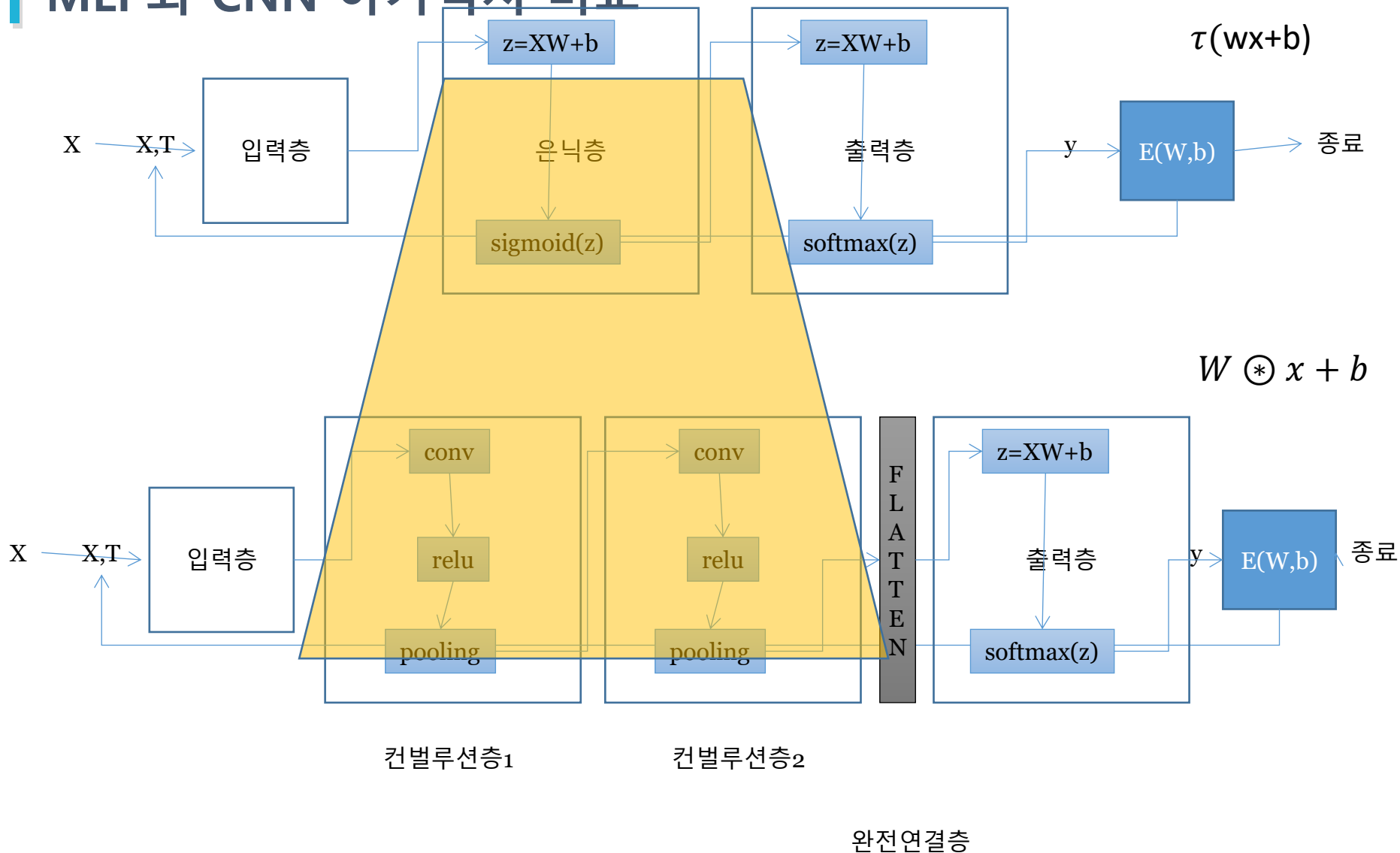
- 표면 결함 검사
- 용접 품질 검사
- 조립 정확도 확인
- 포장 불량 탐지





# 1. 데이터 분석 신경망 구조

## MLP와 CNN 아키텍처 비교





# 1. 데이터 분석 신경망 구조

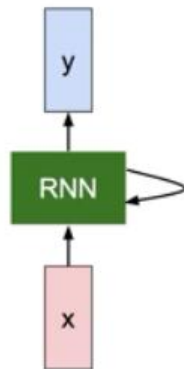
## RNN (Recurrent Neural Network)

### ● 개념:

- ● 시계열 데이터 처리에 특화
- ● 이전 정보를 기억하며 학습
- ● LSTM, GRU 등 고급 변형 존재

### ● 제조업 적용 사례:

- ● 설비 센서 데이터 패턴 학습
- ● 고장 예측 (예지 보전)
- ● 생산량 예측
- ● 에너지 소비 패턴 분석



$$h_t = f_W(h_{t-1}, x_t)$$

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

$$y_t = W_{hy}h_t$$

### ● 장점: 시간 순서 정보를 고려한 정확한 예측



# 1. 데이터 분석 신경망 구조

## Transformer 모델

- 개념:

- Attention 메커니즘 기반
- GPT, BERT의 핵심 구조
- 복잡한 패턴 인식에 강력

- 제조업 적용:

- 다변량 시계열 예측
- 복잡한 공정 최적화
- 품질 인자 간 상호작용 분석

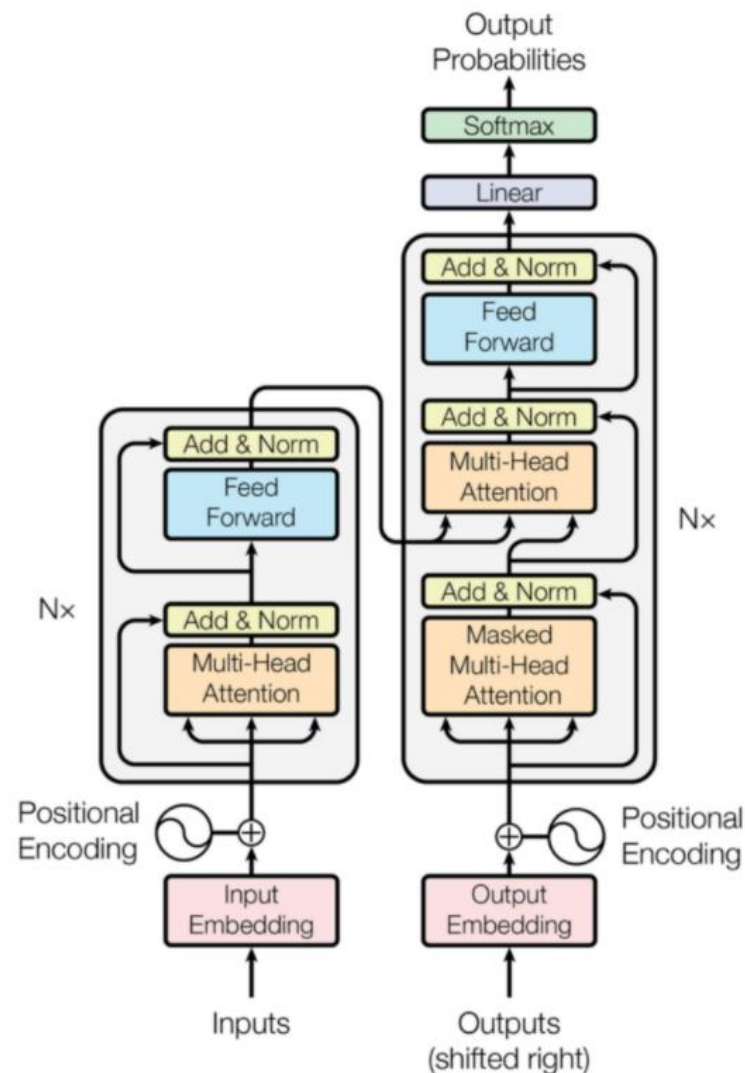


Figure 1: The Transformer - model architecture.



# 1. 데이터 분석 신경망 구조



## AutoML (Automated Machine Learning)

- 개념:

- 머신러닝의 전 과정을 자동화하는 기술

- 자동화 범위:

- 데이터 전처리 (결측치, 이상치, 인코딩)
- Feature Engineering (변수 생성 및 선택)
- 알고리즘 선택 (수십 가지 모델 자동 비교)
- 하이퍼파라미터 튜닝 (최적값 탐색)
- 모델 앙상블 (여러 모델 조합)

- 대표 라이브러리: H2O.ai, Auto-sklearn, TPOT, PyCaret



# 1. 데이터 분석 신경망 구조

## 시계열 예측 (Time Series Forecasting)

- 시간에 따라 변화하는 데이터의 미래 값을 예측
- 제조업 적용 분야:
  - 수요 예측 (생산 계획 수립)
  - 설비 고장 시점 예측
  - 재고 최적화
  - 에너지 수요 예측
- 주요 기법:
  - ARIMA: 전통적 통계 기법
  - Prophet: 페이스북 개발, 사용 편의성
  - LSTM: 딥러닝 기반, 복잡한 패턴



# 1. 데이터 분석 신경망 구조

## A/B 테스트: 모델 성능 검증

- 개념:
  - 두 가지 모델을 실제 환경에서 비교 실험
- 예시:
  - 모델 A: Random Forest (기존)
  - 모델 B: XGBoost (신규)
- 실험 설계:
  - 50% 트래픽 → 모델 A
  - 50% 트래픽 → 모델 B
  - 2주간 운영 후 성능 비교
- 평가 지표:
  - 정확도, 재현율뿐만 아니라
  - 비즈니스 지표 (불량률 감소, 비용 절감)



# 1. 데이터 분석 신경망 구조

## A/B 테스트: 모델 성능 검증

- 실험 설계:
- 가설:
  - "새로운 XGBoost 모델이 기존 모델보다 불량 탐지 성능이 우수하다"
- 그룹 분할:
  - A그룹: 생산 라인 1-5 (기존 Random Forest)
  - B그룹: 생산 라인 6-10 (신규 XGBoost)



# 1. 데이터 분석 신경망 구조

## A/B 테스트: 모델 성능 검증

- 측정 지표:
  - 1. 정량 지표
    - 불량 탐지율
    - 오탐률
    - 응답 시간
  - 2. 비즈니스 지표
    - 실제 불량률 감소
    - 검사 시간 단축
    - 재작업 비용
- 통계 검정:
  - t-test로 유의성 확인
  - $p\text{-value} < 0.05$ 면 채택



## 2. 제조 불량 예측에 적용

### | 문제 정의

- 목표: 공정 조건/운영 정보로 불량 발생 확률  $P(\text{불량}=1)$  예측
- 타입: 이진 분류(Binary Classification)
- 출력: 0~1 확률 + 임계값(예: 0.5)으로 분류



## 2. 제조 불량 예측에 적용

### 데이터 파이프라인(전처리) 구조

- 입력 피쳐 구성

- 수치형(5개): 온도, 압력, 속도, 진동, 설비연식
- 범주형(3개): 원료등급(A/B/C), 교대조(주간/야간), 요일(월~일)

- 전처리

- 결측치 처리: 현재 결측 없음(추후 대비 로직 포함)
- 이상치/센서값 점검: 진동 음수 등 비현실 값 처리 검토(클리핑/필터링)
- 스케일링(수치형): StandardScaler 또는 MinMaxScaler
- 범주형 인코딩
  - 간단: One-hot encoding (권장, 현재 범주 수 적음)
  - 대안: Embedding(범주 수가 커질 경우)

- 데이터 분할

- Train / Validation / Test (예: 70/15/15)
- 클래스 불균형 있으면: class\_weight 또는 resampling 고려



## 2. 제조 불량 예측에 적용

### 신경망(MLP) 모델 구조 아웃라인

- 기본 구조 (Baseline MLP)

- Input layer

- 입력 차원 = (수치 5) + (원료등급 3 one-hot) + (교대 2 one-hot) + (요일 7 one-hot)
- 합계 예:  $5 + 3 + 2 + 7 = 17$ 차원

- Hidden layers

- Dense(64) → ReLU → Dropout(0.2)
- Dense(32) → ReLU → Dropout(0.2)
- Dense(16) → ReLU

- Output layer

- Dense(1) → Sigmoid (불량 확률 출력)

- 정규화/일반화 옵션

- BatchNorm(각 Dense 뒤)
- L2 regularization(가중치 규제)
- EarlyStopping(검증 손실 기준 조기 종료)



## 2. 제조 불량 예측에 적용

### | 학습(Training) 설정

- Loss: Binary Cross Entropy
- Optimizer: Adam (lr=1e-3 시작 권장)
- Metrics
  - Accuracy (참고용)
  - Precision / Recall / F1 (불량 예측에서는 중요)
  - ROC-AUC
  - 불량(양성) 중심이면 PR-AUC도 추천
- Threshold 튜닝
  - 운영 목적에 따라 임계값을 0.5 고정하지 않고 조정
  - 예: 불량 놓치면 큰 손실 → Recall 우선(임계값 낮춤)



## 2. 제조 불량 예측에 적용

### | 평가 및 해석(현업 적용을 위한 단계)

- 평가 리포트
  - Confusion Matrix
  - ROC Curve / PR Curve
  - 그룹별 성능(예: 원료등급 C에서 성능, 야간에서 성능)
- “현장 적용 가능한” 해석
  - Feature importance:
    - 간단: Permutation importance
    - 권장: SHAP (온도/압력/설비연식 영향 정량화)
  - 위험 구간 룰 추출
    - 예: 온도 > 90 또는 압력 > 160 조합에서 불량률 급증



## 2. 제조 불량 예측에 적용

### 모델 고도화 옵션(필요 시)

- 범주형 Embedding + 수치형 MLP 결합
  - 원료등급/교대/요일을 embedding으로 넣고 concat 후 MLP
- Calibration(확률 보정)
  - Platt scaling / Isotonic regression으로 확률 신뢰도 개선
- 비용 민감 학습
  - 불량 놓침 비용이 크면 가중치 조정



## 2. 제조 불량 예측에 적용

### 운영 배포 구조

- 입력(센서/공정 로그) → 전처리(스케일러/인코더) → 모델 → 불량 확률 출력
- 알람/제어 연동:
  - 확률 > 임계값 → 경고
  - 특정 조합(고온+고압+노후) → 강경고



# 3. 데이터 정제 및 이상치 제거



## 데이터 정제(Data Purification)란?

데이터 정제는 모델 학습 전에 데이터를 정확하고 일관되며 신뢰 가능한 상태로 만드는 과정.

### 목적:

- 노이즈 제거
- 센서 오류 수정
- 이상치 탐지
- 결측값 처리
- 모델 성능 향상
- 잘못된 의사결정 방지





# 3. 데이터 정제 및 이상치 제거



## 데이터 정제의 주요 단계

### 1. 결측값 처리

- 방법
  - 평균/중앙값 대체
  - KNN 기반 보간
  - 회귀 기반 예측 대체
  - 삭제(비율 낮을 경우)
- AI 기반 접근
  - Autoencoder 복원
  - MICE (Multiple Imputation)
  - 시계열 예측 모델 활용



# 3. 데이터 정제 및 이상치 제거

## 데이터 정제의 주요 단계

### 2. 이상치(Outlier) 탐지

- 이상치는 다음과 같은 경우 발생
  - 센서 오류
  - 비정상 공정 조건
  - 데이터 입력 오류
- 이상치 탐지 방법
  - (1) 통계 기반 방법
    - Z-score ( $\pm 3\sigma$ )
    - IQR 방식 ( $Q1 - 1.5IQR, Q3 + 1.5IQR$ )
    - Grubbs test
    - 장점: 간단
    - 단점: 복잡한 패턴 탐지 어려움



# 3. 데이터 정제 및 이상치 제거

## 데이터 정제의 주요 단계

### 2. 이상치(Outlier) 탐지

- 이상치 탐지 방법

- (2) 머신러닝 기반 방법

- ✓ Isolation Forest

- 랜덤 분할 기반

- 고차원 데이터에 적합

- 제조 데이터에 매우 많이 사용됨

- ✓ One-Class SVM

- 정상 데이터 학습 후 이상 탐지

- ✓ Local Outlier Factor (LOF)

- 주변 밀도 기반 이상치 판별



# 3. 데이터 정제 및 이상치 제거



## 데이터 정제의 주요 단계

### 2. 이상치(Outlier) 탐지

- 이상치 탐지 방법

- (3) 딥러닝 기반 방법

- ✓ Autoencoder
  - 입력 → 압축 → 복원
  - 복원 오차가 크면 이상치
- ✓ Variational Autoencoder (VAE)
  - 확률 분포 기반 이상 탐지
- ✓ LSTM (시계열 데이터)
  - 시간 흐름 기반 이상 탐지



## 3. 데이터 정제 및 이상치 제거



### 데이터 정제의 주요 단계

#### 3. 이상치 처리 방법

- 이상치를 발견한 후 선택지
  - 제거 (Drop)
  - 수정 (Winsorizing / Clipping)
  - 별도 라벨링 후 분석
  - 모델 학습 시 가중치 낮춤



### 3. 데이터 정제 및 이상치 제거

#### 전체 프로세스 구조

Raw Data



결측값 처리



이상치 탐지 (통계 + ML + DL)



정규화 / 스케일링



모델 학습



성능 평가



### 3. 데이터 정제 및 이상치 제거

#### 실습

실습 2-1: 결측치 확인

실습 2-2: 이상치 탐지

실습 2-3: 이상치 제거 실습

실습 2-4: 중복 데이터 확인

# 『Day 2』

## AI 활용 데이터 변환 및 분석

- Part 1. 1일차 복습 및 2일차 개요
- Part 2. 현장 데이터 분석 기초
- Part 3. AI 학습 데이터 변환







# 1. AI 학습 데이터 변환

## 정규화 (Normalization)

- "모든 숫자를 같은 기준으로 맞추기"
- 왜 필요할까?
  - 문제 상황:
  - 온도: 75도
  - 진동: 0.5
- AI가 보기에?
  - → 온도(75)가 진동(0.5)보다 150배나 중요해 보임!
  - → 하지만 실제로는 둘 다 똑같이 중요할 수 있음
- 비유:
  - 키를 cm로 재면 170
  - 몸무게를 kg으로 재면 70
  - 하지만 키가 더 중요한 건 아님.



# 1. AI 학습 데이터 변환

## 정규화 공식

$$\text{정규화 값} = (\text{현재 값} - \text{최솟값}) / (\text{최댓값} - \text{최솟값})$$

예시: 온도 85도를 정규화하면?

온도 범위: 60도 ~ 90도

$$\begin{aligned} 85\text{도 정규화} &= (85 - 60) / (90 - 60) \\ &= 25 / 30 \\ &= 0.83 \end{aligned}$$

의미: 85도는 전체 범위에서 83% 지점에 있음



# 1. AI 학습 데이터 변환-정규화

언제 정규화가 특히 중요한가?

모델	필요성
신경망	매우 중요
KNN	매우 중요
SVM	중요
선형회귀	권장
트리모델(RandomForest)	필요 없음



# 1. AI 학습 데이터 변환

## 정규화의 효과

**Before (정규화 전):**

- 온도: 75도 ← AI가 이게 더 중요하다고 착각
- 진동: 0.5 ← 너무 작아서 무시됨

**After (정규화 후):**

- 온도: 0.5 ← 둘 다 0~1 범위
- 진동: 0.5 ← 공정하게 비교 가능!

**실제 효과:**

- AI 학습이 안정적
- 학습 속도 빨라짐
- 모든 변수가 공정하게 영향을 미침



# 1. AI 학습 데이터 변환-데이터 분할

## 왜 데이터를 나누는가?

목적: 모델의 "일반화 성능" 평가

모델을 학습 데이터로만 평가하면,

- 과적합(Overfitting) 가능
- 훈련 데이터만 잘 맞는 모델 생성
- 실제 현장 적용 시 성능 저하

따라서,

- Train → 학습
- Validation → 처음 보는 데이터로 성능 평가



# 1. AI 학습 데이터 변환-데이터 분할

## 왜 데이터를 나누는가?

**나쁜 예: 모든 데이터로 학습**

- 시험공부: 기출문제 100문제 다 외움
- 실제시험: 똑같은 100문제 나옴
- 결과: 100점!
- 하지만... 새로운 문제는 못 풀음 😞

**좋은 예: 데이터 분할**

- 학습 데이터 80개: 이걸로 공부함
- 검증 데이터 20개: 처음 보는 문제로 실력 테스트
- → 진짜 실력을 알 수 있음! ✅



# 1. AI 학습 데이터 변환-데이터 분할

## 왜 80:20 비율인가?

### ✓ 이유 1: 충분한 학습 데이터 확보

- 모델이 패턴을 충분히 학습하려면 많은 데이터 필요

### ✓ 이유 2: 검증 데이터도 충분히 확보

- 20%면 통계적으로 평가 가능
- 너무 적으면 평가 불안정



# 1. AI 학습 데이터 변환-데이터 분할

## 왜 80:20 비율인가?

너무 많이 나누면 (50:50)?

- 학습 데이터가 부족 → AI가 제대로 못 배움

너무 적게 나누면 (95:5)?

- 검증 데이터가 부족 → 실력 평가가 부정확

80:20이 딱 좋음!

- 학습: 충분히 배울 수 있음
- 검증: 신뢰할 수 있는 평가





# 1. AI 학습 데이터 변환-데이터 분할

## 일반적인 분할 기준

데이터 규모	권장 비율
데이터 적음 (<1000)	70:30
중간 규모 (~2000)	80:20
매우 많음	90:10
딥러닝 대규모	Train/Val/Test 70/15/15



# 1. AI 학습 데이터 변환-데이터 분할

## 인코딩 (Encoding)

"글자를 숫자로 바꾸기"

AI는 숫자만 이해해요!

문제:

AI: "원료등급 A가 뭐야? 먹는 건가?"

사람: "아니, A가 제일 좋은 등급이야"

AI: "그게 몇이야? 숫자로 말해줘!"



# 1. AI 학습 데이터 변환-데이터 분할

## 변환 방법

### 1) Label Encoding (순서가 있을 때)

원료등급: A → 1

B → 2

C → 3

의미: A가 제일 좋고, C가 제일 나쁨 (순서 있음)

### 2) Binary Encoding (2가지만 있을 때)

교대조: 주간 → 0

야간 → 1

의미: 순서 없음, 그냥 다름

### 3) Ordinal Encoding (요일처럼 순서가 있을 때)

요일: 월 → 1

화 → 2

수 → 3

목 → 4

금 → 5



# 1. AI 학습 데이터 변환-데이터 분할

## 주의사항

**잘못된 예:**

색깔: 빨강 → 1

파랑 → 2

노랑 → 3

AI 생각: "파랑이 빨강보다 2배 크네?"

→ 틀렸음! 순서가 없는데 순서를 만듦

**올바른 예:**

색깔: 빨강 → [1, 0, 0]

파랑 → [0, 1, 0]

노랑 → [0, 0, 1]

→ One-Hot Encoding (순서 없을 때)



# 1. AI 학습 데이터 변환

## 실습

- 실습 3-1: 범주형 변수 인코딩
- 실습 3-2: 정규화 실습
- 실습 3-3: 데이터 분할
- 실습 3-4: 변환 검증



## 2. 불량률 예측 및 원인 분석



### | 불량률이란?

- 불량률 = (불량 개수 ÷ 전체 개수) × 100

- 예시:

100개 만들었는데 20개가 불량

→ 불량률 =  $20/100 = 20\%$



## 2. 불량률 예측 및 원인 분석



### 불량 예측은 어떻게?

- 1단계: 과거 데이터 보기
  - 온도 85도 이상일 때
  - → 100번 중 70번 불량 발생
  - → 불량률 70%
- 2단계: 새로운 상황에 적용
  - 오늘 온도가 87도네?
  - → 과거 데이터상 70% 확률로 불량 예상
  - → 주의 필요!



## 2. 불량률 예측 및 원인 분석

### AI의 예측 방법

● 로지스틱 회귀 = 확률 계산기

입력: 온도 85도, 압력 155, 속도 105



AI가 계산 중...



출력: 불량 확률 65%

의미:

- 65% 확률로 불량

- 35% 확률로 정상

→ 위험한 상황!

**불량 확률 =  $1 / (1 + e^{(-\text{점수})})$**

점수가 크면 → 확률 높음

점수가 작으면 → 확률 낮음





## 2. 불량률 예측 및 원인 분석

### 원인 분석 이론

#### 1단계: 단일 요인 분석

- "범인을 하나씩 조사하기"

#### ● 방법 1: 상관계수

- 온도와 불량률의 관계: 0.68
- → 온도가 높으면 불량률도 많음

- 압력과 불량률의 관계: 0.09
- → 압력은 별 영향 없음

#### ● 쉬운 해석:

- 1에 가까우면: 강한 관계 (범인일 가능성 높음)
- 0에 가까우면: 관계 없음 (무죄)



## 2. 불량률 예측 및 원인 분석








### | 원인 분석 이론

- **방법 2: 평균 비교**
  - 불량 제품의 평균 온도: 82도
  - 정상 제품의 평균 온도: 71도
  - 차이: 11도
- → 확실히 다름! 온도가 범인!



## 2. 불량을 예측 및 원인 분석

### 원인 분석 이론

- 2단계: 구간별 분석
  - "어느 구간이 위험한지 찾기"
  - 온도 구간별 불량률:
    - 60~70도: 15%  안전
    - 70~80도: 25%  주의
    - 80~90도: 67%  위험!
    - 90도 이상: 75%   매우 위험!
  - 결론: 80도가 경계선
- 실무 적용:
  - → 목표: 70~75도 유지
  - → 알람: 80도 도달 시
  - → 긴급: 85도 이상 시 조치



## 2. 불량률 예측 및 원인 분석

### 원인 분석 이론

- 3단계: 복합 원인 분석
  - "범인이 여럿일 수도!"
- 교호작용 (Interaction)이란?
  - 비유: 감기약 + 술
  - 감기약만: 괜찮음
  - 술만: 괜찮음 (적당히)
  - 감기약 + 술: 위험! ⚠
- → 1+1이 2가 아니라 10이 됨!



## 2. 불량률 예측 및 원인 분석

### 제조 현장 예시

- 단일 조건:

온도 85도: 불량률 60%

C등급 원료: 불량률 40%

- 복합 조건:

온도 85도 + C등급 원료: 불량률 85%!

왜?  $1+1=2$ 가 아니라

온도  $\uparrow$  + 낮은 등급 = 상승효과!



## 2. 불량률 예측 및 원인 분석

### 제조 현장 예시

- 더 복잡한 예:

조건 조합	불량률
온도 85도	60%
설비 8년	45%
야간 교대	40%
온도 85 + 설비 8년	75% (복합 효과)
온도 85 + 야간	70% (복합 효과)
셋 다 동시	90%! (최악)



## 2. 불량률 예측 및 원인 분석

### 실무 적용 방법

- Step 1: 데이터 준비
  - 1. 정규화: 모든 숫자를 0~1로 맞춤
  - 2. 인코딩: 글자를 숫자로 변환
  - 3. 분할: 학습용 80%, 검증용 20%
- Step 2: 원인 찾기
  - 1. 단일 원인: 상관계수로 주범 찾기
  - 2. 구간 분석: 위험 구간 파악
  - 3. 복합 원인: 위험한 조합 찾기
- Step 3: 예측하기
  - 1. AI 학습: 과거 패턴 학습
  - 2. 확률 예측: 새 상황의 불량 확률
  - 3. 의사결정: 임계값 넘으면 조치



## 2. 불량률 예측 및 원인 분석

### 실습

- 실습 4-1: 주요 변수 분석
- 실습 4-2: 조건별 불량률 분석
- 실습 4-3: 복합 원인 분석
- 실습 4-4: 개선 방안 도출
- 실습 4-5: 예측 모델 해석





### 3. 종합 실습 및 케이스 스터디

#### 학습 목표

- 배운 모든 내용을 통합하여 실무 문제를 해결한다
- 분석 결과를 보고서 형태로 작성한다



## 3. 종합 실습 및 케이스 스터디

### 실습 시나리오

#### [상황]

공장장이 불량률이 너무 높다며 원인 분석을 요청했습니다.  
당신은 데이터 분석 담당자로서 다음 작업을 수행해야 합니다.

#### [요구사항]

1. 데이터 품질 확인 (결측치, 이상치, 중복)
2. 필요시 데이터 정제
3. 주요 불량 원인 3가지 도출
4. 각 원인의 영향도 분석 (데이터와 통계로 증명)
5. 개선 방안 제시 (구체적인 조치와 예상 효과)
6. 최종 보고서 형태로 정리

보고서는 다음 형식으로 작성해줘:

- 요약 (3줄 이내)
- 데이터 분석 결과 (표와 수치 포함)
- 주요 발견사항 (불량 원인 TOP 3)
- 개선 권고안 (우선순위별)
- 결론



# 3. 종합 실습 및 케이스 스터디



## | 주요 활동

개인별로 프롬프트 실행  
AI와 대화하며 분석 수행  
결과를 정리하여 발표 준비



### 3. 종합 실습 및 케이스 스터디

#### 예상 소요 시간

분석 실행: 20분

보고서 정리: 15분

확인: 5분

- 프롬프트 작성의 적절성
- 분석 깊이와 논리성
- 실무 적용 가능성
- 개선 방안의 구체성

#### 토론:

- "각자의 분석 결과에서 공통점과 차이점은?"
- "왜 같은 데이터로도 다른 결론이 나올 수 있을까?"



## 4. 심화 실습: 가상 시나리오 예측

### 예상 소요 시간

다음 가상 시나리오에서 불량률을 예측해 보자.

새로운 생산 계획:

- 온도: 82도
- 압력: 155 bar
- 속도: 105 rpm
- 진동: 0.6
- 설비연식: 6년
- 원료등급: B
- 교대조: 야간
- 요일: 목요일

현재 데이터를 바탕으로:

1. 이 조건에서 불량률이 발생할 확률을 추정해줘
2. 가장 유사한 과거 데이터 5개를 찾아 그 결과를 보여줘
3. 위험 요소가 있다면 어떤 변수를 조정하면 좋을지 제안해줘
4. 조정 후 예상 불량률을 계산해줘

# THANK YOU.

앞으로의 엔지니어는 단순한 '코더'나 '기계 조작자'가 아니라 뇌-기계 인터페이스를 통해 지식과 능력을 즉각 확장하는 존재(뉴로-인터페이스: Neuro Interface)가 될 수 있습니다.

- 🎯 목표 달성을 위한 여정이 시작됩니다.
- 🌟 궁금한 점이 있으시면 언제든지 문의해주세요!
- 🚀 함께 코더와 프롬프트 전문가로 성장해 나갑시다!