

# 생성형 AI와 대규모 언어 모델(LLM) 이해

Day 4 - Part 1

---

## 이번 파트의 학습 목표

---

이번 시간을 통해 우리는 다음 개념들을 배우고, 생성형 AI의 시대를 이해하는 것을 목표로 합니다.

- 생성형 AI(Generative AI)의 개념을 정의하고, 전통적인 AI와의 근본적인 차이점을 설명할 수 있습니다.
- 대규모 언어 모델(LLM)이 무엇인지 이해하고, GPT, BERT, T5와 같은 주요 모델들의 구조적 차이와 용도를 비교할 수 있습니다.
- LLM의 발전을 이끈 핵심 기술인 Transformer와 Self-Attention의 원리를 이해합니다.
- “크면 클수록 좋다”는 스케일링 법칙(Scaling Laws)이 LLM 발전에 어떤 의미를 갖는지 설명할 수 있습니다.
- LLM이 현실 세계의 다양한 산업(콘텐츠 제작, 챗봇, 코딩 등)에 어떻게 응용되고 있는지 사례를 통해 이해합니다.

# 생성형 AI(Generative AI)란?

## "패턴 인식"을 넘어 "패턴 창조"로

생성형 AI는 학습한 데이터의 패턴과 구조를 바탕으로, 세상에 없던 **새로운 콘텐츠(텍스트, 이미지, 코드, 음악 등)**를 만들어내는 인공지능 기술입니다.

- **학습 단계**: 방대한 양의 데이터(예: 웹 텍스트, 이미지 모음)를 학습하며 그 안에 내재된 통계적 패턴과 규칙을 익힙니다.
- **생성 단계**: 사용자로부터 받은 입력(**프롬프트, Prompt**)을 조건으로, 학습한 패턴에 따라 가장 그럴듯한(probable) 결과물을 새롭게 생성합니다.
- **핵심 차별점**: 기존 AI가 주어진 데이터를 **분류(Classification)**하거나 **예측(Prediction)**하는 '판별' 모델에 가까웠다면, 생성형 AI는 새로운 데이터를 '생성'하는 '**생성**' 모델이라는 점에서 근본적인 차이가 있습니다.

### 작동 방식 요약

"수많은 고양이 사진을 본 AI에게 '귀여운 고양이를 그려줘'라고 했을 때, 사진들의 공통적인 특징(뾰족한 귀, 동그란 눈, 수염 등)을 조합하여 세상에 단 하나뿐인 새로운 고양이 이미지를 창조해내는 것"

# 전통적 AI vs. 생성형 AI 비교

## 무엇을 할 수 있는가?: 판단 vs. 창조

전통적 AI와 생성형 AI는 목표와 작동 방식에서 명확히 구분됩니다.

### 전통적 AI (판별형 AI)

- **목표:** 데이터 분류, 예측, 추천
- **작동 원리:** 정해진 규칙 또는 데이터 패턴 분석
- **산출물:** 예측 값, 분류 레이블 (e.g., 스팸/정상)
- **창의성:** 없음 (정해진 범위 내에서 동작)
- **데이터:** 주로 정형화된 데이터에 강점
- **예시:** 스팸 메일 필터, 상품 추천 시스템

### 생성형 AI

- **목표:** 새로운 콘텐츠 생성
- **작동 원리:** 학습된 데이터의 확률 분포 기반 생성
- **산출물:** 텍스트, 이미지, 코드 등 새로운 데이터
- **창의성:** 높음 (새로운 아이디어/결과물 생성)
- **데이터:** 대규모 비정형 데이터 기반 학습
- **예시:** ChatGPT, DALL-E, GitHub Copilot

### 핵심 비유

전통적 AI가 "이 사진에 고양이가 있는지 없는지 판단하는 심판"이라면, 생성형 AI는 "상상 속의 고양이'를 새롭게 그려내는 화가"입니다.

# 대규모 언어 모델(LLM)이란?

## 언어를 이해하고 생성하는 거대한 뇌

대규모 언어 모델(Large Language Model, LLM)은 수천억 개 이상의 매개변수(Parameter)를 가진 거대한 인공 신경망으로, 방대한 양의 텍스트 데이터를 학습하여 인간의 언어를 이해하고 생성하는 능력을 갖춘 모델입니다.

- **핵심 기술:** 대부분의 현대 LLM은 **Transformer** 아키텍처를 기반으로 합니다.
- **학습 데이터:** 인터넷의 웹 페이지, 책, 뉴스 기사 등 상상할 수 있는 거의 모든 텍스트를 학습합니다.
- **능력:** 문맥 이해, 문장 생성, 번역, 요약, 질의응답 등 다재다능한 언어 과제를 수행할 수 있습니다.
- **대표 주자:** OpenAI의 **GPT** 시리즈, Google의 **BERT, T5**, Meta의 **LLaMA** 등이 있습니다.

### "대규모"의 의미

LLM의 '대규모'는 단순히 큰 것이 아니라, 모델의 크기(파라미터 수)가 일정 수준을 넘어서면서 기존 모델에서는 볼 수 없었던 **창발적 능력(Emergent Abilities)** - 예를 들어 추론, 복잡한 지시 이해 등 - 이 나타나기 시작했다는 점에서 중요합니다.

# 주요 LLM 아키텍처 비교: GPT vs BERT vs T5

## 목적에 따라 진화한 세 가지 형태

LLM은 Transformer 구조를 어떻게 활용하는지에 따라 크게 세 가지 계열로 나뉩니다.

계열	대표 모델	Transformer 구조	핵심 특징	주요 용도
디코더-Only	GPT, LLaMA, PaLM	Decoder	단방향(왼쪽→오른쪽) 문맥을 보고 다음 단어를 예측(Auto-regressive). 문장 생성에 특화.	텍스트 생성, 챗봇, 창작, 요약
인코더-Only	BERT, RoBERTa	Encoder	양방향 문맥을 모두 보고 빈칸(MASK)을 예측. 문장 전체의 의미 이해에 특화.	자연어 이해(NLU), 문장 분류, 감성 분석, 개체명 인식
인코더-디코더	T5, BART	Encoder + Decoder	입력 시퀀스를 다른 출력 시퀀스로 변환하는 Seq2Seq 구조. 모든 NLP 문제를 "Text-to-Text"로 접근.	번역, 요약, 문서 변환

# 핵심 혁신 ①: Transformer & Self-Attention

## LLM 시대를 연 결정적 기술

2017년 논문 "Attention Is All You Need"에서 등장한 Transformer는 순차적으로 정보를 처리하던 기존 RNN/LSTM의 한계를 극복했습니다.

### RNN의 한계

- **순차 처리**: 문장이 길어지면 처리 속도가 느림 (병렬화 불가).
- **장기 의존성 문제**: 문장 앞부분의 정보가 뒤로 갈수록 희미해짐.

### Transformer의 혁신

- **병렬 처리**: 문장 전체를 한 번에 입력받아 모든 단어 간 관계를 동시에 계산.
- **Self-Attention**: 멀리 떨어진 단어 간의 관계도 직접적으로 파악.

## Self-Attention: 단어 간의 관계 네트워크

Self-Attention은 문장 내 한 단어가 다른 모든 단어와 얼마나 관련이 있는지(Attention Score)를 계산하여, 중요한 단어의 정보에 더 '집중'하는 메커니즘입니다.

"못했다"라는 단어를 처리할 때, '동물', '건너지', '피곤해서' 등의 단어에 Attention을 줌

그      동물은      길을      건너지      못했다      왜냐하면      피곤해서

## 핵심 혁신 ②: 스케일링 법칙 (Scaling Laws)

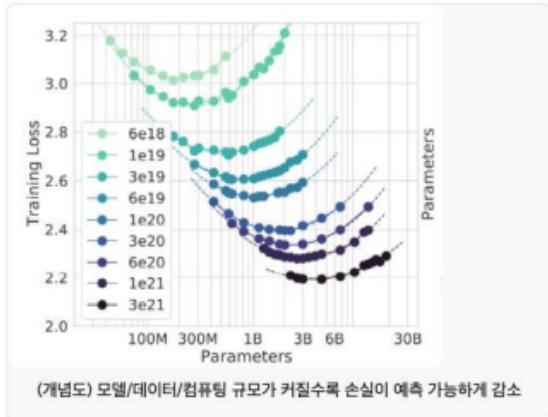
### "크기가 성능을 만든다"

2020년 OpenAI의 연구는 LLM의 성능이 세 가지 요소에 따라 예측 가능하게 향상된다는 것을 실험적으로 증명했습니다. 이를 **스케일링 법칙**이라고 합니다.

$$\text{성능} \propto (\text{모델 크기}, \text{데이터셋 크기}, \text{컴퓨팅 파워})$$

- **모델 크기 (Parameters)**: 신경망의 파라미터(가중치) 수. 클수록 더 복잡한 패턴 학습 가능.
- **데이터셋 크기 (Tokens)**: 모델이 학습하는 텍스트의 양. 많을수록 더 넓은 지식 습득.
- **컴퓨팅 파워 (FLOPs)**: 학습에 사용되는 총 연산량.

이 세 가지 요소를 균형 있게 함께 늘리면, 모델의 성능(손실 값 감소)은 역함수 (Power-law) 형태로 꾸준히 향상됩니다.



### 스케일링 법칙의 시사점

"더 똑똑한 모델을 만들고 싶다면, 더 큰 모델을, 더 많은 데이터로, 더 오래 학습시켜라!"

이 법칙은 GPT-3와 같은 초기대 AI 모델 개발의 이론적 토대가 되었고, 이후 LLM 경쟁을 촉발시켰습니다.

# LLM의 현실 세계 응용 사례

## 산업을 바꾸는 게임 체인저

LLM은 다양한 산업 분야에서 혁신적인 서비스를 가능하게 하며 빠르게 확산되고 있습니다.

### 콘텐츠 생성 & 마케팅

블로그 글, 광고 카피, 이메일, 제품 설명 등을 자동으로 생성하여 생산성을 극대화합니다.

**예시:** Jasper, Copy.ai

### 대화형 AI & 챗봇

단순 응대를 넘어, 고객의 의도를 파악하고 사람처럼 자연스러운 대화가 가능한 지능형 챗봇을 구현합니다.

**예시:** ChatGPT, 기업용 AI 상담원

### 문서 요약 & 지식 관리

긴 보고서, 뉴스 기사, 논문을 핵심 내용만 간추려 요약하고, 문서 기반 질의응답 시스템을 구축합니다.

**예시:** 회의록 요약, 법률 문서 분석

### 코드 생성 & 개발 지원

개발자의 주석이나 함수명만 보고 필요한 코드를 자동으로 완성해주어 개발 속도를 향상시킵니다.

**예시:** GitHub Copilot, Amazon CodeWhisperer

### 주의할 점: 환각과 편향

LLM은 매우 유용하지만, 때때로 사실이 아닌 내용을 그럴듯하게 지어내는 **환각(Hallucination)** 현상이나, 학습 데이터에 내재된 **편향(Bias)**을 드러낼 수 있습니다. 따라서 중요한 작업에는 반드시 사람의 검토가 필요합니다.

### 이번 시간 정리

- **생성형 AI vs 전통적 AI**: 전통적 AI가 **판단**과 **예측**에 집중했다면, 생성형 AI는 새로운 콘텐츠를 **창조**하는 데 중점을 둡니다.
- **LLM**: Transformer 아키텍처 기반의 거대한 신경망으로, 방대한 텍스트 데이터를 학습해 인간 수준의 언어 능력을 갖추었습니다. 구조에 따라 GPT(생성), BERT(이해), T5(변환) 계열로 나뉩니다.
- **핵심 혁신**: 순차 처리의 한계를 극복한 **Transformer & Self-Attention**과, “크기가 성능을 결정한다”는 **스케일링 법칙**이 LLM의 비약적인 발전을 이끌었습니다.
- **응용**: LLM은 챗봇, 콘텐츠 생성, 코드 작성, 문서 요약 등 우리 삶과 산업 전반에 이미 깊숙이 들어와 있습니다.

질문 있으신가요?