

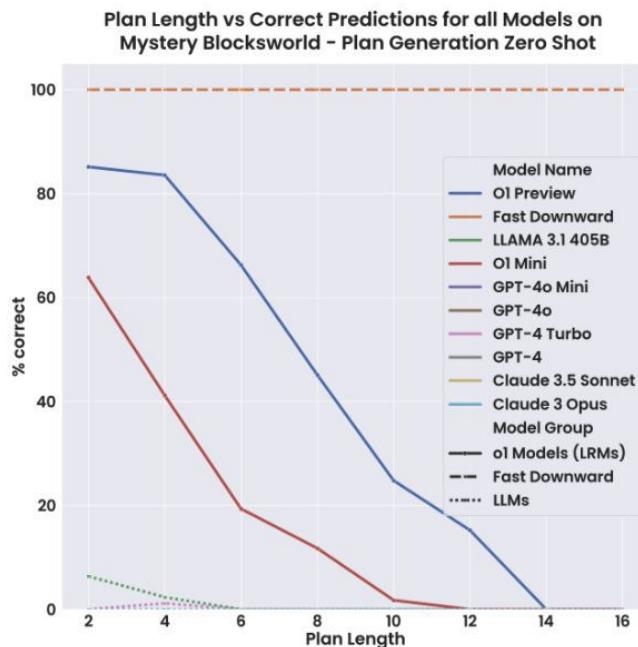
Neural & Symbolic Decision Making Framework

2026.02.04
peny.official

kakao

LLM의 한계 - 추론과 계획의 부재

- 대화와 생성 능력은 뛰어나나, 복잡한 제약 조건이 있는 추론 및 계획(Reasoning & Planning)에서 낮은 성능을 보임.
- e.g., 여행 계획(Travel Planning) 문제에서 GPT-4 Turbo의 성공률은 단 0.6%



세 가지 해결 방향 (Solutions)

- Option 1: Scaling Law
 - 더 많은 데이터와 자원 투입 (비용 과다 및 근본적 한계).
- Option 2: Hybrid System
 - 딥러닝과 심볼릭 솔버(Symbolic Solver)의 결합
- Option 3: Symbolic Emergence
신경망 내부에서 심볼릭 구조가 스스로 수렴하는지 탐구

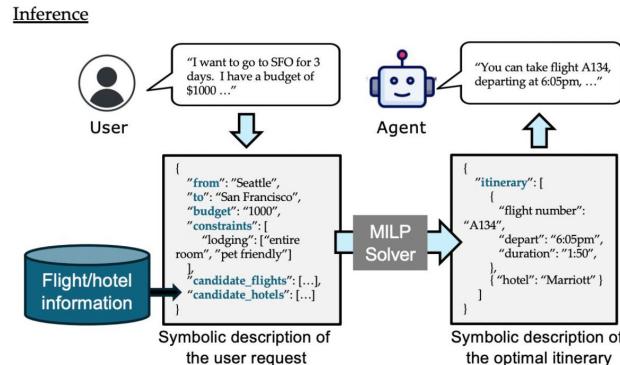
Hybrid System

하이브리드 접근 (1) - Tool Use & APEC-Travel

모델이 솔버를 '도구'로 쓰는 것: LLM이 자연어 요청을 JSON 형태의 심볼릭 언어로 번역하면, 전문 솔버가 최적해를 계산.

- APEC-Travel 사례

- 1. 사용자 요청 → LLM 번역 (JSON)
- JSON + DB 정보 → MILP 솔버 입력
- 실시간 최적 경로 도출



$$\begin{aligned} \max z &= 4x + 5y \\ \text{s.c. } &2x + y \leq 8 \\ &x + 2y \leq 7 \\ &y \leq 3 \\ &x, y \geq 0, \text{ integers} \end{aligned}$$

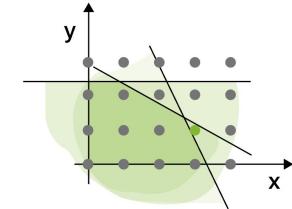


Figure 2. Example of a mixed integer programming (MIP) model and its solution space: x and y are the decision variables, and z is the objective function. The inequalities form the constraint boundaries, represented as lines. Blue dots indicate feasible (valid) solutions that satisfy all constraints, while the green dot marks an optimal solution that maximizes the objective.

하이브리드 접근 (2) - Search Augmented, SearchFormer

Search Augmented: Symbolic Solver가 생성한 탐색 과정(Trace)

데이터를 모델 학습에 활용하는 방식

(vs Solution Only: 쿼리에서 바로 최종 정답을 예측하도록 학습 하는 방식)

1. “SearchFormer”: A* 탐색 알고리즘의 과정을 토큰 예측으로 모델링.
 - a. Search Augmented: Model은 이 문제를 설명하는 프롬프트를 입력 받아 Solver의 탐색 과정을 한 토큰씩 예측하도록 교사 학습(Teacher Forcing) 방식으로 학습
 - b. 부트스트래핑(Bootstrapping): 모델이 생성한 경로 중 더 효율적인 것을 골라 재학습
2. “DualFormer”(“SearchFormerv2”): 탐색 과정(Trace) 데이터의 누락 전략
 - a. 모델은 문제의 난이도에 따라 스스로 모드를 전환하는 능력을 갖추게 됨.

하이브리드 접근 (3) - 통합형 접근 (E2E) - SurCo

딥러닝 모델을 심볼릭 솔버를 통합합니다.

현실의 문제

비선형적 목적함수 + 조합된 제약 조건(Combinatorial Constraints) ->
기존 Solver 활용으로는 해결하기 힘듦.

해결책

잠재 공간(Latent Space)에서 선형 대리(Surrogate) 문제를 구성하여 해결하지

- 비선형 문제를 잠재 공간에서 솔버가 다룰 수 있는 선형 대리 문제로 변환.
- 솔버를 미분 가능한 레이어로 취급하여 전체 시스템을 최적화

하이브리드 접근 (3) - 통합형 접근 (E2E) - SurCo

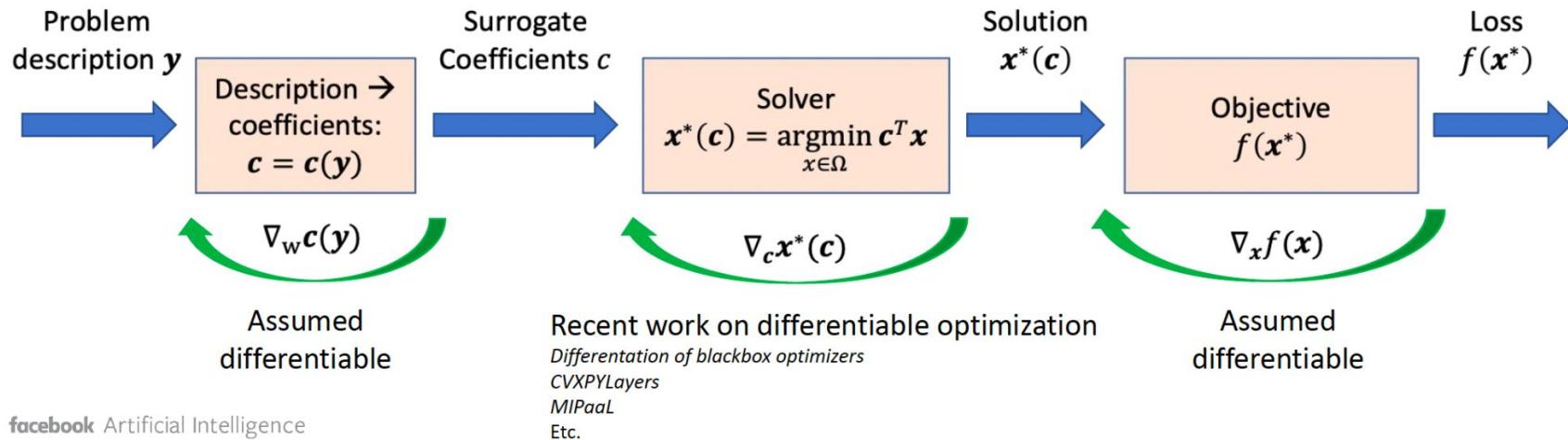
e.g., 임베딩 테이블들을 GPU(장치)들에 할당하되, 각 GPU의 메모리 용량 제한을 지키면서 시스템의 전체 처리 지연 시간(Latency)을 최소화

$$\min_x f(x; y) \text{ s.t. } x \in \Omega(y) \longrightarrow \hat{x}^*(y) = \arg \min_x c(y)^T x \text{ s.t. } x \in \Omega(y)$$

- $f(x; y)$: 풀고자 하는 비선형 최적화 문제
- x : 어떤 테이블을 어느 GPU에 배치할지를 결정하는 이진 변수(0,1)
 - $x \in \Omega(y)$: y 에 의해 정의된 '영역(Feasible Region)' 안에 존재해야 함
- y : 테이블의 메모리 요구량, 각 장치의 메모리 용량 제한 등 문제의 명세가 입력됩니다.
- $c(y)$: 선형 대리 비용 계수(c)를 예측하는 신경망 모델
- c : $c(y)$ 에 의해 예측되는 각 테이블을 특정 장치에 할당할 때 발생하는 예상비용

하이브리드 접근 (3) - 통합형 접근 (E2E) - SurCo

e.g., 임베딩 테이블들을 GPU(장치)들에 할당하되, 각 GPU의 메모리 용량 제한을 지키면서 시스템의 전체 처리 지연(Latency)을 최소화



- 먼저 딥러닝 모델(신경망)이 제약조건을 입력받아, 각 테이블을 특정 장치에 할당할 때 발생하는 가상의 비용인 선형 대리 비용 계수(c)를 예측
- 모델이 예측한 계수 c 를 사용하여 단순한 선형 최적화 문제($\min c^T x$)를 구성
- MILP 솔버는 메모리 제한 등 모든 조합론적 제약 조건을 엄격히 준수하면서, 예측된 비용 c 를 최소화하는 최적의 배치 해(x^*)를 도출
- 수신기마다 온 베하 경로(차로)와 개인화된 상황에 맞는 차로 확률(0~1)에 따라 차로 평가

하이브리드 접근 (3) - SurCo의 한계점

- 비선형 목적 함수 f 가 해 x 에 대해 미분 가능해야 Gradient Optimization 사용 가능
- 학습 과정에서 신경망을 업데이트할 때마다 매번 조합 최적화 솔버를 호출해야 합니다.
- Vertex Solution: 선형 솔버의 특성상 최적해가 항상 **꼭짓점 (Vertex)**에서만 발견됨.
(그러나 문제들의 최적해가 꼭짓점에 위치하는 경우가 많아 실무적으로는 여전히 유효함)

Extreme Point Theorem

Every linear program that has an optimal solution has an optimal solution that is a vertex of the feasible polyhedron. (Linear Programming, W. H. Freeman, 1983)

선형계획문제에 최적해가 존재한다면, 그 최적해 중 하나는 반드시 가능해역의 꼭짓점에서
발생된다. 결과는 항상 $(-1, -1), (1, -1), (-1, 1), (1, 1)$ 중 하나의 꼭짓점

Symbolic Emergence

Is LLM doing retrieval or true reasoning?

LLM의 한계와 새로운 돌파구 - 검색(Retrieval)을 넘어 추론(Reasoning)

- **현재의 논쟁:** LLM은 데이터를 단순히 '근사적으로 검색'하는가, 아니면 '논리적 추론'을 하는가?
- **실험적 증거:** 수학 문제에 무관한 문장을 섞으면 성능이 급락하는 현상(GSM-NoOp)은 모델이 데이터 검색에 의존함을 시사함.
- **해결의 실마리:** 모듈러 덧셈과 같은 기초 산술 과제를 통해 신경망 내부의 표현 방식을 정밀 분석함.

주파수 영역에서의 푸리에 변환을 통한 가중치 분석

- 문제: 신경망의 원래 가중치(w)는 사람이 이해할 수 없는 무작위 숫자의 나열(블랙박스)처럼 보임.
- 해결책: 실수 가중치(w)를 (Scaled) Fourier Transform을 통해 주파수 영역의 복소수 계수(z)로 변환 -> 주파수(k)와 히든 레이어 (j) 사이에서 규칙적인 패턴

$$z_{pkj} = \sum_{m=0}^{d-1} w_{pmj} e^{imk/d}$$

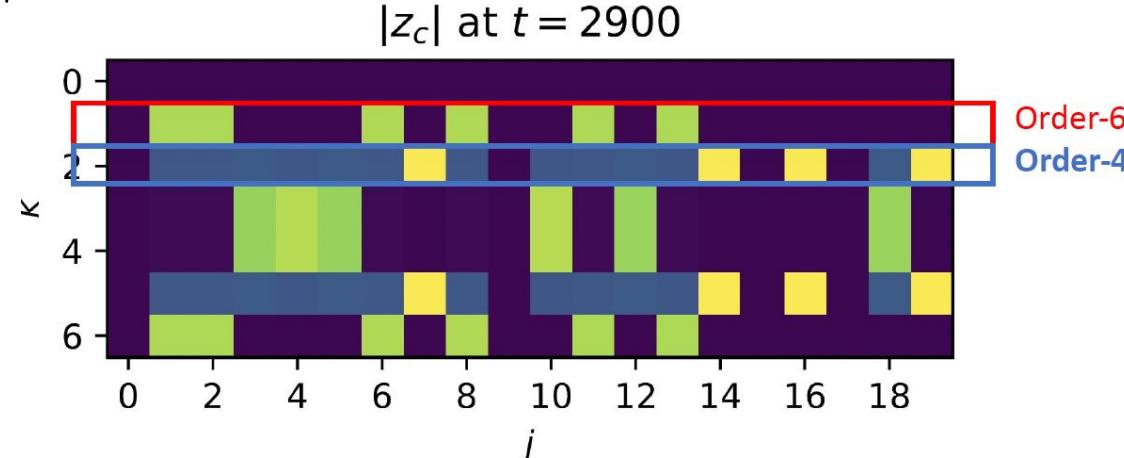
- w : 신경망의 hidden layer에 연결된 가중치 (일반적 DL/ML의 파라미터 Weight)
- $e^{imk/d}$: 푸리에 기저(지표함수)로 Weight의 숫자를 특정 주기를 가진 파동 패턴으로 매핑하는 역할
- z_{pkj} : 푸리에 기저를 통해 주파수로 변환된 복소수 계수. 이 계수에서 수학적 패턴을 발견함
- k : 주파수 (frequency): 패턴의 주기

kakao

차수(Order)와 노드 활성화

모델이 정답을 찾는 단위 - Order-4와 Order-6

- 차수(Order): 특정 주파수 성분을 처리하기 위해 할당된 은닉 노드의 개수
 - e.g., 주파수 $k=1$ 인 구역에 6개의 노드가 활성화되어 있다면 이를 Order-6 해라고 부릅니다.
- 노드 활성화 판정: 복소수 계수 z 의 크기가 역치(0.05)를 넘을 때 해당 노드가 해당 주파수 노드에 차수라는 거다. 가즈해스! (→ $|z_c| > 0.05$)



CoGS 프레임워크: 부분해의 결합을 통한 전역 최적해 구성

가중치를 숫자 집합이 아닌 연산 법칙이 적용되는 반환(Semi-ring) 공간으로 인식

- 링 덧셈(+): 은닉 노드들을 가로로 이어 붙이는 연산(Concatenate)
- 링 곱셈(*): 은닉 차원을 따라 가중치 간의 크로네커 곱(Kronecker product) 수행

$$\ell_k(z) = -2r_{kkk} + \sum_{k_1, k_2} |r_{k_1 k_2 k}|^2 + \frac{1}{4} \left| \sum_{p \in \{a, b\}} \sum_{k'} r_{p, k', -k', k} \right|^2 + \frac{1}{4} \sum_{m \neq 0} \left| \sum_{p \in \{a, b\}} \sum_{k'} r_{p, k', m-k', k} \right|^2$$

$\|Output - one_hot(c)\|_2^2$

The diagram illustrates the decomposition of the loss function $\ell_k(z)$ into two main components:

- Red Box (Synthesis Component):** This part represents the synthesis of three hidden states (k_1, k_2, k) into a single output. It consists of:
 - A term involving the magnitude squared of the weight vector $r_{k_1 k_2 k}$.
 - A term involving the magnitude squared of the sum of weights $r_{p, k', -k', k}$ for $p \in \{a, b\}$.
- Blue Box (Volume Component):** This part represents the volume component, involving the magnitude squared of the sum of weights $r_{p, k', m-k', k}$ for $m \neq 0$ and $p \in \{a, b\}$.

Below the equation, arrows point from the terms to their respective labels:

- $z_{syn}^{(k)}$ (차수 3 부분해) points to the red box.
- $z_{\nu}^{(k)}$ (차수 2 부분해) points to the blue box.

At the bottom, the formula for the final output is given as:

$$z_{F6} = \frac{1}{\sqrt[3]{6}} \sum_{k=1}^{(d-1)/2} z_{syn}^{(k)} * z_{\nu}^{(k)} * y_k$$

차수 6 최적 해
(Order-6)

조립된 해가 정답이 되는 이유

모델이 이차 활성화 함수(x^2)를 사용하기 때문에 성립합니다.

- 복잡한 L2 손실 함수를 합 전위(Sum Potentials, SPs)들의 다항식으로 완벽하게 분해
- 가중치 공간에서의 조립 결과가 실제 손실 값 계산에서도 보존되는 환 준동형 사상 성질

$$\|Output - one_hot(c)\|_2^2$$

$$\ell_k(z) = -2r_{kkk} + \sum_{k_1, k_2} |r_{k_1 k_2 k}|^2 + \frac{1}{4} \left| \sum_{p \in \{a, b\}} \sum_{k'} r_{p, k', -k', k} \right|^2 + \frac{1}{4} \sum_{m \neq 0} \left| \sum_{p \in \{a, b\}} \sum_{k'} r_{p, k', m-k', k} \right|^2$$

"손실을 줄여라" → "특정 합 전위값을 0 또는 1로 맞춰라"

(*합전위 : 모든 은닉 노드(j)에 걸쳐 특정 가중치들의 곱(단항식)을 합산한 값)

kakao

$$r(z) = \sum_j \prod z_{pkj}$$

CoGS 의의

실제 경사 하강법으로 얻은 해의 ****95~98%****가 이론적으로 조립한 Order-4/6 구조와 정확히 일치

시사점

신경망은 우리가 모르는 효율적인 심볼릭 표현을 스스로 학습함.

경사 하강법 없이 대수적 조립만으로 모델을 학습시키는 새로운 알고리즘 가능성.

딥러닝의 블랙박스 내부를 수학적 원칙에 따라 투명하게 이해하는 경로 제시

한계

모듈러 연산(아벨 군) 문제 + 2차 활성화 함수"라는 조건 하에서 이루어진 것

END