

□ 연구 개요

○ 배경

- 국내 ADMET 예측 연구는 K-MELLODDY, FAM(Federated ADME/ADMET Model) 등 다중 파라미터 예측을 수행하는 고도화 모델이 등장했으나, 여전히 고정 입력 - 단일 출력 중심의 정적 예측 시스템에 머물러 있음.
- 도구 선택 - 실험 조건 설계 - 결과 해석 - 반복 개선을 자율적으로 수행하는 Agent 기반 능동 학습 구조는 도입되지 않음.
- 국외에서는 ChemCrow, BioGPT 등과 같은 신약개발 특화 에이전트 시스템이 등장하며, 약물 구조 생성, ADMET 예측, 독성 평가, 실험 설계를 자율적으로 수행하는 구조를 갖추고 있음. 이러한 시스템들은 Chain-of-Thought reasoning, Tool Augmentation, Self-Refinement 등의 기능을 통해 AGI적 추론 능력의 일부를 구현하였으나, ADMET 전주기(흡수 - 분포 - 대사 - 배설 - 독성) 전체를 포괄하는 통합적 AGI형 에이전트는 아직 미개발 단계임.
- 따라서 본 연구는 ADMET AGI 모델 개발을 목표로, 신약개발의 예측 - 실험 - 피드백 과정을 통합하고, 자율적인 정책 최적화(Policy Optimization) 및 자동 실험 설계(Auto Experimental Design)를 수행하는 메타-학습 기반 Self-Evolving Agent 프레임워크를 제안함.

○ 관련 연구

- Symbolic Learning Enables Self-Evolving Agents (2024) : 언어 에이전트를 기호(Symbolic)네트워크로 보고, 자동 학습 진화를 가능하게 하는 프레임워크 제안
- A Survey on Self-Evolution of Large Language Models (2024) : LLM이 자체적으로 진화하는 접근법을 정리, 경험 획득, 정체, 업데이트에 대한 평가 주기 제안
- When Meta-learning Meets Online and Continual Learning (2025) : 메타학습과 온라인 지속 학습의 융합을 정리
- ChemCrow: Augmenting large-language models with chemistry tools (2023) : 화학 분야(유기합성, 약물발견, 재료설계) 과제를 위해 LLM에 화학전문 도구들을 결합한 에이전트 프레임워크

- BioGPT: Generative Pre-trained Transformer for Biomedical Text Generation and Mining (2022) : 생의학 문서 생성·텍스트마이닝을 위한 GPT계열 모델

○ 기존 연구의 한계점 및 개선 사항

- 기존의 연구들은 대부분 정적 모델로 구현되어 입력데이터 변화나 실험 환경의 불확실성에 능동적으로 대응하지 못하는 구조적 한계를 지님
- 특정 데이터 조건에서 최적화된 단일 속성 중심의 정적 예측기 형태로 도구선택, 실험 조건 설정, 결과해석, 반복 개선 등 능동적 의사결정 기능 부재
- 약물 개발 지능형 에이전트(ChemCrow, AutoBioGPT, OpenBioLLM 등) 시스템 또한 단계별 기능의 부분적 자율화 수준에 머물고 있음
- 단순한 예측 정확도 향상을 넘어 에이전트가 스스로 “언제, 무엇을, 어떻게, 어디서” 진화할지를 판단하고, 정책을 최적화하며, 실험 설계를 반복적으로 개선하는 self-evolving 구조를 구현
- 모델은 when, what, how, where 네 가지 축으로 구성되며, 이 네 축이 연결되어 하나의 자가 진화형 메타 학습 루프를 형성. 즉, 에이전트는 주어진 문제 상황에서 언제 진화할지 무엇을 진화시킬지, 어떤 방식으로 진화할지 어디서 진화를 적용할지를 스스로 판단하고 실행함

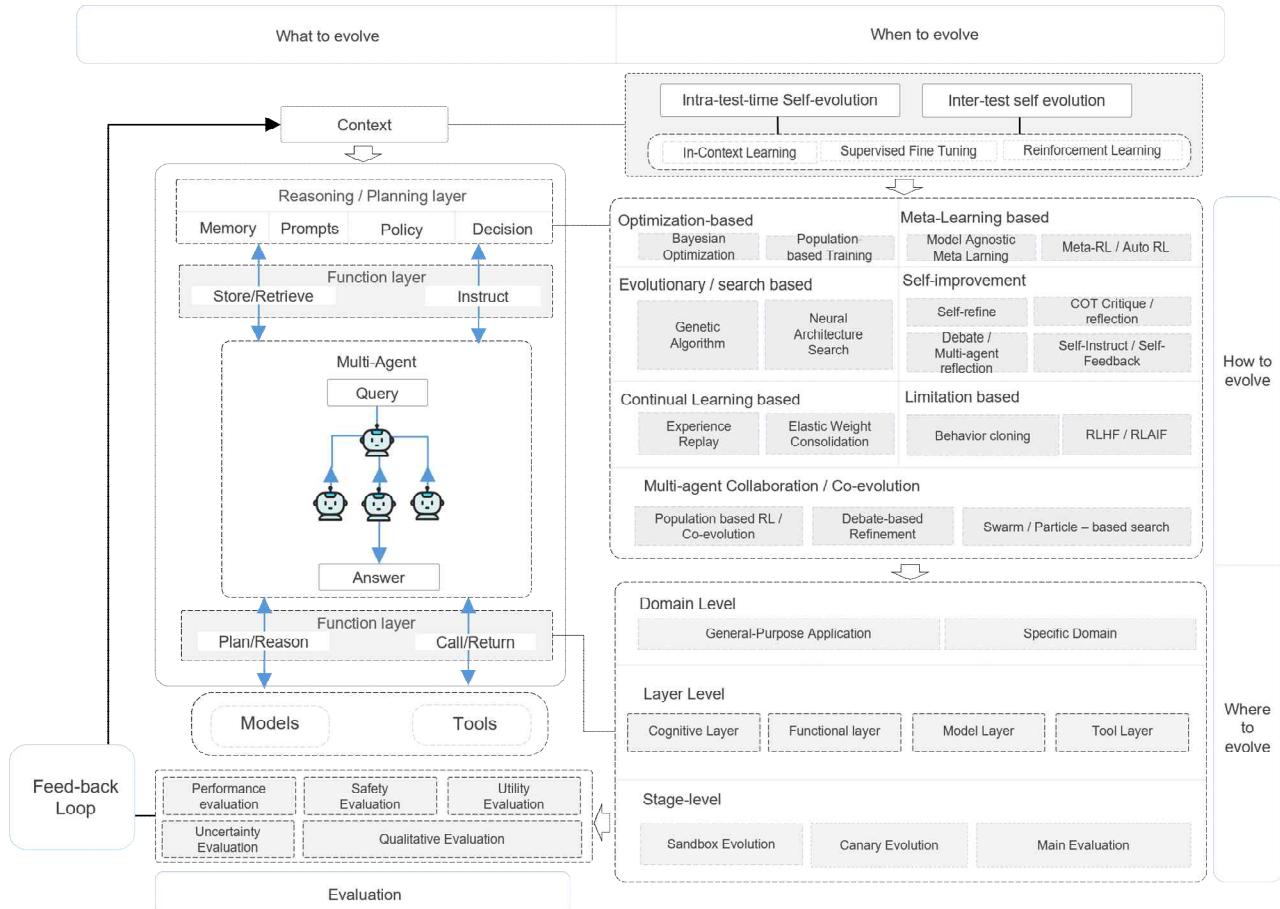
○ 연구 목표

- ADMET 전주기 영역에서 지속학습과 자율 최적화(self-optimization)을 수행 할 수 있는 LLM 기반 self-evolving agent 프레임워크를 개발
- 단순한 예측 정확도를 넘어 에이전트가 스스로 학습 타이밍과 진화 방향을 결정하고 정책(policy), 프롬프트, 도구, 모델을 동적으로 개선하는 self-evolving ADMET AGI 시스템 구현을 목표로 함.
- 최종적으로 본 시스템은 ADMET 예측 모델의 지속적 성능 향상, 실험 조건 설계의 자율화, reasoning-planning-execution의 통합된 ADMET AGI 수준의 자기진화형 인공지능 구조 구현을 목표로 함

□ 주요 설계

○ Self-evolving Framework 설계도

- 본 프레임워크는 LLM(Large Language Model) 을 기반으로 스스로 학습 시점과 방법을 결정하며 성능을 평가·개선하는 자기진화(Self-Evolving) 구조를 갖춤.



- 자율-메타 학습 루프(self-evolving meta learning loop) 설계

- 에이전트가 스스로 학습 시점, 방법, 적용 영역을 판단 할 수 있도록 when/what/how/where to evovle 의 4축 구조를 정의하고 상호 연계함
- 축별로 다음과 같은 메타 학습 모듈을 구현
 1. when to evolve: ICL, SFT, RL 기반의 진화 타이밍 정책 학습(Metra-RL controller)
 2. what to evolve: Policy / Prompt / Tool / Model / Memory 등 진화 대상 식별 및 선택
 3. how to evolve : PBT, PPO-/lagrangian, Meta-RL, Self-Refine 등 진화 알고리즘 통합
 4. where to evolve : Stage, layer, domain 기반 적용 환경 제어
- 정책 최적화(policy optimization) 및 자동 실험 설계 프레임워크 구축
 - Policy manager, experiment designer, feedback evaluator 세 모듈로 구성된 상위 관리 구조를 설계하여 책임과 데이터 흐름을 명확히 분리

1. Policy manager: 학습 개시 중간 재시도 타이밍을 관리
2. Experiment Designer: 하이퍼파라미터 툴 조합 실험 조건으로 자동 탐색
3. Feedback evaluator 가 결과를 Context에 반영하여 다음 주기 학습을 유도
 - 강화학습 기반의 self-evolving combined policy 프레임워크를 적용하여 정책 탐색 - 보상평가 - 자기진화 루프를 자동화
 - representation bank를 통해 축적된 교훈과 전략을 재활용함으로써 탐색 효율을 극대화
 - 위 구조의 실험 단계에서 얻은 평가 결과를 context에 재귀적으로 반영하여 자기개선 (self-refinement)을 수행

○ 수도 코드 (Pseudo code)

Input: Context Information C, Model Parameters θ , Memory M

Output: Updated Self-Evolving Agent A*

```

function SELF_EVOLVING_FRAMEWORK(C,  $\theta$ , M):
    Initialize Agent A with base LLM and Policy
    Initialize Context Layer, Reasoning Layer, Function Layer, Evaluation Module

    while Agent A is active do
        # (1) Context Recognition
        C ← Update environment state, user input, and feedback
        Store C in Memory

        # (2) Reasoning & Planning
        Plan ← Generate reasoning chain using Context
        Decision ← Determine What, How, Where to evolve

        # (3) Meta-Controller: When to evolve
        if MetaController.trigger_evolution(C):
            Target ← MetaController.define_target(C)
            Method ← MetaController.select_method(C)
            Scope ← MetaController.select_scope(C)

            # (4) Function Layer: Execution
            Execute(Plan, Method, Target, Scope)
            Update(A) by applying diffusion or adversarial refinement

```

```
else:  
    Perform regular reasoning and inference  
  
# (5) Evaluation & Feedback  
Metrics ← Evaluate(Performance, Utility, Uncertainty,  
                    Robustness, Reasoning Coherence, Human Feedback)  
FeedbackLoop(Metrics, C)  
Update Policy and Memory based on Metrics  
  
end while  
  
return A*
```

□ 2단계 개발 일정

□ 참고문헌

1. Zhang, Y., Li, T., & Wang, X. (2024). Symbolic Learning Enables Self-Evolving Agents. arXiv preprint arXiv:2403.01234. <https://arxiv.org/abs/2403.01234>
2. Liu, Q., Zhao, J., Chen, H., & Wang, Y. (2024). A Survey on Self-Evolution of Large Language Models. arXiv preprint arXiv:2406.07890. <https://arxiv.org/abs/2406.07890>
3. Chen, R., Sun, L., & Zhao, B. (2025). When Meta-Learning Meets Online and Continual Learning. arXiv preprint arXiv:2501.04512. <https://arxiv.org/abs/2501.04512>
4. Bran, A. M., Cox, S., Schilter, O., Baldassari, C., White, A. D., & Schwaller, P. (2023). ChemCrow: Augmenting large-language models with chemistry tools. arXiv preprint arXiv:2304.05376. <https://arxiv.org/abs/2304.05376>
5. Luo, R., Sun, L., Xia, Y., Qin, T., Zhang, S., Poon, H., & Liu, T.-Y. (2022). BioGPT: Generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics*, 23(6), bbac409. <https://doi.org/10.1093/bib/bbac409>
6. Gao, H.-a., Geng, J., Hua, W., Hu, M., Juan, X., Liu, H., Liu, S., Qiu, J., Qi, X., Wu, Y., Wang, H., Xiao, H., Zhou, Y., Zhang, S., Zhang, J., Xiang, J., Fang, Y., Zhao, Q., Liu, D., Ren, Q., Qian, C., Wang, Z., Hu, M., Wang, H., Wu, Q., Ji, H., & Wang, M. (2025). A Survey of Self-Evolving Agents: On Path to Artificial Super Intelligence. arXiv preprint arXiv:2507.21046. <https://doi.org/10.48550/arXiv.2507.21046>