

코파일럿과 코드 수정

이창공

2023년 11월 23일

Abstract

프로그램 자동 수정기술 대중화의 첫걸음은 언어모델이 내디뎠다. 언어모델을 사용한 프로그램 자동 수정은 고유한 강점이 존재한다. 이러한 방식은 열거식 프로그램 합성 기법과 유사한 측면이 있다. 따라서 언어모델 기법의 프로그램 자동 수정의 다음 단계는 유한한 시간에 조건을 만족하는 수정을 만들어 내는 보장성에 있을 것으로 예상된다.

프로그램 자동 수정기술 대중화의 첫걸음은 언어모델이 내디뎠다. 언어모델 기반 코드 생성기인 코파일럿이 자동 수정 서비스 제공을 시작했다. 구문 분석 시 나타나는 오류 메시지와 오류에 해당하는 코드 구문, 그리고 소스코드의 일부 문맥을 모델에게 입력으로 주고 오류 부분을 지우고 다시 예측하는 방식이다. 사용자는 예측된 수정을 오류가 있는 부분에 적용하거나 다음으로 확률이 높은 수정을 선택 하는 등 수정의 순수 자동화보다는 이름에 맞게 개발자를 보조하는 목적으로 사용된다. 이러한 방식은 작년인 22년에 논문으로 발표된 최첨단 기술 알파수리와 고수준에서 같은 방식이다. 본래 언어모델이 그러하듯, 해당 기능은 신뢰 할 수 없지만 편리하고, 수정의 실패가 잦고 불안전하지만 빠르다.

언어모델을 사용한 프로그램 자동 수정은 고유한 강점이 존재한다. 대형 언어 모델이 화두가 되기 이전인 2020년, 캐나다 워털루 대학의 연구진은 프로그램 자동 수정 문제에서는 이례적으로 인공 신경망을 사용한 프로그램 수정 기술을 개발했다. 당시에는 오류 지점과 오류 지점의 주변 문맥을 개별적으로 학습하여 사용하는 앙상블 학습 방식으로 코드의 트리 구조를 더 잘 표현할 수 있는 합성곱 신경망(CNN) 구조와 신경망 기계 번역기를 같이 사용하였다. 코코넛의 성능은 초기 버전임에도 기존 방식의 성능을 압도할 뿐만 아니라 여러 언어를 동시에 지원할 수 있다는 새로운 가능성을 보였다. 그뿐만 아니라, 전통적인 방식들이 고칠 수 없었던 복잡한 수정을 이후 언어모델 기반 수정 기술들이 고쳐나가기 시작하면서 코파일럿이 대중화의 선두에 나서는 상황이 만들어지는 배경을 만들게 된 것이다.

이러한 방식은 열거식 프로그램 합성 기법과 유사성이 존재한다. 코파일럿과 같이 구문 오류를 단적인 예로 구문 오류가 없어질 때까지 계속해서 다음 순위의 수정을 생성한다면, 열거식 프로그램 합성 기법과 유사하다고 볼 수 있을 것이다. 그러나 이러한 방식은 합성 기법과 달리 유한한 시간에 올바른 생성이 완료된다는 보장이 없고 따라서 비용적인 측면에 대한 보장도 없다.

언어모델 기법의 프로그램 자동 수정의 다음 단계는 유한한 시간에 조건을 만족하는 수정을 만들어 내는 보장성에 있을 것으로 예상된다. 이러한 문제는 사실 언어모델뿐만 아니라 인공 신경망 기술 공통으로 마주한 확률적 방법론의 문제이기도 하다. 가까운 미래에 자율 주행처럼 나의 프로그램을 믿고 맡길 수 있는 자동 수정 기술이 탄생할지 기대와 우려되는 바이다.