ChatGPT를 활용한 프로그래밍 문제의 정답 생성 및 테스트 케이스 검증 연구

이도현[1,+], 최예나[2,+], 김미수[3,*]
[1]전남대학교 인공지능학부 학부생
[2]전남대학교 인공지능융합학과 석사과정
[3]전남대학교 인공지능융합학과 교수
(+: 공동 1 저자, *: 교신저자)
leedo573@jnu.ac.kr, ynaa550@jnu.ac.kr, misoo.kim@jnu.ac.kr

A study on programming problem solution generation and test case verification using ChatGPT

Do-Hyun Lee[1], Ye-Na Choi[2], Mi-Soo Kim[2]
[1]Dept. of Artificial Intelligence, Chonnam University
[2]Dept. of Artificial Intelligence Convergence, Chonnam University

요 약

인공지능과 자연어 처리 기술 발전으로 ChatGPT는 프로그래밍 분야에서도 활용될 수 있다. ChatGPT로 생성한 문제를 학습 도구로 사용하기 위해 테스트 케이스의 정확성을 검증하는 것은 필수적이다. 본 연구는 ChatGPT를 통해 프로그래밍 문제, 정답 코드, 테스트 케이스를 생성하고 이들의 정확성을 평가하고자 한다. 생성된 테스트 케이스의 정확성을 높이기 위해, 본 연구에서는 5번의 반복적인 테스트 케이스 생성 및 검증 과정을 거쳐 최종적으로 87.8%의 정확성을 달성했다. 이는 ChatGPT를 통한 프로그래밍 학습의 신뢰도를 높일 수 있음을 보여준다.

1. 서론

최근 인공지능 기술, 특히 자연어 처리(NLP) 기술의 발전은 다양한 산업에 혁신을 가져왔으며, ChatGPT[1]는 이러한 기술을 활용하여 프로그래밍 분야에서 코드 생성, 프로그램 복구, 코드 요약 등에 활용된다[2]. ChatGPT를 프로그래밍 학습 도구로 활용하기 위해 ChatGPT에게 문제, 문제에 대한 정답 코드, 코드 평가를 위한 테스트 케이스 생성을 요청할 수 있다[3]. 그러나 ChatGPT가 생성한 결과물에 대한 신뢰성의 문제에 직면할 수 있다[4]. 이 문제를 해결하기 위해 프로그래밍 문제에 대한 정답 코드와 테스트 케이스의 정확성을 검증할 필요가 있다. 본 연구에서는 ChatGPT를 활용하여 프로그래밍 문제를 생성하고 이를 바탕으로 생성된 정답 코드 및 테스트 케이스의 정확도를 분석하고자 한다.

2. 연구방법

2.1 문제, 정답 코드, 테스트 케이스 생성

본 연구에서는 ChatGPT의 다양한 버전 중 사용자의 접근성이 가장 높은 'GPT-4o mini'를 사용하였다. 문제 생성에 앞서 중복된 문제 생성을 방지하기 위해, 프로그래밍 문제 풀이 플랫폼인 LeetCode[5]에서 전체 71개의 문제 유형 중 C++ 언어와 관련이 없는 Database와 Shell을 제외한 69개의 문제 유형을 선택하였다. 이 선택된 문제 유형들을 프로그래밍 문제 생성 요청 프롬프트에 입력값으로 활용하였다.

'{문제 유형 입력} 유형으로 C++ 프로그래밍 문제를 만들어 주세요'라고 프롬프트를 작성하여 69개의 문제를 생성하였다. 그 이후 '{ChatGPT가 생성한 문제 입력}의 C++ 정답 코드를 만들어 주세요'와 '{ChatGPT가 생성한 정답 코드 입력}을 테스트하기 위한 10개의 테스트 케이스를 만들어 주세요'라는 순차적인 프롬프트를 통해 각각 정답 코드 69개, 테스트 케이스 690개를 생성하였다.

2.2 정답 코드의 정확성 평가

ChatGPT가 생성한 문제의 정답을 확인하기 위해 문제와 정답 코드를 순차적으로 불러와 정답 코드의 컴파일 에러를 점검하였다. 또한 문제의 요구사항이 정답 코드에서 적절히 구현되었는지 수동 검토하여 정확도를 평가하였다.

2.3 테스트 케이스의 정확성 평가 및 개선

ChatGPT가 생성한 테스트 케이스의 유효성을 평가하기 위해, 이전 단계에서 오답으로 분류된 문제를 제외한 나머지 정답 코드를 실행하고, 각 정답 코드의 테스트 케이스 내 입력값을 프로그램에 입력하여 실제 결과를 얻었다. 그리고 이 값을 ChatGPT가 제공한 테스트 케이스 내 예상 출력값과 비교하여 두 값이 일치하지 않는 경우, 해당 테스트 케이스는 실패한 것으로 평가하였다.

ChatGPT는 반복적인 대화를 통해 작업 성능을 높일 수 있다[6]. 이에 따라 실패한 테스트 케이스를 해결하기 위해 ChatGPT에 정답 코드에 맞는 새로운 테스트 케이스 생성을 요청하였다. 이 과정을 총 5회 반복하여 정확도 향상을 평가하였다.

3. 연구결과

3.1 ChatGPT가 생성한 정답 코드의 정확도

69개의 문제에 대한 정답 코드를 평가한 결과, ChatGPT는 66개 문제에서 정답을 생성하여 95.7%의 정확도를 기록하였다(표 1). 다만, 오답의 경우 정답 코드가 문제의 요구사항은 적절하게 구현했으나, 컴파일 에러가 발생하는 문제가 있었다.

<표 1> 정답 코드의 정확도

정답 수	오답 수	전체 문제 수	정확도
66	3	69	95.7%

3.2 ChatGPT가 생성한 테스트 케이스의 정확도

정답 코드를 생성한 66개의 문제 중 테스트 케이스를 평가할 수 있는 함수를 제공한 37개의 정답 코드에 대해 테스트 케이스 정확도 평가 결과를 표 2에 요약하였다.

<표 2> 테스트 케이스의 정확도

TC 생성 횟수	TC 성공 개수	TC 실패 개수	전체 TC 개수	정확도
1	244	136	370	65.9%
2	296	74	370	80%
3	315	55	370	85.1%
4	318	52	370	85.9%
5	325	45	370	87.8%

테스트 케이스의 첫 번째 생성에서는 전체 370개의 테스트 케이스 중 244개가 성공하여 65.9%의 정확도를 기록하였다. 그러나 테스트 케이스 생성 횟수를 거듭하면서 테스트 케이스의 정확도는 점차 향상되어 다섯 번째 테스트 케이스 생성에서는 87.8%에 달했다. 이러한 결과는 테스트 케이스의 반복적인 생성 및 평가 과정이 테스트 케이스의 품질을 높이는 데 중요한 역할을 한다는 것을 보여준다.

4. 결론

본 연구를 통해 ChatGPT는 프로그래밍 문제와 정답 생성에 효과적인 도구임을 확인하였다. 69개의 문제 중 66개에서 올바른 정답을 생성하여 95.7%의 높은 정확도를 기록한 결과는 ChatGPT의 자연어 처리 능력이 프로그래밍 언어를 이해하고 활용하는 데 충분히 신뢰할 수 있음을 보여준다.

그러나 테스트 케이스 생성에 있어서는 여전히 개선의 여지가 있다. 5번의 반복적인 테스트 케이스 생성 과정을 통해 최종적으로 87.8%에 도달했다. 이는 테스트 케이스의 품질 향상에 긍정적인 영향을 미쳤지만, 근본적으로는 입력값의 다양성, 경계 조건, 예외 상황에 대해 보다 체계적으로 고려하는 접근이 필요하다. 향후 정교한 프롬프트 작성과 다양한 테스트 케이스 생성 전략을 통해 신뢰성을 더욱 높여야 한다.

사사

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 소프트웨어중심대학사업(2021-0-01409)과 과학기술정보통신부 및 정보통신기획평가원의 인공지능융합혁신인재양성사업(IITP-2023-RS-2023-00256629), 대학ICT연구센터사업(IITP-2024-RS-2024-00437718)의 연구 결과로 수행되었음

참고문헌

[1] ChatGPT. https://openai.com/blog/chatgpt

[2] Tian, Haoye, et al. "Is ChatGPT the Ultimate Programming Assistant–How Far Is It?" arXiv preprint arXiv:2304.11938, 2023.

[3] YANG, Boyang, et al. "CREF: An LLM-Based Conversational Software Repair Framework for Programming Tutors" In: Proceedings of the 33rd ACM SIGSOFT International Symposium on Software Testing and Analysis, 2024, p. 882-894.

[4] Lo, Chung Kwan. "What Is the Impact of ChatGPT on Education? A Rapid Review of the Literature" Education Sciences, 13.4: 410. 2023.

[5] LeetCode. https://leetcode.com

[6] OUYANG, Long, et al. "Training language models to follow instructions with human feedback" Advances in neural information processing systems, 35: 27730-27744. 2022.