

대형 언어 모델을 이용한 로봇 작업 분해 및 번역

김주혜, 김혜진, 임혁

한양대학교, 한국에너지공과대학교

pskjh155@hanyang.ac.kr, {hyejinkim, hlim}@kentech.ac.kr

Robot Task Decomposition and Translation Using Large Language Model

JuHye Kim, Hyejin Kim, Hyuk Lim

Hanyang University, Korea Institute of Energy Technology (KENTECH)

요약

로봇이 사용자의 복잡한 지시를 수행하기 위해서는 지시문을 이해하고 로봇이 수행해야 하는 일련의 세부 작업으로 분할한 후 로봇이 처리할 수 있는 동작 명령으로 변환하여야 한다. 본 논문에서는 로봇 작업 분할 및 명령어 변환을 처리하기 위해서 large language model (LLM)의 사용을 제안한다. 사용자 지시어로부터 로봇이 처리할 수 있는 동작 명령 변환의 효율적 처리를 위해 LLM의 fine-tuning 기법을 적용하였다.

I. 서론

오늘날 로봇이 사회 여러 분야에서 다양한 역할을 수행함으로써 우리 삶이 보다 편리해지고 있다. 예를 들어 특정 기능만을 수행하는 제조 로봇과 서비스 로봇이나, 주변 환경을 인식하여 자율적으로 판단하고 행동하는 지능형 로봇이 활발히 활용되고 있다. 최근 인공지능 기술이 지능형 로봇 분야에 적용되고 있지만, 사용자가 로봇에게 복잡한 지시를 전달하고 로봇이 이를 이해하여 수행하도록 하는 것은 여전히 어려운 문제로 남아 있다. 또한, 로봇은 사용자의 지시를 단순히 이해하는 것뿐만 아니라 최종 목표에 도달하기 위해 수행해야 하는 모든 작업과 단계를 체계적으로 정리하고 분석하는 과정이 필요하다. 이를 구현하기 위해서는 주어진 환경에서 사용자의 지시에 따라 로봇이 수행해야 하는 동작을 결정하기 위해, 복잡한 개별 알고리즘을 개발해야 한다.

대형 언어 모델 (large language model, LLM)은 대규모 데이터셋으로 학습된 인공지능 모델로써, 복잡한 자연어를 이해하고 처리한 다음, 추론 및 예측 과정의 결과로 의미 있는 정보를 제공한다. 즉, LLM은 사람처럼 텍스트를 이해하여, 새로운 텍스트를 생성하거나 요약하고, 질의응답을 하는 등의 높은 수준의 작업을 수행할 수 있다. 또한, LLM에 축적된 방대한 지식은 프롬프트 엔지니어링 (prompt engineering), fine-tuning과 같은 기법을 통해 목적에 맞추어 활용 가능하다. 프롬프트 엔지니어링은 LLM이 사용자가 원하는 정보를 출력하도록 프롬프트 입력을 설계하고 조정하는 과정을 의미한다. Fine-tuning은 LLM을 사용자가 원하는 특정 작업과 도메인에서 효과적으로 활용하기 위해 재학습하는 과정이다. 이와 같이, 수많은 지식을 내포하고 있는 LLM은 로봇을 포함한 다양한 분야의 연구와 산업에서 활발히 응용되고 있다 [1], [2].

II. 본론

본 논문에서는 로봇이 사용자의 복잡한 지시를 이해하고 수행하도록 하기 위한 방법으로 LLM의 사용을 제안한다. 자연어로 이뤄진 사용자의 지시를 현재 상황에 따라 이해하고 이에 대응하기 위한 Task와 각 Task를 수행하기 위한 동작 단위를 구분하도록 LLM을 활용할 수

```
Instruction=""
SYSTEM:
You are a Turtlebot3 that navigates environments without a pre-built map.
Turtlebot3 will provide answers in a structured list format.
Turtlebot3 have to follow the guidelines provided.

[Basic Tasks]
1. 정지 - stop
2. 각도에 따른 회전 - turn (degree)
3. 원형 회전 - rotate
4. 직진하기 - go straight
5. 순찰하기 - patrol between specified points
6. 촬영하기 - take a photo
7. 경비 모드 활성화 - activate guard mode
8. 보고 전송 - transmit report

[Turtlebot3 Profile]
- Turtlebot3 shaped like a turtle.
- Turtlebot3 have a camera to take a photo.
- Turtlebot3 have a LED to activate guard mode.
- Turtlebot3 have a buzzer to transmit report.

[INST]
Based on the guidelines provided:
1. Decompose the task into smaller subtasks.
```

[그림 1] 로봇 정보와 작업 처리 방식에 대한 instruction 프롬프트 제작

있다. LLM을 로봇 작업 수행에 효과적으로 활용하기 위해 fine-tuning 기법을 적용하였다. LLM에 fine-tuning 기법을 적용하기 위해서는 학습 데이터를 목적에 맞게 수집하는 것이 중요하다. 본 논문은 학습 데이터의 생성 및 수집을 위해 프롬프트 엔지니어링 기법을 활용하였다. 학습 데이터를 생성하기 이전에, 로봇이 수행할 수 있는 instruction 집합을 [그림 1]과 같이 정의하였다. Instruction 집합에는 로봇의 역할에 대한 정보, 로봇이 실행 가능한 기초 동작, 로봇이 처리 가능한 형식의 명령어, 모델이 사용자의 지시문을 처리하는 방법, 사용자의 특정 지시 사항에 대한 로봇의 응답 example이 명시되어 있다. 이 중에서, 모델이 사용자의 지시문에 대해 처리하는 방법을 다음과 같이 정의하였다. 사용자의 지시문을 이해하여 최종 목표 Task를 설정하고, 최종 목표에 도달하기 위해 로봇이 수행할 동작을 단계적으로 세분화하도록 하였다. 또한, 구체적인 세부 Tasks를 로봇이 처리할 수 있는 명령어로 변환하도록 설정하였다.

이와 같이 사전에 정의한 instruction 집합을 프롬프트 엔지니어링에서 다음과 같이 활용하였다. 프롬프트를 통해 모델에 instruction 정보를 알려준 다음, 사용자의 특정 지시 사항에 대한 프롬프트의 응답을 확인하고, 원하는 결과값을 출력하도록 반복적으로 조정하였다. 프롬

프롬프트 엔지니어링을 통해 찾은 잘 나오는 프롬프트는 차용하였으며, 실제 운영 시 확보된 모든 예제를 fine-tuning 학습 데이터의 example로 포함하였다. 또한, 학습 데이터셋의 출력 형태를 고려하여, 로봇의 수행 작업을 우리가 정한 기본 Tasks로 나눔으로써, 수많은 학습 데이터가 존재하지 않더라도 기존의 Task를 참조하여 LLM 모델이 적절한 작업 수행 결과를 도출할 수 있도록 하였다.

III. 실험 결과

로봇 작업 수행에 대해 LLM의 적용 가능성을 확인하기 위해, 경찰 로봇 시나리오를 가정하고 실험을 진행하였다. 실험에서 사용한 LLM은 OPENAI의 GPT-3.5-turbo 사전 학습 모델이며, fine-tuning 학습 데이터는 GPT-3.5-turbo의 Chat completions API를 활용하여 생성하였다. 실험 로봇은 터틀봇3 (Turtlebot3)을 사용하였다.

3.1. 프롬프트 엔지니어링을 통한 학습 데이터셋 생성

Fine-tuning을 위한 학습 데이터셋은 Chat completions API를 활용하여 생성하였다. Chat completions API는 리스트 형태의 messages를 입력으로 받으며, [표 1]은 messages의 example을 보여준다. messages 리스트의 각 요소는 role과 content로 구성되어 있다. Role이 system인 경우의 content는 모델에게 주는 사전 정보이고, user의 content는 모델에게 요청하는 내용이며, assistant의 content는 이전에 API로부터 응답받은 내용 또는 원하는 결과의 형태를 제공할 때 사용된다.

경찰 로봇 실험의 학습 데이터셋 생성을 위해 [표 1]의 (1), (2), (3)에 다음과 같은 값을 입력하여 프롬프트 엔지니어링하였다. (1)에는 [그림 1]의 instruction 정보를 주었고, (2)와 (3)에는 사용자의 지시문과 이에 대한 터틀봇3 답변 형태를 입력하였다. 여러 가지 지시문에 대해 프롬프트가 원하는 결과를 도출하게 되면 해당 프롬프트를 차용하여, 학습 데이터를 생성하였다. 또한, 테스트 단계에서 사용했던 예제도 학습 데이터에 포함하였다. 이와 같은 방법으로 생성한 학습 데이터셋은 모델 fine-tuning을 위해 사용되었다.

```
messages: [{"role": "system", "content": "(1) Marv is a factual chatbot that is also sarcastic."}, {"role": "user", "content": "(2)What's the capital of France?"}, {"role": "assistant", "content": "(3) Paris, as if everyone doesn't know that already."}]
```

[표 1] OPENAI에서 제공하는 API messages의 example

3.2. 실험 결과

본 논문에서 가정한 경찰 로봇에 복잡한 지시문을 주었을 때의 결과를 확인하였다. [그림 2]는 fine-tuning된 모델에 사전에 정의한 instruction을 주고, 사용자가 지시문을 입력한 예시를 보여준다. 지시문은 “터틀봇3가 A 지점으로 이동해 360 회전한 후 B 지점으로 가서 사진을 찍어. 그런 다음 C 지점으로 이동해 경비모드를 활성화하고, D 지점으로 이동해 순찰을 시작해. 마지막으로 A, B, C 지점을 차례대로 순찰한 후 보고를 전송해줘.”이며, 이 명령에 따른 결과는 [그림 3]과 같다. [그림 3]에서 보이는 바와 같이, 학습 데이터셋에 포함되지 않은 복잡한 지시문을 입력하여도, fine-tuning한 모델은 전체 작업

(task)을 판단하고, 이를 순차적인 세부 작업 (subtasks)으로 분해하였다. 또한, subtasks를 로봇이 수행할 수 있는 action으로 변환함을 확인할 수 있다.

```
response=client.chat.completions.create(
    model='ft:gpt-3.5-turbo-0125:personal:9z10c1x',
    messages=[
        {"role": "system", "content": instruction},
        {"role": "user", "content": "Turtlebot3가 A 지점으로 이동해#
360도 회전한 후 B 지점으로 가서 사진을 찍어.#
그런 다음 C 지점으로 이동해 경비 모드를 활성화하고,#
D 지점으로 이동해 순찰을 시작해. #
마지막으로, A, B, C 지점을 차례대로 순찰한 후 보고를 전송해줘."}]
response_content = response.choices[0].message.content
for line in response_content.split(','):
    print(line.strip())
```

[그림 2] Fine-tuning이 적용된 모델에 입력한 사용자의 지시문

```
{"task": "move
photograph
activate guard mode
patrol
and report"
"subtasks": [{"action": "go straight"
"to": "Position A"}
{"action": "turn"
"degree": "360"}
{"action": "go straight"
"to": "Position B"}
{"action": "take a photo"}
{"action": "go straight"
"to": "Position C"}
{"action": "activate guard mode"}
{"action": "go straight"
"to": "Position D"}
{"action": "patrol"
"between": ["Position A"
"Position B"
"Position C"]}
{"action": "transmit report"}
{"action": "stop"}]}
```

[그림 3] [그림 2]의 지시문에 대한 모델의 응답 결과

IV. 결론

본 논문에서는 LLM의 fine-tuning 기법 적용이 사용자의 지시를 세부 작업으로 분해하고, 로봇이 수행할 수 있는 동작 명령으로 변환할 수 있음을 확인하였다. 로봇의 위치 기반 경찰 시스템 시나리오를 통해 LLM의 적용 가능성을 볼 수 있었다.

ACKNOWLEDGMENT

이 논문은 정부(과학기술정보통신부) 재원으로 정보통신기획평가원의 지원 (No. 2021-0-00379, AI 시스템의 프라이버시 위험 분석 및 대응 기술 개발) 및 한국에너지공과대학교 KENTECH Research Grant (KRG 2022-01-013)의 지원을 받아 수행된 연구임.

참고 문헌

[1] M. Ahn, *et al.*, “Do as I can, not as I say: Grounding language in robotic affordances,” 2022, arXiv:2204.01691.
 [2] J. Wang, *et al.*, “Large language models for robotics: Opportunities, challenges, and perspectives,” 2024, arXiv:2401.04334.