

Instruction Tuning을 통한 한국어 언어 모델 문장 생성 제어

장진희¹, 서대룡², 전동현³, 강인호⁴, 나승훈⁵
전북대학교^{1,*5}, 네이버^{2,3,4}

{zhenxi, nash}@jbnu.ac.kr, {daeryong.seo, donghyeon.jeon, once.ihkang}@navercorp.com

Instruction Tuning for Controlled Text Generation in Korean Language Model

Jinhee Jang¹, Daeryong Seo², Donghyeon Jeon³, Inho Kang⁴, Seung-Hoon Na⁵
Jeonbuk National University^{1,*5}, NAVER Corporation^{2,3,4}

요약

대형 언어 모델(Large Language Model)은 방대한 데이터와 파라미터를 기반으로 문맥 이해에서 높은 성능을 달성하였지만, Human Alignment를 위한 문장 생성 제어 연구는 아직 활발한 도전 과제로 남아있다. 본 논문에서는 Instruction Tuning을 통한 문장 생성 제어 실험을 진행한다. 자연어 처리 도구를 사용하여 단일 혹은 다중 제약 조건을 포함하는 Instruction 데이터 셋을 자동으로 구축하고 한국어 언어 모델인 Polyglot-Ko 모델에 fine-tuning 하여 모델 생성이 제약 조건을 만족하는지 검증하였다. 실험 결과 4개의 제약 조건에 대해 평균 0.88의 accuracy를 보이며 효과적인 문장 생성 제어가 가능함을 확인하였다.

주제어: 문장 생성 제어(Controlled Text Generation), Instruction Tuning, 대형 언어 모델(Large Language Model)

1. 서론

대형 언어 모델(Large Language Model)은 크기는 수십억 개의 파라미터를 포함하며, 다양한 언어와 주제에 대한 광범위한 지식을 학습하였다. GPT-4[1], PaLM[2] 등은 높은 문맥 이해력을 기반으로 사람의 명령에 따르는 보조도구로서 실생활에 활발하게 적용되고 있다. 이에 따라 언어 모델의 출력이 사람의 의도나 사회적 정서 및 가치와 통한다는 의미인 Human Alignment 여부가 중시되며 제어 가능한 언어모델의 필요성이 대두되었다.[3, 4] 하지만 Black-box인 언어 모델의 특성상 모델의 출력을 제어하는 것은 쉽지 않다.

현재까지의 문장 생성 제어(Controlled Text Generation) 분야의 연구로는 특정 속성의 데이터(e.g., 긍정 혹은 부정)를 모델에 직접 fine-tuning 하거나 모델 파라미터를 업데이트하는 대신 디코딩 시에 확률 분포만을 수정하여 특정 토큰의 등장 확률을 증가시키는 등의 방식이 있었다. 하지만 전자의 경우 단일 속성에만 맞춰 모델을 학습해야 한다는 단점이 있으며, 후자의 경우 제어 accuracy와 fluency의 trade-off 문제, 추론(Inference) 시의 생성 지연 문제가 있다.

한편 대형 언어 모델의 In-Context Learning의 성능이 증명되면서[5] Zero-shot 성능을 향상하는 Instruction Tuning을 사용하여 Human Alignment를 개선하려는 연구가 계속되고 있다. 본 논문에서는 대형 언어 모델의 문장 생성 제어 Task의 일환으로써 Instruction Tuning 방식을 사용할 것을 제안한다. Instruction Tuning 방식은 이전 방식의 단점을 보완하며 장점이 뚜렷하다. 언어 모델의 많은 파라미터를 기반으로 추론한 토큰들의 확률분포를 훼손하지 않기 때문에 자연스러운 문장을

생성할 수 있으며, 특정 속성과 도메인에 제한을 두지 않는다.

Instruction Tuning을 통한 문장 생성 제어 성능을 확인하기 위해 간단한 자연어 처리 도구를 활용하여 4개의 제약 조건(단어, 어체, 감정, 길이)을 포함한 한국어 Instruction 데이터 셋을 총 70만 건가량 구축하여 실험을 진행하였다. 구축한 데이터 셋을 한국어로 학습된 대형 언어 모델인 Polyglot-Ko[6]에 fine-tuning 하여 제약 조건을 만족하는지 측정하였다. Instruction 내에 자연어 제약 문장을 추가하여 학습하는 것은 다른 문장 생성 제어 방식에 비해 간단한 방법임에도 불구하고, 단일 그리고 다중 제약 조건에 대해서도 잘 제어된 생성 양상을 보였다. 또한 학습 데이터 셋에 존재하지 않는 새로운 제약에 대한 Zero-shot 실험을 진행하여 문장 생성 제어 자체에 대한 성능이 향상되었는지 확인하였다.

2. 관련 연구

2.1 문장 생성 제어(Controlled Text Generation)

문장 생성 제어란 특정 제약 조건을 충족하도록 모델의 생성을 제어하는 연구이다. 기본적인 생성 작업에서는 주어진 입력에 따라 자연스러운 텍스트를 생성하는 것에 집중하였다면, 문장 생성 제어는 조건과 목적 등 구체적 방향성이 제시되는 한 단계 발전한 형태라고 볼 수 있다. 깊은 신경망을 가진 Black-box 모델을 해석하고 제어한다는 것은 한계가 존재하기 때문에 도전적인 연구 분야이기도 하다.[7]

생성된 타겟 문장이 $Y = \{y_1, y_2, \dots, y_n\}$ 이며, y_i 가 주어진 사전(Vocabulary)에 속할 때, 문장 생성 제어 task를 다음과

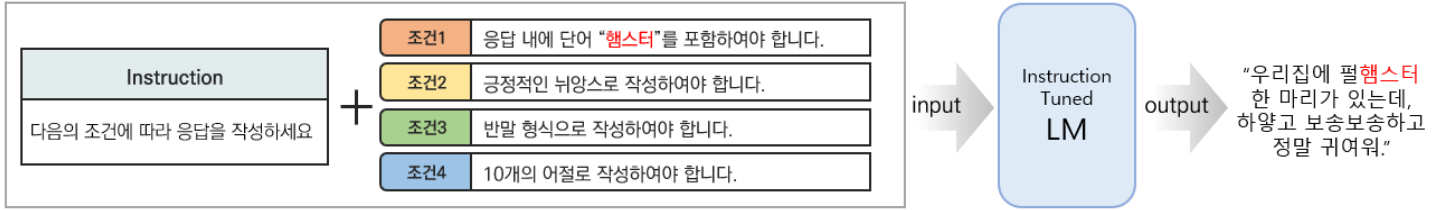


그림 1. 텍스트 생성 제어를 위한 Instruction 형태 : 단일 혹은 다중 제약 조건을 포함한 Instruction이 모델에 입력으로 들어가서 조건에 맞게 제어된 텍스트가 출력된다. 모델을 Instruction Tuning하기 위해 생성을 목표로 하는 label 텍스트에서 자동으로 제약 조건을 추출할 수 있다.3. 본 논문에서는 학습시에 단어, 감정, 어체 그리고 길이의 총 4가지 제어 조건을 주었지만, 조건은 갯수와 도메인에 상관없이 추가할 수 있다.

같은 수식으로 정의할 수 있다.

$$P(Y|C) = p(y_1, y_2, \dots, y_n|C) \quad (1)$$

여기서 C 는 제약 조건(constraint)을 나타낸다. 문장 생성 제어 task에서 타겟 문장 Y 는 제약 조건 C 에 따른 생성이다.

[8]에서는 모델을 학습하지 않고도 간단한 속성 분류기를 사용하여 모델의 출력을 제어하는 방법인 PPLM(Plug and Play Language Model)을 제시한다. 속성 분류기가 판별자(discriminator)로 작용하여 디코딩 시에 토큰의 확률 분포를 영향을 끼쳐 원하는 속성에 가까운 토큰을 출력하게 한다. 이는 단일 제약 조건에 대해서만 제어가 가능하다는 단점이 있다. 이와 비슷하게 판별자를 사용하여 토큰의 확률 분포를 수정하는 방법으로는 [9, 10]가 있다. COLM Decoding[11]은 단일 혹은 다중 제약 조건을 에너지 함수로 정의하고 Langevin dynamics를 이용한 gradient 기반 샘플링을 통해 제약에 맞는 문장을 생성한다. 모델을 따로 학습할 필요가 없다는 장점이 있지만, 생성 효율성 문제가 아직 존재한다.

2.2 Instruction Tuning

Instruction Tuning은 [12]에서 모델의 Zero-shot 성능을 높이기 위해 제시된 방법이다. 해당 논문에서 제안한 모델 FLAN(Finetuned LAnguage Net)은 60개 이상의 다른 Instruction에 fine-tuning 하여 사람의 다양한 명령을 이해하고 처리하는 능력을 크게 향상하였다. 최근 가장 주목 받은 연구 중 하나인 InstructGPT[13]에서도 강화학습을 통한 Instruction Tuning 방식을 사용하였다. 먼저 Instruction 데이터 셋을 구축하기 위해 labeler가 구축한 데이터 셋을 사용하여 GPT-3을 fine-tuning 한다. 그 후 학습된 GPT-3에 prompt를 넣어서 나온 결과물에 더 나은 답변을 고르도록 하는 Human Feedback을 더해 강화학습을 진행하여 Instruction Following에서 좋은 성능을 달성하였다. 사람이 직접 Instruction 데이터 셋을 구축하는 경우 양질의 데이터 셋을 구축할 수 있지만, 시간과 비용이 많이 든다는 단점이 있다. 이러한 단점을 돌파하기 위해 [14]에서는 Self-Instruct 프레임워크를 제안했다. 해당 프레임워크는

Seed Instruction 데이터 셋을 언어 모델에 내재된 지식으로 쉽게 증강할 수 있는 방법을 제시했다.

본 논문에서는 자동으로 데이터를 구축하고 다중 제약 조건에 대해서도 언어 모델이 비교적 손쉽게 제어 될 수 있도록 Instruction Tuning을 진행하는 것에 초점을 맞춘다.

3. Instruction 데이터 셋 자동 구축

본 섹션에서는 웹에서 수집한 여러 형태의 데이터 셋을 사용하여 제약 조건을 포함한 Instruction 데이터 셋으로 자동 구축하기 위한 과정을 서술하였다.

3.1 데이터 셋 수집 및 정제

수집한 데이터는 ①모두의 말뭉치의 온라인 개시 자료 말뭉치 2022, ②개인이 크롤링하여 github에 업로드한 네이버 쇼핑 리뷰 데이터 그리고 ③AI 허브의 요약문 및 레포트 생성 데이터이며 수량은 각 723,507건, 199,224건 그리고 160,002건이다. 수집 후에는 정제 작업을 거쳤다. ①,②에서 웹 크롤링 데이터가 다수 포함되었기 때문에, #(해시태그)가 과다 포함되거나 자소가 분리된 문장 등은 대부분 삭제하였고 총 785,043건으로 학습을 진행하였다.

①,②의 경우 label이 없는 문장으로 구성되며, ③은 요약 task를 데이터 셋으로 요약의 대상이 되는 문단과 요약된 타겟 label을 포함한 문장으로 구성된다. 해당 데이터 셋은 대분류의 요약 작업에 세부 분류의 제약 조건을 포함했을 때, task를 성공적으로 수행하는지 검증하기 위해 수집되었다.

3.2 제약 조건을 포함한 Instruction 생성

형태소 분석기, Word2Vec등 간단한 자연어 처리 도구를 사용하여 문장 내 제약 조건이 될 수 있는 정보를 자동으로 추출할 수 있다. 표 1에 제약 조건과 Instruction에 추가되는 조건문 예시 그리고 추출 방법을 정리하였다. 제약 조건은 단어(포함), 단어(불포함), 감정, 어체, 길이, 접두사, 접미사 총 7개로 세부 요건은 다음과 같다.

1. 단어(포함) : 특정 단어를 포함할 것

	제약 조건	조건문 예시	조건 추출 방법
1	단어(포함)	응답 내에 단어 “햄스터”, “강아지”를 포함하여야 합니다.	Mecab 형태소 분석기로 1~5개의 명사 추출
2	감정	긍정적인(부정적인) 뉘앙스로 작성하여야 합니다.	네이버 쇼핑 리뷰 데이터 셋의 긍/부정 label 활용
3	어체	반말(존댓말) 형식으로 작성하여야 합니다.	style classifier 활용
4	길이	응답은 최대 n+3에서 최소 n-3개의 어절로만 구성되어야 합니다.	len(label_sentence)
5	단어(불포함)	응답 내에 단어 “고양이”를 포함하지 않아야 합니다.	Mecab 형태소 분석기로 1~5개의 명사 추출
6	접두사	응답이 “우리집 햄스터는 하얗고 모송하고”로 시작하여야 합니다.	Mecab 형태소 분석기로 문장 분할 후 최후 3~5개 단어 추출 후 사전 학습된 Word2Vec모델로 유의어 변환
7	접미사	응답이 “정말 귀여워”라는 문장으로 끝나야 합니다.	Mecab 형태소 분석기로 문장 분할 후 최초 3~5개 단어 추출

표 1. 제약 조건과 조건 추출 방법

task 분류	Instruction	label
제약 task	다음의 조건에 따라 응답을 작성하세요. 조건1: 반말 형식으로 작성하여야 합니다. 조건2: 응답 내에 단어 “세무서”, “덕수동”, “어머니”, “상추쌈”, “뒤편”, “골목”을 포함하여야 합니다. 응답:	세무서 뒤편 골목에 위치한 덕수동 집밥에도 책자가져다 드린다. 여기는 그냥 집이다. 집에서 어머니가 해주시는 밥 처럼 그렇게 나온다. 내가 찾았을땐 수육과 상추쌈이 나왔었다. 이정도면 가격이...음... 그냥 정식 값이 었다.
제약+요약 task	다음의 문단을 조건에 따라 요약하세요. 그래서 일군의 학자들은 이상적인 공동체를 ‘사회의 품격이 높은 사회’라고 정의한다. 사회의 품격은 ‘공동체가 제공하는 사회, 경제, 문화생활에 참여하여 개인이 자신의 잠재력과 복지를 극대화할 수 있는 상생적 관계 맺기가 가능한 정도’를 의미한다. 달리 말해 사회의 품격은 ‘개인의 자기실현’과 ‘사회적 맥락에서 구성되는 다양한 집합적 정체성’ 간의 상시적인 긴장과 대항관계의 함수로서, 한편으로 개인의 생애사적 발전과 사회적 발전을 가르는 축(행위자와 구조)과, 다른 한편으로 조직화된 공식성과 친밀한 비공식적 세계를 가르는 축(체계와 생활 세계)을 교차하여 살펴볼 수 있다. 조건1: 요약 내에 단어 “사회”, “학자”, “품격”, “공동체”를 포함하여야 합니다. 응답:	어떤 학자들은 ‘사회의 품격이 높은 사회’가 이상적인 공동체라 한다. 사회의 품격은 공동체에서 개인 복지가 극대화되는 정도로, 즉 자기실현과 집합적 정체성 간의 관계이다.

표 2. 구축된 데이터 셋 예시

2. 감정 : 긍정 또는 부정의 감정 상태의 문장을 생성할 것
3. 어체 : 반말 또는 존댓말의 형태로 문장을 생성할 것
4. 길이 : n개의 단어로 문장을 구성할 것
5. 단어(불포함) : 특정 단어를 포함하지 않을 것
6. 접두사 : 특정 접두사로 문장을 시작할 것
7. 접미사 : 특정 접미사로 문장을 끝낼 것

본 실험에서는 수집한 데이터 셋을 label로 지정한다. 이때 요약 task가 포함된 데이터 셋의 경우 요약된 문장이 label이 된다. label에서 제약 조건들을 추출하고 해당 조건들을 포함한 조건문을 만들었다. 7개의 조건 중 1~4번은 Instruction Tuning 시 train과 validation에 사용되며, 나머지 5~7번은 학습되지 않은 새로운 제약 조건에 대해 Zero-shot 성능을 test 하기 위해 사용된다. train 데이터 셋에는 1~4개의 제약이 랜덤하게 포함된다. Instruction 문장은 다음과 같이 설정하였다.

- 제약 task : 다음의 조건에 따라 응답을 작성하세요.

- 제약+요약 task : 다음의 문단을 조건에 따라 요약하세요.
- 표 2에 실제로 구축된 Instruction 데이터 샘플을 볼 수 있다. 모델은 Instruction에서 요구하는 제약 조건에 상응하는 label을 학습하게 된다.

4. 실험

4.1 평가 지표

문장 생성 제어 분야에서는 accuracy와 fluency를 중점으로 성능을 평가한다. accuracy는 조건을 만족했는지 여부에 따라 달라지며 fluency는 생성한 문장이 얼마나 자연스럽게 유창한지를 판단한다.[7]

accuracy 평가를 위해 downstream task에 훈련된 분류기와 Python 문자열 메소드를 사용한다. 감정 제어 평가에는 ELECTRA 기반 한국어 언어 모델인 KcELECTRA를 한국어 이진 감성 분석 데이터 셋에 fine-tuning한 모델인 한국어 감성 분류

기¹를 사용했다. 또한 어체 제어 평가에는 BERT 기반 한국어 언어 모델인 KcBERT를 한국어 존댓말/반말 데이터 셋에 fine-tuning한 한국어 존댓말 반말 분류기²를 사용하였다. 그 외의 평가에는 모두 Python 문자열 메소드를 사용하여 점수를 측정하였다.

fluency 점수는 자동적인 방법과 수동적 방법으로 측정하여 확인했다. 자동적 방법에는 이전 연구와 같이[15] 모델의 난해성을 나타내는 지표인 perplexity를 사용했다. 모델이 생성한 결과물을 다시 언어 모델에게 입력값으로 주어 각 토큰의 등장 확률값을 통해 문장이 자연스러운지 여부를 간접적으로 확인할 수 있다.[16] perplexity를 측정할 모델은 실험한 모델과 동일한 Polyglot-Ko 5.8b이다. 수동적 방법으로는 정성 평가를 진행한다. [8]에서의 문장 평가 방법에 따라 문장이 자연스러운지 여부를 1~5점으로 평가하여 점수를 산출한다. 정성평가는 전체 생성 문장인 1,000건에 대해 한 명이 진행하였다.

4.2 모델 학습

Instruction Tuning을 위해 EleutherAI의 오픈 소스 한국어 언어 모델인 Polyglot-Ko[6]를 사용한다. Polyglot-Ko는 GPT-NeoX의 코드를 베이스로 1.2TB의 한국어 데이터 셋을 학습시킨 대형 언어 모델이다. 본 실험에서는 5.8b 모델로 full fine-tuning을 진행하였다. 실험 시 Optimizer는 AdamW, train epoch는 10, batch size는 1,024, 그리고 learning rate는 7e-5를 사용하였다.

5. 실험 결과

학습된 제약 조건과 학습되지 않은 제약 조건을 포함한 test 데이터 셋을 학습된 모델에 입력으로 주고 출력값과 성능을 확인한다. 전자의 경우 단일 혹은 다중 제약 조건을 포함하며 후자의 경우 단일 제약 조건만 포함하였다. test 데이터 셋 수량은 각 1,000건이며 제약 조건은 같은 같은 비율로 포함되었다.

5.1 학습된 제약 조건

	제약	제약+요약	조건별 평균	전체 평균
단어	0.9299	0.8789	0.9044	0.8892
감정	0.8755	-	0.8755	
어체	0.9531	-	0.9531	
길이	0.7621	0.8852	0.8236	

표 3. 학습된 제약 조건에 대한 accuracy 점수

¹https://huggingface.co/matthewburke/korean_sentiment

²<https://github.com/jongmin-oh/korean-formal-classifier>

표 3에서 제약 조건에 대한 accuracy 점수를 확인할 수 있다. 요약 task의 경우 신문, 보도자료, 간행물 등이 출처인 데이터 셋으로 감정과 어체의 길이를 특정하기 힘들기 때문에 단어(포함)와 길이 조건에 대해서만 평가하였다. 1~4개의 제약 조건을 포함하는 Instruction에 대해 전체 평균 0.88 이상의 성능을 보인다. 특히 어체 제약에서는 0.95가 넘는 높은 성능을 보였으며 요약 task와 제약 조건을 함께 Instruction으로 넣어줬을 경우에도 두 가지 제약에 대해 모두 0.87 이상의 성능을 보였다.

perplexity (↓)	human eval (↑)
25.46	0.94

표 4. 학습된 제약 조건에 대한 fluency 점수

perplexity와 정성 평가를 통해 fluency 점수를 산출한 결과 perplexity 점수는 25.46, 정성 평가 결과로는 0.94의 점수를 보였다. 실제 생성된 문장은 표 6의 generated text 컬럼에서 확인할 수 있다. 제약 조건만 있는 문장뿐만 아니라 요약 task가 추가된 경우에도 조건을 지키면서 자연스러운 문장을 생성하였다.

5.2 새로운 제약 조건

	단어(불포함)	접두사	접미사
Polyglot-Ko 5.8b (Instruction tuned)	0.5004	0.4639	0.1341
Polyglot-Ko 5.8b (base)	0.0467	0.0402	0.0216

표 5. 새로운 제약 조건에 대한 accuracy 점수

학습한 모델의 일반화 성능을 확인하기 위해 새로운 제약 조건에 대해서 Zero-shot 실험을 진행하였다. 표 5에서는 동일한 모델의 학습전 상태를 baseline으로 설정하고 학습 전-후의 Instruction Following 성능을 비교하였다. Instruction Tuning을 하지 않은 모델은 단일 제약 조건에 대해 모두 0.05가 안되는 낮은 accuracy를 보인다. 이에 비해 tuning을 완료한 모델은 더 높은 accuracy를 달성했다. 단어(불포함) 제약에서 0.5, 접두사 제약에서 0.46의 성능을 보이지만 접미사 제약에서는 0.13의 낮은 성능을 보였다. 두 가지 모델 모두 접미사에서 가장 부진한 점수를 보였기 때문에 해당 제약 조건이 다른 조건에 비해 모델 생성 제어에 미치는 영향력이 적다는 것을 알 수 있다.

6. 결론

현재 언어 모델은 방대한 데이터와 많은 수의 파라미터를 기반으로 문맥 이해력이 높아졌지만, Human Alignment를 위한

	Instruction	generated text
제약	다음의 조건에 따라 응답을 작성하세요. 조건1: 반말 형식으로 작성하여야 합니다. 조건2: 응답 내에 단어 “모성애”, “사투리”를 포함하여야 합니다. 조건3: 응답은 최소 8개에서 최대 18개의 어절로만 구성되어야 합니다. 응답:	모성애가 없는게 아니라 사투리가 심해서 못알아 듣은거 같은데?????
	다음의 조건에 따라 응답을 작성하세요. 조건1: 부정적인 뉘앙스로 작성하여야 합니다. 조건2: 존댓말 형식으로 작성하여야 합니다. 조건3: 응답 내에 단어 “할머니”를 포함하여야 합니다. 조건4: 응답은 최소 5개에서 최대 10개의 어절로만 구성되어야 합니다. 응답:	할머니택에 보내드렸는데 맛이 없다고 하시네요
제약+요약	다음의 문단을 조건에 따라 요약하세요. 본 연구의 결과를 살펴보면 또래와의 의사소통과 신뢰형성 수준이 높은 청소년이 교우관계나 교사관계와 같은 학교생활적응 수준도 높은 것으로 나타났다. 또한 또래와의 의사소통과 신뢰형성의 초기 수준은 시간이 지나도 학교생활적응에 영향을 미치는 것으로 나타났다. 다시 말해, 초기 또래관계를 잘 형성하는 청소년일수록 나중까지 교우관계나 교사와의 관계도 잘한다고 볼 수 있을 것이다. 또래관계를 잘 형성하는 청소년은 또한 학습활동에도 적극적이고 잘 적응하는 것으로 나타났다. 이러한 연구 결과는 청소년이 또래관계를 잘 형성하는 것이 학교생활적응에 있어 중요한 역할을 한다는 시사점을 갖는 것이다. 조건1: 요약 내에 단어 “또래”, “연구”, “의사소통”, “결과”를 포함하여야 합니다. 조건2: 요약은 최소 14개에서 최대 24개의 어절로만 구성되어야 합니다. 응답:	본 연구의 결과를 살펴보면 또래와의 의사소통과 신뢰형성 수준이 높은 청소년이 교우관계나 교사관계와 같은 학교생활적응 수준도 높은 것으로 나타났다.
	다음의 문단을 조건에 따라 요약하세요. 본 연구에서 다른 무기효과 데이터는 화력 무기체계와 표적 간의 교전에서 발생하는 피해효과를 다양한 척도로 산출한 데이터를 의미한다. 무기효과 데이터는 획득, 군수, 비용분석, 전력분석, 훈련/연습 등의 여러 분야에서와 같이 화력무기체계에 의한 표적의 피해정도를 정량적 요소로서 고려하여 분석/활용할 수 있는 다양한 영역에서 중요한 기초자료로 활용되고 있다. 하나의 예로서, 한미 탄약위원회(영문)에서 적정 탄약 소요량에 대한 합의점을 도출할 때도 각 탄약에 대한 무기효과 데이터가 기초데이터로 활용되었으며 최종 합의량 도출에도 결정적 영향요소로 작용하였다. 조건1: 요약 내에 단어 “선주민”, “영향”, “사업”을 포함하여야 합니다. 조건2: 요약은 최소 12개에서 최대 22개의 어절로만 구성되어야 합니다. 응답:	영향을 주는 사업 수행시, 특히 이주, 선주민 토지에 위험물질 저장이나 처리를 할 때 사전동의를 구하도록 하고 있다.

표 6. 생성한 문장

문장 생성 제어 연구는 아직 활발한 도전 과제로 남아있다. 본 논문에서는 문장 생성 제어 작업을 위해 Instruction Tuning을 사용하여 실험을 진행하였다. 단일 혹은 다중 제약을 포함한 Instruction 데이터를 간단한 자연어 처리 도구로 자동 구축하여 한국어 대형 언어 모델인 Polyglot-Ko 5.8b에 학습한 결과 단어, 감정, 어체, 길이 제약 task에서 평균 0.88의 성능을 보였다. 학습하지 않은 새로운 제약 task에서도 원 모델보다 10% 개선된 Zero-shot 성능을 보였다. 이로써 Instruction Tuning이 문장 생성 제어의 효과적인 방법 중 하나가 될 수 있다는 사실을 확인하였다.

6.1 향후 연구

Instruction Tuning으로 언어 모델을 제어하기 위해서는 모델이 실제로 Instruction을 따르는 방법을 학습한 것인지, 혹은 그저 자연어 문장 자체의 특성만 학습한 것인지에 대한 고찰이 필요하다. 향후 연구에서는 학습에 쓰인 prompt 디자인을 다채롭게 변경하며 실험을 진행하여 대형 언어 모델의 Instruction Following 능력에 대한 검증은 추가할 예정이다. 학습시의 효율성을 역시 고려할 필요가 있다. 대형 언어 모델을 Adaptor[17]나 Quantization[18]등의 기법을 통해 Instruction Tuning하여 성능을 평가하고, 제약 문장으로 인해 prompt가 길어지는 만큼 Long In-Context Learning과도 관련지어 업데이트 하지 않고 제약 문장을 파라미터화 하여 내제화 할 수 있는 방법에 대한 연구 역시 진행할 예정이다.

참고문헌

- [1] OpenAI, “Gpt-4 technical report,” 2023.
- [2] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel, “Palm: Scaling language modeling with pathways,” 2022.
- [3] D. Hendrycks, C. Burns, S. Basart, A. Critch, J. Li, D. Song, and J. Steinhardt, “Aligning ai with shared human values,” 2023.
- [4] S. L. Blodgett, S. Barocas, H. Daumé III, and H. Wallach, “Language (technology) is power: A critical survey of “bias” in NLP,” *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5454–5476, Jul. 2020. [Online]. Available: <https://aclanthology.org/2020.acl-main.485>
- [5] D. Dai, Y. Sun, L. Dong, Y. Hao, S. Ma, Z. Sui, and F. Wei, “Why can gpt learn in-context? language models implicitly perform gradient descent as meta-optimizers,” 2023.
- [6] H. Ko, K. Yang, M. Ryu, T. Choi, S. Yang, J. Hyun, S. Park, and K. Park, “A technical report for polyglotko: Open-source large-scale korean language models,” 2023.
- [7] H. Zhang, H. Song, S. Li, M. Zhou, and D. Song, “A survey of controllable text generation using transformer-based pre-trained language models,” 2023.
- [8] S. Dathathri, A. Madotto, J. Lan, J. Hung, E. Frank, P. Molino, J. Yosinski, and R. Liu, “Plug and play language models: A simple approach to controlled text generation,” *CoRR*, Vol. abs/1912.02164, 2019. [Online]. Available: <http://arxiv.org/abs/1912.02164>
- [9] B. Krause, A. D. Gotmare, B. McCann, N. S. Keskar, S. Joty, R. Socher, and N. F. Rajani, “Gedi: Generative discriminator guided sequence generation,” 2020.
- [10] K. Yang and D. Klein, “FUDGE: Controlled text generation with future discriminators,” *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2021. [Online]. Available: <https://doi.org/10.18653/2021.naacl-main.276>
- [11] L. Qin, S. Welleck, D. Khashabi, and Y. Choi, “Cold decoding: Energy-based constrained text generation with langevin dynamics,” 2022.
- [12] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le, “Finetuned language models are zero-shot learners,” 2022.
- [13] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, and R. Lowe, “Training language models to follow instructions with human feedback,” 2022.
- [14] Y. Wang, Y. Kordi, S. Mishra, A. Liu, N. A. Smith, D. Khashabi, and H. Hajishirzi, “Self-instruct: Aligning language models with self-generated instructions,” 2023.
- [15] W. Zhou, Y. E. Jiang, E. Wilcox, R. Cotterell, and M. Sachan, “Controlled text generation with natural language instructions,” 2023.
- [16] C. Meister and R. Cotterell, “Language model evaluation beyond perplexity,” 2021.
- [17] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, “Lora: Low-rank adaptation of large language models,” 2021.
- [18] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, “Qlora: Efficient finetuning of quantized llms,” 2023.