

# 거대 언어 모델의 내재된 지식을 활용한 질의 응답 방법

심묘섭<sup>o</sup>, 민경구, 박민준, 최주영, 정해민, 최정규

LG AI 연구원

{myoseop.sim, kyungkoo.min, minjun.park, jooyoung.choi, haemin.jung, stanleyjk.choi}@lgrsearch.ai

## Question Answering that leverage the inherent knowledge of large language models

Myoseop Sim<sup>o</sup>, Kyungkoo Min, Minjun Park, Jooyoung Choi, Haemin Jung, Stanley Jungkyu Choi  
LG AI Research

### 요약

최근에는 질의응답(Question Answering, QA) 분야에서 거대 언어 모델(Large Language Models, LLMs)의 파라미터에 내재된 지식을 활용하는 방식이 활발히 연구되고 있다. Open Domain QA(ODQA) 분야에서는 기존에 정보 검색기(retriever)-독해기(reader) 파이프라인이 주로 사용되었으나, 최근에는 거대 언어 모델이 독해 뿐만 아니라 정보 검색기의 역할까지 대신하고 있다.

본 논문에서는 거대 언어 모델의 내재된 지식을 사용해서 질의 응답에 활용하는 방법을 제안한다. 질문에 대해 답변을 하기 전에 질문과 관련된 구절을 생성하고, 이를 바탕으로 질문에 대한 답변을 생성하는 방식이다.

이 방법은 Closed-Book QA 분야에서 기존 프롬프팅 방법 대비 우수한 성능을 보여주며, 이를 통해 대형 언어 모델에 내재된 지식을 활용하여 질의 응답 능력을 향상시킬 수 있음을 입증한다.

주제어: 거대 언어 모델, 질의 응답, 프롬프트

### 1. 서론

거대 언어 모델[1-3]은 인공 지능의 다양한 분야에서 뛰어난 성능을 보여주고 있다. 특히, 질의 응답(Question Answering, QA) 분야[4]에서도 거대 언어 모델을 활용하기 위해 다양한 연구가 진행되고 있다.

기존 질의 응답 분야는 크게 두 가지 유형으로 나눌 수 있다. 하나는 “Open Domain QA”로, 모델이 질문에 대한 정답을 얻기 위해 검색이 가능한 문서 또는 지식 베이스 등 외부 정보에 접근할 수 있는 방식이다. 다른 하나는 “Closed-Book QA”로 모델은 미리 주어진 지식 범위 내에서만 답을 찾아야 하는 방식이다.

Open Domain QA에서는 일반적으로 검색 엔진과 같은 정보 검색기(Retriever)와 문서를 이해하고 정답을 찾기 위한 독해기(Reader)로 파이프라인이 구성된다. 최근에는 거대 언어 모델이 독해기로 사용되고 있으며, 더 나아가 정보 검색기의 역할까지 대체하는 경우도 있다.

Close Domain QA는 거대 언어 모델의 활용이 더욱 중요한 분야다. 외부 정보 검색이 제한되는 경우, 답은 모델의 파라미터에 내재되어 있는 지식[5]에만 의존해야 하기 때문이다. 최근 연구에서는 언어 모델이 내부에 지식을 암기하거나 저장하고 있고, 이는 모델의 파라미터가 클수록 많은 지식을 저장할 수 있음이 확인되었다.[6-7]

거대 언어 모델은 문맥 내 학습(In-Context-Learning)을 통해 질의 응답 분야에서 인상적인 성능 향상을 달성했다. 입력 프롬프트에 질의-응답 예시 쌍을 추가하는

직접 프롬프팅(Direct Prompting)을 통해 더욱 사실적인 답변을 생성할 수 있다. 또한 질의-응답 예시 쌍을 고정해 놓지 않고, 질문에 따라 유사한 예시를 외부 문서에서 검색해온 문장들로 구성하는 검색 증강 프롬프팅 방식을 통해 더욱 사실에 기반한 답변을 할 수도 있다. 그림 1.[8-10]

본 연구에서는 외부에서는 문서를 검색할 수 없는 Closed-Book QA에서, 모델의 내부 지식을 활용해서 관련 문서를 생성하고, 이를 바탕으로 답변을 생성하는 방법을 제안한다.

해당 방법은 기존 프롬프팅 방법 대비 우수한 성능을 보여주며, 이를 통해 거대 언어 모델에 내재된 지식을 활용하여 질의 응답 능력을 향상시킬 수 있음을 입증한다.

### 2. 관련 연구

#### 2.1 Open Domain Question Answering

Open Domain QA 분야에서는 주로 검색 후 독해를 하는 방법을 사용한다. 질문이 주어지면 검색기 모델은 위키 피디아와 같은 외부 문서에서 질문과 관련 있는 문서를 가져온다.

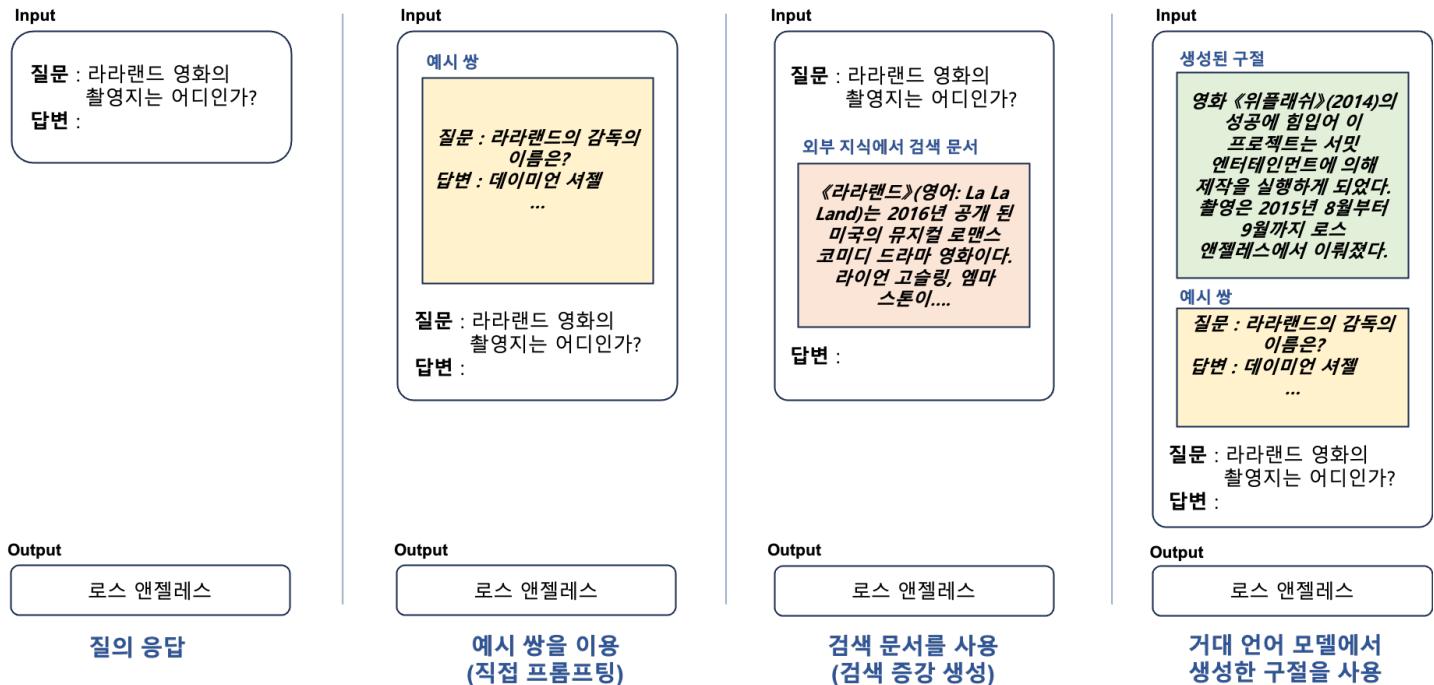


그림 1. 질의 응답에서 거대 언어 모델 사용 예시

그런 다음 독해 모델이 문서를 이해하고 정답을 예측한다. 검색 모델의 경우 초기에는 BM25와 같은 검색기 [11]가 사용되었고, 문맥 벡터를 활용한 DPR[12] 모델이 사용되면서부터 많은 성능 향상이 있었다. 독해 모델의 경우, [13-14]에서 성능 향상을 위한 연구가 진행되었다. 본 연구에서는 검색기 대신 거대 언어 모델의 파라미터에 내재된 지식을 검색해오는 방식을 대안으로 제시한다.

## 2.2 문맥 내 학습

거대 언어 모델[1]은 특정 문제에 대한 사전 학습을 받지 않고 미세 조정 없이, 풀고자 하는 문제에 대한 예시 쌍을 입력해주면 이를 학습하여 문제 해결 능력이 높아지는 문맥 내 학습이 가능하다. 이는 거대 언어 모델의 파라미터 크기가 클수록 성능이 높아진다. 최근 연구 [9]는 모델의 사이즈를 줄이고 미세 조정을 하거나 또는 입력 프롬프트에 예사 쌍을 검색 결과로 증강시키는 방법을 통해 성능을 더욱 향상시킨다. 본 연구에서는 문맥 내 학습을 위해 ‘검색 결과’로 프롬프트를 증강시키는 대신, ‘거대 언어모델에 내재된 지식에서 생성된 결과’로 프롬프트를 증강시키는 방법을 제안한다.

## 2.3 지식 기반 거대 언어 모델

언어 모델은 학습 과정에서 언어적 표현 뿐만 아니라 다양한 지식을 포함하여 학습한다.[1] 언어 모델에는 사실적 지식과 상식이 포함되어 있고[5], 이는 언어 모델의 사전학습 단계에 파라미터에 저장된다.[15] 따라서 거대 언어 모델은 제로샷을 이용해 문제를 풀 수 있는 능력이 있으며 의미 있는 성능을 달성했다.

## 2.4 거대 언어 모델의 문장 생성

최근 질의 응답 분야에서는 거대 언어 모델의 내재된 지식을 활용해 검색기의 역할을 대체하는 연구가 진행되었다. [16]에서는 검색 결과 대신 거대 언어 모델의 문서 생성 결과를 사용해 질의 응답 분야에서 기준보다 우수한 성능을 보였다. 또한 프롬프트에 사용된 질의-응답 예시에 정답의 풀이 과정을 포함함으로써 모델의 추론 성능을 향상시키는 방법도 연구되었다.[17]

거대 언어 모델이 질문에 답하기 전에, 질문과 관련된 문서를 생성하고 이를 바탕으로 정답을 도출하는 방법 [18]도 연구되었다. 이는 본 연구와 가장 유사한 방식이지만, 우리는 더 나아가 다양한 관련 문서 생성을 위한 프롬프트 구성 방안을 함께 제시한다.

## 3. 제안 방법

해당 섹션에서는 질의 응답 분야에서 거대 언어 모델의 지식을 활용하기 위한 방법에 대해 설명한다. 먼저 질문과 관련된 구절을 생성하여 프롬프트로 구성하고, 이를 바탕으로 답변을 생성한다. 이는 인간이 문제를 풀 때 관련 내용을 먼저 떠올린 후에 질문에 대한 정답을 찾는 행동과 유사한 부분이 있다.

### 질문: 바그너는 고테의 파우스트를 읽고 무엇을 쓰고자 했는가?  
위 질문에 대한 답변은 다음 위키피디아 문장에서 찾을 수 있습니다.

### 답변: 1839년 바그너는 고테의 파우스트를 처음 읽고 그 내용에 마음이 끌려... (중략)

### 질문: 1939년 9월 1일, 전함 술레스비히-홀슈타인의 일제 포격으로 시작된 폴란드 침공으로 발발한 전쟁은?  
위 질문에 대한 답변은 다음 위키피디아 문장에서 찾을 수 있습니다.

### 답변: 1939년 9월 1일 현지 시간 4시 48분, 전함 술레스비히-홀슈타인이 폴란드 ... (중략)

### 질문: 초우라늄 원소가 존재할 가능성을 처음 제안한 사람은?  
위 질문에 대한 답변은 다음 위키피디아 문장에서 찾을 수 있습니다.

### 답변: 초우라늄 원소의 존재는 1934년, 엔리코 페르미가 그의 연구를 바탕으로 ... (중략)

### 질문: 대스몬드 도스는 교회의 창문을 갈던 중 차에 깔린 남자를 발견하고 그를 지혈하기 위해 무엇을 사용했는가?  
위 질문에 대한 답변은 다음 위키피디아 문장에서 찾을 수 있습니다.

### 답변: 대스몬드 도스는 교회의 창문을 갈던 중 차에 깔린 남자를 발견하고 ... (중략)

그림 2. 관련 구문 생성을 위한 프롬프트 예시

### 3.1 관련 구절 생성을 통한 증강

질문에 대한 관련 구절을 생성하기 위해 예시 쌍을 포함한 프롬프트를 그림 2와 같이 구성한다. 이를 통해 생성된 구절은 거대 언어 모델의 내부 지식으로부터 문장을 검색해 온 것과 유사한 의미를 지닌다. 검색을 통한 프롬프트 증강 방법과 유사하게, 정답을 찾기 위한 프롬프트는 그림 3과 같이 구성한다. 질의-응답 예시 앞에 생성된 구절을 추가해 정답을 도출한다.

**질문** **답 예시 쌍**

질문: 한종우 교수가 김 전 대통령이 가장 큰 공헌을 한 분이라고 평가한 것은?  
정답: 민주화

질문: 한교공회에서는 악질적 행위를 벌인 누구를 사살했는가?  
정답: 백토벌

질문: 김규식이 김구의 장례식과 관련하여 장례위원회 측과 정부측을 조율하여 결정한 장례 방식은?  
정답: 국민장

질문: 라라랜드 영화의 촬영지는 어디인가?  
정답:

그림 3. 정답 생성을 위한 프롬프트 예시

### 3.2 관련 구절 다양화 방법

[18]의 관련 구절 생성 방법은 동일한 프롬프트를 사용해 독립적으로 여러 번 생성하는 방식이다. 이유는 생성되는 구절들이 여러 지식을 기반으로 다양하게 생성되기를 기대하기 때문이다. 하지만 생성된 구절들은 기대와 다르게 대부분이 유사한 내

용과 형식을 가진다. 따라서 이 방식은 거대 언어 모델의 다양한 지식을 활용했다고 보기 어렵다.

관련 구절의 다양성을 확보하기 위해 본 연구에서는 프롬프트에 사용되는 예시 쌍을 매번 변경하는 방식을 사용했다. 먼저 데이터의 학습셋에서 질문-관련문서 쌍을 샘플링해 150개의 예시 쌍을 만든다. 그리고 관련 구절 생성을 위한 프롬프트를 구성할 때 무작위로 4쌍의 예시를 샘플링하여 사용한다. 이와 같은 방법을 통해 관련 구절을 더욱 다양하게 생성할 수 있다.

### 3.3 정답 일관성을 위한 방법



그림 4. 정답 일관성 향상을 위한 투표

거대 언어 모델에 의해 생성된 구절은 정답을 포함하고 있는지 여부에 따라 정답을 대답할 수 있는지 없는지가 결정된다. 따라서 정답에 대한 일관성을 높이기 위해 하나의 질문에 대해 관련 구절을 최대 20번 독립적으로 생성한다. 그리고 그림 4와 같이 생성된 20개의 정답 중 투표(voting)을 통해 가장 빈도수가 많은 답을 정답으로 선정하는 방법을 사용한다.

### 4. 실험 및 결과

#### 4.1 데이터셋 및 모델

실험은 한국어 질의 응답 데이터셋 KorQuAD1.0[19] 데이터셋을 사용했다. KorQuAD는 Machine Reading Comprehension을 위한 질의 응답 데이터셋이다. 위키피디아의 문서를 사용해 만들어졌으며, “context”를 기반으로 관련 질의-응답 쌍으로 구성되어 있다.

평가를 위해 Dev Set 약 5700개의 질의 응답 중 1024개의 질의-응답 쌍을 샘플링하여 사용했다.

관련 구절을 생성하기 위한 프롬프트 구성에는 KorQuAD 데이터셋의 Train Set을 사용했다. “context”와 “question” 쌍을 150쌍 구성했고, 프롬프트 구성 시에는 4쌍씩 샘플링하여 사용했다.

실험에 사용한 거대 언어 모델은 GPT-3.5-Turbo 모델이다. 이 모델은 text-davinci-003을 기반으로 서비스되고 있으며 약 175B의 파라미터를 가진다.

#### 4.2 실험 및 결과 분석

KorQuAD 데이터셋 1024개를 대상으로 질의 응답 실험을 진행했다. 성능 측정은 F1 점수와 EM 점수를 측정했다. 결과는 표 1에서와 같이 예시 쌍을 입력에 넣어주는 직접 프롬프팅 방법 대비 우수한 성능을 보여준다. 결과 옆 팔호 안의 숫자는 정답 선정을 위한 투표에 몇 개의 관련 구절을 사용했는지를 표시한다. 따라서 표 1에 나

온 최종 성능은 하나의 질문에 대해 14개의 관련 구절을 생성하고, 이를 적용해 나온 정답 14개 중 투표를 통해 최종 정답을 도출한 결과다.

질문만 입력하는 방식은 거대 언어 모델이 답변의 형식에 대해 가이드를 받지 않았으므로 대부분 긴 문장으로 답변하는 경향이 있다. 따라서 EM 점수가 낮게 측정되었다.

표 1. 생성 구절 적용 시 결과 비교

	F1 Score	EM Score
질문만 입력	8.21	0
직접 프롬프팅 (예제 쌍)	21.44	5.17
구절 생성 프롬프팅 (관련 구문 + 예제 쌍)	27.13 (14)	7.61 (14)

거대 언어 모델의 답변 특성으로 인해, 긴 답변 문장에 정답의 포함 여부를 판단하기 위해 의미적으로 정답인 경우가 있는지를 측정해 보았다. 표 2에서와 같이 SM(Semantic Score)라는 평가 방식을 설정했다. 이는 답변 문장 안에 정답을 포함하고 의미적으로 정답이라고 여기고 1, 없으면 0으로 점수를 계산했다.

표 2. 생성 구절 적용 시 결과 비교. Semantic Score

	SM Score
질문만 입력	10.45
직접 프롬프팅 (예제 쌍)	6.37
구절 생성 프롬프팅 (관련 구절 + 예제 쌍)	11.32 (14)

질문만 입력하는 방식의 경우는 긴 답변으로 인해 F1 점수와 EM 점수가 낮았지만, 답변 안에 정답을 포함하고 있을 확률은 올라가기 때문에 SM 점수에서는 기존 대비 높은 성능을 내는 것을 알 수 있다. 반면, 직접 프롬프팅 방식은 예제 쌍으로 인해 짧은 답변을 해야 한다는 것을 알고 있지만 실제 오답을 말하는 경우가 많다는 것을 알 수 있다.

우리가 제안한 구절 생성 프롬프팅 방법은 답변 형식에 맞게 짧은 답변을 하고, 직접 프롬프팅에 비해 높은 정답율을 내는 것을 알 수 있다.

그림 5에서는 관련 구절을 최대 20번까지 생성하며, 투표를 통한 정답을 생성하는 방식의 성능 추이를 살펴보았다. 질문과 관련된 구절이 한번만 추가 되더라도 성능이 오르는 것을 볼 수 있으며, 약 10회 생성까지 성능이 오르는 것을 볼 수 있다. 이는 생성한 관련 구절이 정답과 관련이 없는 내용이라 하더라도 투표 방식을 통해 정답의 일관성을 높여준다는 것을 입증하는 결과이다.



그림 5. 구절 생성 횟수별 성능 추이



그림 6. 관련 구절 다양화를 위한 랜덤 프롬프트 적용 결과

그림 6에서는 다양한 관련 구절 생성을 위한 방법을 적용한 결과를 비교한다. 예시는 “임종석이 여의도 농민폭력 시위를 주도한 혐의로 지명수배된 연도는?”이라는 질문에 대해 20회씩 관련 구절을 생성한 결과이다. 거대 언어 모델이 질문과 관련된 지식을 통해 다양한 구절을 생성하기를 바라지만, 그림 6의 위 예시에서 볼 수 있듯이 매번 유사한 내용과 형태의 구절을 생성한다. 이를 개선하기 위해 프롬프트의 예시 쌍을 매번 무작위로 선정해 구성하는 방식을 적용하면 그림 6의 아래와 같이 보다 다양한 형태의 구절을 생성하는 것을 확인할 수 있다.

## 5. 결론

본 논문에서는 외부 지식에 접근이 없는 환경에서 질

의-응답을 하기 위한 거대 언어 모델 활용 방법을 제안했다. 거대 언어 모델이 질문에 답을 하기 위해서 관련 구절을 미리 한번 생각해보는 것이 도움을 준다는 가설을 확인할 수 있었다. 이를 위해 다양한 관련 구절 생성 방법, 생성된 구절이 정답과 관련이 없더라도 정답에 일관성을 높이기 위한 방법 등을 제안했다. KorQuAD 데이터셋에 대해서 성능 향상을 통해 해당 방법의 유효성을 입증했다.

하지만, 이 방법은 몇가지 한계점이 있다. 먼저, 거대 모델 학습 시 포함되지 않은 특수한 도메인 지식에 대해서는 해당 방법을 활용할 수 없다. 또한 최신 정보에 관한 질문에 대해서도 마찬가지로 해당 방법을 활용할 수 없다. 거대 언어 모델을 재학습 시키거나 미세 조정을 하기 위해서는 많은 자원과 비용이 들기 때문에 내재된 지식에 대한 업데이트가 어렵다.

향후 연구에서는 파라미터 수가 조금 더 작은 모델을 활용한 방법 검증, 특정 도메인 지식을 거대 언어 모델에 주입할 수 있는 방법 등에 대해 연구하며 상기된 한계점을 개선할 수 있는 연구를 계속할 예정이다.

## 참고문헌

- [1] Brown, Tom, et al. "Language models are few-shot learners." *Advances in neural information processing systems* 33. 1877–1901. 2020.
- [2] Rae, Jack W., et al. "Scaling language models: Methods, analysis & insights from training gopher." *arXiv preprint arXiv:2112.11446*. 2021.
- [3] Chowdhery, Aakanksha, et al. "Palm: Scaling language modeling with pathways." *arXiv preprint arXiv:2204.02311*. 2022.
- [4] Chen, Danqi, et al. "Reading wikipedia to answer open-domain questions." *arXiv preprint arXiv:1704.00051*. 2017.
- [5] Petroni, Fabio, et al. "Language models as knowledge bases?." *arXiv preprint arXiv:1909.01066*. 2019.
- [6] Roberts, Adam, Colin Raffel, and Noam Shazeer. "How much knowledge can you pack into the parameters of a language model?." *arXiv preprint arXiv:2002.08910*. 2020.
- [7] Talmor, Alon, et al. "oLMpics-on what language model pre-training captures." *Transactions of the Association for Computational Linguistics* 8. 743–758. 2020
- [8] Guu, Kelvin, et al. "Retrieval augmented language model pre-training." *International conference on machine learning*. PMLR, 2020.
- [9] Izacard, Gautier, et al. "Few-shot learning with retrieval augmented language models." *arXiv preprint arXiv:2208.03299*. 2022.
- [10] Lewis, Patrick, et al. "Retrieval-augmented generation for knowledge-intensive nlp tasks." *Advances in Neural Information Processing Systems* 33. 9459–9474. 2020.
- [11] Chen, Danqi, et al. "Reading wikipedia to answer open-domain questions." *arXiv preprint arXiv:1704.00051*. 2017.
- [12] Karpukhin, Vladimir, et al. "Dense passage retrieval for open-domain question answering." *arXiv preprint arXiv:2004.04906*. 2020.
- [13] Izacard, Gautier, and Edouard Grave. "Leveraging passage retrieval with generative models for open domain question answering." *arXiv preprint arXiv:2007.01282*. 2020.
- [14] Cheng, Hao, et al. "UnitedQA: A hybrid approach for open domain question answering." *arXiv preprint arXiv:2101.00178*. 2021.
- [15] Poerner, Nina, Ulli Waltinger, and Hinrich Schütze. "Bert is not a knowledge base (yet): Factual knowledge vs. name-based reasoning in unsupervised qa." *arXiv preprint arXiv:1911.03681* 3. 2019.
- [16] Yu, Wenhao, et al. "Generate rather than retrieve: Large language models are strong context generators." *arXiv preprint arXiv:2209.10063*. 2022.
- [17] Wei, Jason, et al. "Chain-of-thought prompting elicits reasoning in large language models." *Advances in Neural Information Processing Systems* 35. 24824–24837. 2022.
- [18] Sun, Zhiqing, et al. "Recitation-augmented language models." *arXiv preprint arXiv:2210.01296*. 2022.
- [19] 임승영, 김명지, and 이주열. "KorQuAD: 기계독해를 위한 한국어 질의응답 데이터셋." *한국정보과학회 학술발표논문집*. 539–541. 2018