

효율적 복습을 위한 모의고사 자동제작 사이트 개발

자연어처리와 교육 중간발표
심리학과 송상록

목차

- (1) 진행 상황 보고
- (2) 작동 영상 시청
- (3) 구현 원리 설명
- (4) 앞으로 할 일 정리

1. 진행 상황 보고

- 웹사이트의 핵심 기능은 구현 완료 (첨부파일 업로드 기능만 아직 구현하지 못함)
- 중간발표는 웹사이트 개발에 사용된 패키지 및 기술에 대한 설명 위주로 진행
- 프로젝트 관련 선행 연구, 사용성 평가, 의의와 한계, 느낀 점 등은 기말발표 때 다룰 예정

태스크 정의

본 프로젝트에서 제작하는 웹사이트는 다음 기능을 수행할 수 있어야 한다.

~~(1) 수업 내용을 담은 text data를 입력받으면, 자동으로 객관식 문제를 제작한다. text data는 .docx(word)나 .pdf 파일로 업로드할 수도 있고, 학습자가 직접 타이핑해 입력할 수도 있다.~~

~~(2) 제작한 객관식 문제를 웹사이트상에서 풀고 채점할 수 있는 기능을 제공한다. 문제풀이 UI는 흔히 운전면허 이론 시험, CBT 자격증 시험에서 볼 수 있는 형태와 비슷하게 구현할 계획이다.~~

~~(3) 틀린 문제에 대한 구체적인 풀이를, 문제집 해설지와 유사한 형태로 제공한다.~~

그 외에도 개발 상황에 따라 다른 기능이 추가될 수도 있다.

2. 작동 영상 시청

- 소설 <동백꽃> 전문을 입력해 생성된 문제 풀어보기
- <https://youtu.be/RqFRrDG1Cw0>

모의고사 자동제작 서비스

내용 입력

동백꽃

2
오늘도 또 우리 수탉이 막 쫓기었다. 내가 점심을 먹고 나무를 하러 갈 양으로 나올 때이었다. 산으로 올라서려니까 등뒤에서 푸드득 푸드득 하고 닭의 헛소리가 야단이다. 깜짝 놀라서 고개를 돌려 보니 아니나 다를까 두 놈이 또 얼리었다.

3
점순네 수탉(대강이가 크고 똑 오소리같이 실팍하게 생긴 놈)이 덩저리 작은 우리 수탉을 함부로 해내는 것이다. 그것도 그냥 해내는 것이 아니라 푸드득하고 면두를 찢고 물리었다가 좀 사이를 두고 푸드득하고 모가지를 찢었다. 이렇게 멧을 부려 가며 여지없이 닭아 놓는다. 그러면 이 못생긴 것은 쪼일 적마다 주둥이로 땅을 받으며 그 비명이 킁, 킁, 할뿐이다. 물론 미쳐 아물지도 않은 면두를 또 찢으며 붉은 선혈은 똑똑 떨어진다. 이걸 가만히 내려다보자

언어 선택 (영어가 더 정확합니다.)

☒ 한국어
☐ English

생성할 문제 수 설정 (인공지능의 한계로 더 적은 수의 문제가 생성될 수 있음)

5

1 20

모의고사 생성하기

3. 구현 원리 설명 (Langchain)

- Prompt Engineering: 다양한 프로그램이나 연구 주제에 언어 모델을 효율적으로 사용할 수 있도록, 프롬프트(명령문)를 개발해 최적화하는 분야
- Langchain: Prompt Engineering을 가능하게끔 하는 Python 패키지
- 본 프로젝트에선 Langchain을 통해 OpenAI의 GPT-4에 적절한 명령을 내려 웹사이트 구현
- (GPT에 대해선 강의계획서상 8차시 '트랜스포머'에서 더 다루는 것 같습니다)



3. 구현 원리 설명 (Langchain)

- `create_structured_output_runnable(output_schema, llm, prompt, ...)`
- 거대언어모형(llm)에게 명령을 지시하고(prompt), 답변을 원하는 형태(output_schema)로 출력

```
chain = create_structured_output_runnable(Questionmaker, model, prompt)
```

```
class SingleQuestion(BaseModel):
    question: str = Field(description="Multiple choice questions")
    selection: List[str] = Field(description="List of selectable answers for each question")
    correct: str = Field(description="Correct answer for question")
    explanation: str = Field(description="Explanation on why the answer is correct for each question")
class Questionmaker(BaseModel):
    questions: List[SingleQuestion] = Field(
        description="List of {num_questions} questions")
```

output_schema

```
model = ChatOpenAI(
    model="gpt-4-0125-preview",
    temperature=0.2,
    max_tokens=None)
```

llm

```
template = '''
You are a study assistant which must generate unique
{num_questions} multiple-choice questions for students.\n
By using [INPUT_DATA], make multiple choice questions in
a structured format.\n
A question can have up to five selectable answers.\n
You must specify the correct number as its number index.
The first selection is 0, second is 1, and so on.\n
TIP: The correct answer must be in a random selection.\n
TIP: Each selection should be no longer than two
sentences.\n
TIP: You must not include 'None of the Above' as a
selection.\n
The language must be in {language}.\n
[INPUT_DATA]:\n
{input_data}
'''
prompt = PromptTemplate.from_template(template)
```

prompt

3. 구현 원리 설명 (프롬프트)

- prompt을 통해, LLM에게 특정 과제를 해 달라고 지시
- LLM의 장점: 말을 빠릿빠릿하게 잘 듣는다.
- LLM의 단점: **말로 설명 안 해주면, 아무 것도 하지 않는다.** 정확한 프롬프트를 작성하는 것이 중요.
- num_questions, language, input_data는 변수: 나중에 입력받음

```
template = '''
You are a study assistant which must generate unique
{num_questions} multiple-choice questions for students.\n
By using [INPUT_DATA], make multiple choice questions in
a structured format.\n
A question can have up to five selectable answers.\n
You must specify the correct number as its number index.
The first selection is 0, second is 1, and so on.\n
TIP: The correct answer must be in a random selection.\n
TIP: Each selection should be no longer than two
sentences.\n
TIP: You must not include 'None of the Above' as a
selection.\n
The language must be in {language}.\n
[INPUT_DATA]:\n
{input_data}
...
prompt = PromptTemplate.from_template(template)
```

- input_data를 사용해서 객관식 문제를 만들어줘.
- 문제는 총 num_questions개 만들어줘.
- 문제는 language(한국어/영어?)로 출제해줘.

3. 구현 원리 설명 (출력값)

- LLM은 답변을 output_schema의 구조화된 형태로 반환
- 웹사이트 제작 시 문제 데이터를 용이하게 사용하려면, 구조화된 형태로 반환하는 것이 중요
- 각 문제는 질문(question), 선지(selection), 정답(correct), 해설(explanation)으로 구성된 SingleQuestion 객체로 생성됨
- 모의고사는 SingleQuestion 객체의 list인 Questionmaker 객체로 생성됨

```
class SingleQuestion(BaseModel):
    question: str = Field(description="Multiple choice questions")
    selection: List[str] = Field(description="List of selectable answers for each question")
    correct: str = Field(description="Correct answer for question")
    explanation: str = Field(description="Explanation on why the answer is correct for each question")

class Questionmaker(BaseModel):
    questions: List[SingleQuestion] = Field(
        description="List of {num_questions} questions")
```

```
[SingleQuestion(question='Neuron이 구성되어 있는 세포 중 soma의 주요 역할은 무엇인가요?', selection=['단백질 합성', '에너지 생성', '정보 전달', '세포 호흡'], correct='0', explanation='Soma는 neuron의 활동에 사용되는 단백질을 합성하는 역할을 합니다.'),
 SingleQuestion(question='Neuron의 axon이 다른 neuron으로 정보를 전달하는 기능을 하는 부분은 어디인가요?', selection=['Soma', 'Dendrite', 'Axon Terminal', 'Synapse'], correct='2', explanation='Neuron의 axon이 다른 neuron으로 정보를 전달하는 기능을 하는 부분은 Axon Terminal입니다.'),
 SingleQuestion(question='Neuron의 axon에서 soma 방향으로의 물질 이동을 담당하는 transport 과정은 무엇인가요?', selection=['Anterograde Transport', 'Retrograde Transport', 'Synaptic Transmission', 'Neurotransmitter Storage'], correct='1', explanation='Neuron의 axon에서 soma 방향으로의 물질 이동을 담당하는 transport 과정은 Retrograde Transport입니다.')] ]
```


3. 구현 원리 설명 (언어모델)

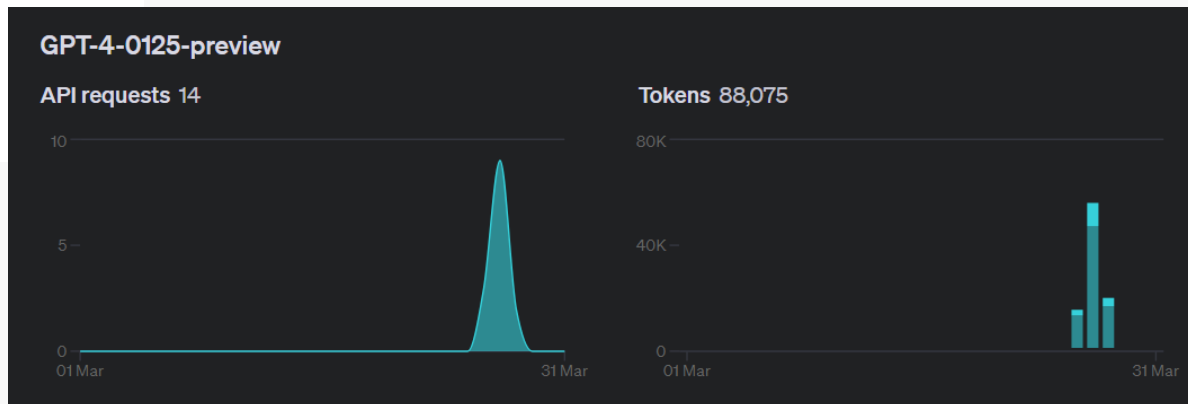
- 11m으론 OpenAI의 GPT-4를 사용
- 공짜가 아님... 처리하는 token 1개당 일정 요금을 받음.
- 앞서 본 <동백꽃> 모의고사를 1번 생성할 때마다 약 \$0.125를 지불해야 함.
- Prompt와 Schema를 영어로 작성한 이유: 한국어보다 영어가 차지하는 토큰 수가 적어서 저렴

```
model = ChatOpenAI  
(model="gpt-4-0125-preview",  
temperature=0.2,  
max_tokens=None)
```

We are excited to announce GPT-4 has [a new pricing model](#), in which we have reduced the price of the prompt tokens.

For our models with **128k** context lengths (e.g. `gpt-4-1106-preview` and `gpt-4-1106-vision-preview`), the price is:

- \$10.00 / 1 million prompt tokens (or \$0.01 / 1K prompt tokens)
- \$30.00 / 1 million sampled tokens (or \$0.03 / 1K sampled tokens)



3. 구현 원리 설명 (체인)

- `created_structured_output_runnable`는 `chain`을 반환
- `chain.invoke(): num_questions, language, input_data`를 `dictionary`로 입력받은 뒤, LLM에 명령을 수행해 응답을 받는 과정을 수행
- 위와 같이 일련의 과정을 수행하기 때문에, 이름이 `chain`인 겁니다 (사슬...)

```
def generate_questionnaire(input_text, language,
                           num_questions):
    response = chain.invoke({"input_data": input_text,
                             "language": language, "num_questions": num_questions})
    return response.questions

with open("neuro.txt", encoding='utf-8') as f:
    sample = f.read()

if __name__ == "__main__":
    pprint(generate_questionnaire(sample, "Korean", 3))
```



```
[SingleQuestion(question='Neuron이 구성되어 있는 세포 중 soma의 주요 역할은 무엇인가요?', selection=['단백질 합성', '에너지 생성', '정보 전달', '세포 보호'], correct='0', explanation='Soma는 neuron의 활동에 사용되는 단백질을 합성하는 역할을 합니다.'),
 SingleQuestion(question='Neuron의 axon이 다른 neuron으로 정보를 전달하는 기능을 하는 부분은 어디인가요?', selection=['Soma', 'Dendrite', 'Axon Terminal', 'Synapse'], correct='2', explanation='Neuron의 axon이 다른 neuron으로 정보를 전달하는 기능을 하는 부분은 Axon Terminal입니다.'),
 SingleQuestion(question='Neuron의 axon에서 soma 방향으로의 물질 이동을 담당하는 transport 과정은 무엇인가요?', selection=['Anterograde Transport', 'Retrograde Transport', 'Synaptic Transmission', 'Neurotransmitter Storage'], correct='1', explanation='Neuron의 axon에서 soma 방향으로의 물질 이동을 담당하는 transport 과정은 Retrograde Transport입니다.')]
]
```

sample 글을 바탕으로 문제 3개를
한국어로 출제해줘!

3. 구현 원리 설명 (Streamlit)

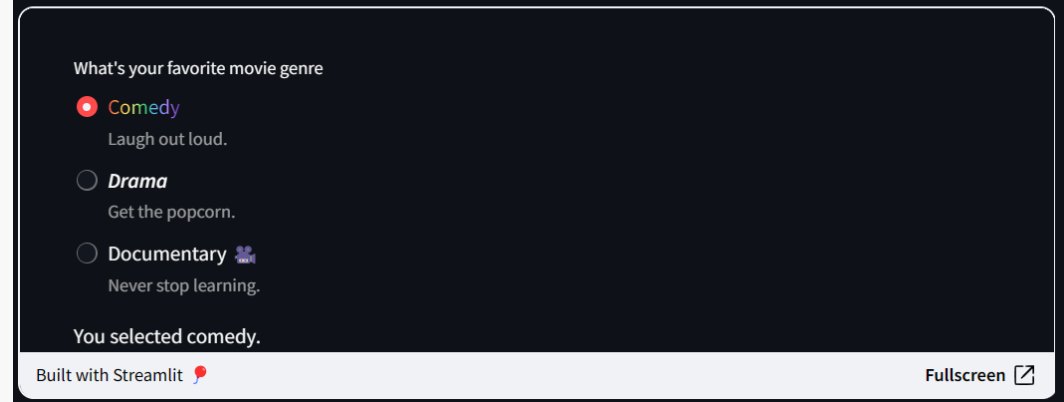
- Streamlit: 간단한 웹사이트를 구현할 수 있는 Python 패키지
- 다른 웹프로그래밍 언어(HTML, CSS, Javascript) 없이도 간단한 웹사이트를 만들 수 있음
- 사용자의 입력을 실시간으로 처리해 프로그램에 활용 가능 (예: radio buttons)



```
import streamlit as st

genre = st.radio(
    "What's your favorite movie genre",
    [":rainbow[Comedy]", "***Drama***", "Documentary :movie_camera:"],
    captions = ["Laugh out loud.", "Get the popcorn.", "Never stop learning."])

if genre == ':rainbow[Comedy]':
    st.write('You selected comedy.')
else:
    st.write("You didn't select comedy.")
```

A screenshot of a Streamlit web application. It displays a radio button interface for selecting a movie genre. The options are "Comedy" (highlighted in rainbow), "Drama" (highlighted in dark red), and "Documentary" (highlighted in purple). Below each option is a corresponding caption: "Laugh out loud.", "Get the popcorn.", and "Never stop learning.". A message at the bottom states "You selected comedy.". The footer includes "Built with Streamlit" and a "Fullscreen" button.

3. 구현 원리 설명 (Streamlit)

- 1단계: num_questions, language, input_data를 웹사이트를 통해 입력받은 뒤 chain으로 전달해 각 문제, 선지, 정답, 해설이 담긴 Questionmaker 객체를 반환받음.
- 2단계: 각 문제를 for문을 이용해 웹사이트에 표시. Radio button을 통해 답안을 입력받음.
- 3단계: 사용자의 답안과, 실제 문제의 답안을 for문을 이용해 비교하여 채점 및 해설 표시.

모의고사 자동제작 서비스

내용 입력

87
점순이가 겁을 잔뜩 집어먹고 꽃밀을 살금살금 기어서 산아래로 내려간 다음 나는 바위를 끼고 엉금엉금 기어서 산위로 치빼지 않을 수 없었다.

언어 선택 (영어가 더 정확합니다.)

☒ 한국어
☐ English

생성할 문제 수 설정 (인공지능의 한계로 더 적은 수의 문제가 생성될 수 있음)

1 20

모의고사 생성하기

점순이 주인공에게 준 것은 무엇인가요?

☐ 사과
☒ 감자
☐ 빵
☐ 초콜릿
☐ 과자

주인공이 점순에게 감자를 돌려준 방식은?

☐ 손으로 직접 건네주었다
☐ 바구니에 담아서 돌려주었다
☒ 어깨 너머로 밀어 버렸다
☐ 편지와 함께 보냈다
☐ 땅에 묻어서 돌려주었다

점순이 주인공에게 무엇을 물었을 때, 주인공이 무턱대고 대답한 말은?

☐ 알겠어
☐ 그래
☐ 모르겠어
☒ 안 할게
☐ 다시는 그러지 않을게

오답입니다! 점순이 '그럼 너 이담부턴 안 그럴 테냐?'라고 물었을 때, 주인공은 '그래!'라고 무턱대고 대답했습니다.

주인공이 점순네 수탉을 때려 었은 이유는 무엇인가요?

☐ 점순네 수탉이 주인공을 공격했기 때문에
☐ 점순네 수탉이 주인공의 수탉을 공격해서
☒ 주인공이 화가 나서
☐ 점순네 수탉이 도망쳤기 때문에
☐ 점순네 수탉이 소리를 너무 크게 내서

정답입니다! 주인공은 화가 나서 나도 모르는 사이에 큰 수탉을 단매로 때려 었었습니다.

4. 앞으로 할 일 정리

- .docx 및 .pdf 파일 첨부 기능 추가
 - python-docx, pypdf 등 패키지로 첨부파일의 내용을 읽어 구현 가능
- Huggingface Spaces를 이용해 웹서비스를 링크로 접속할 수 있게 설정
 - 다만 OpenAI 토큰의 요금 문제로, 완전히 공개하진 않을 예정
- 사용성 평가 진행
 - 5명 정도 참여자를 모아, 사용성 평가를 진행 예정
 - 실험이 아니라 사용성 평가인 만큼, 자신이 원하는 지문을 이용해 사용할 기회를 줄 것임
 - 평가 역시 점수 평가보단, 개선점이나 교육에 미칠 영향 등을 주관식으로 물어볼 예정
- 위 내용은 제 개인일정에 따라 변동될 수 있음.

질문과 답변