

2025 금융 AI Challenge

: 금융 AI 모델 경쟁



MinuSpurs

오민우 (gapbu234@gmail.com)

김성연 (gimseongyeon545@gmail.com)

Contents



1. Introduction

2. Model

3. RAG

4. Prompt

5. Result



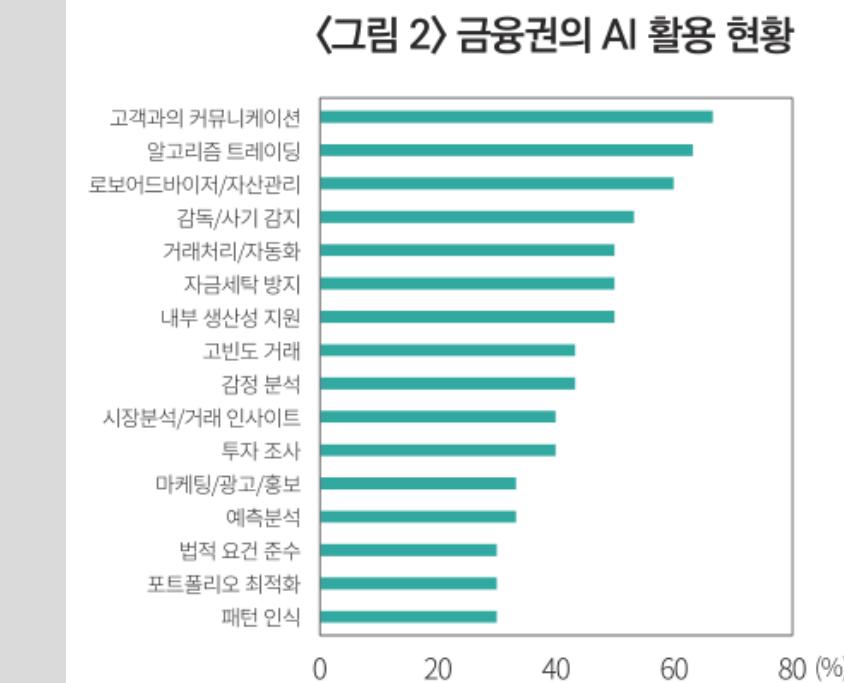
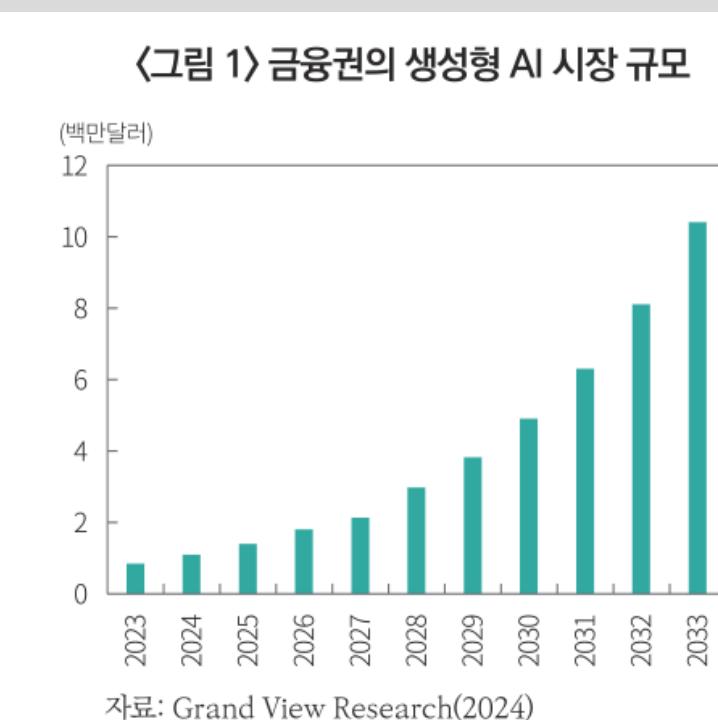
1. Introduction

문제 정의



• 대회 목표

- 금융 보안 실무 관련 객관식 및 주관식 문제에 대한 정확한 응답을 생성하는 AI 모델 개발
- 금융 산업의 AI 활용 활성화 지원



문제 정의



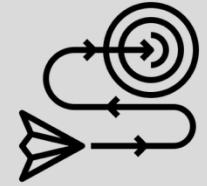
- 데이터셋: **FSKU (Financial Security Knowledge Understanding)**
 - 금융 보안 관련 업무에 활용할 **AI** 모델의 성능을 비교하여 최적의 모델을 선정하기 위해 개발된 질문-답변 데이터셋
 - 주관식 + 객관식
- 평가 산식: 최종 **score** = 객관식 * 0.5 + 주관식 * 0.5
 - 객관식: **Accuracy**
 - 주관식: 0.6 * **Cosine Similarity** + 0.4 * **Keyword Recall**

$$\text{Accuracy} = \frac{\text{정답과 일치한 객관식 문항 수}}{\text{전체 객관식 문항 수}}$$

$$\text{Keyword Recall} = \frac{|\text{예측 답변에 포함된 정답 키워드}|}{|\text{전체 정답 키워드}|}$$

$$\text{CosineSimilarity} = \cos(\theta) = \frac{(\vec{E}_{gt} \cdot \vec{E}_{pred})}{|\vec{E}_{gt}| \cdot |\vec{E}_{pred}|}$$

주요 전략



- **Model**

- 한국어 기반 모델
- **RAG**에 적합한 모델
- 주관식 답변의 표현력 및 문장 완성력을 위한 모델

- **RAG**

- 법령-시행령-시행규칙 / 감독규정 / 운영기준 pdf를 조문 단위 **Chunking**
- **Indexing / Reranking**

- **Prompt**

- [참고할 문서]로 **RAG**를 통해 얻은 정보 추가
- 미응답 방지

개발 환경



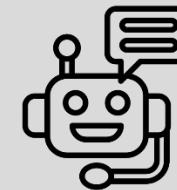
- Ubuntu 22.04
- Language: Python 3.10
- GPU: RTX 4080, NVIDIA CUDA 12.x
- Library

용도	라이브러리
Runtime	Python
ML compute	PyTorch
Model & Embedding	Transformers, Sentence-Transformers, BAAI/bge-m3
Orchestration	LangChain
Vector/Index	FAISS (faiss-cpu)
Data Processing	Pandas, Scikit-learn
Document Parsing	PyPDF, Lxml
Utils	Tqdm





2. Model



- K-intelligence/Midm-2.0-Base-Instruct (2025.07.04 출시 😊)  Mi:dm

- Transformer 구조를 활용한 Autoregressive Language Model

- 고려 사항

- 한국어 특화 모델 & 문장 완성도 (표현력)

- AI-HUB & 국립국어원 말뭉치 데이터를 pre-training → 한국어 및 영어의 이해와 생성에 능숙
 - 한국어 특화 Tokenizer 를 통해 한국어 문장의 압출력 ↑

- 문장 이해도 (RAG) & 환각(Hallucination) ↓

- 검색된 근거를 프롬프트에 넣어주고, 근거를 따르도록 학습됨 → RAG 친화적

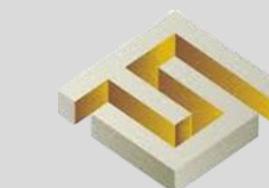
별도의 학습 X
Prompt 에 RAG 컨텍스트를 넣는 전략

- 속도 (용량)

- Capacity: 11.5B (115억 파라미터)
 - Max Context-Length: 32K

- 금융 도메인

- RAG 학습용 문서로 금융 문서가 사용됨



Optimization & Hyperparameter



- Optimization

- 양자화(4bit) → GPU 메모리 효율성
- **temperature = 0, top_p = 1.0** 설정
 - 확률적 다양성 대신 정확도 중시 → 객관식 accuracy ↑

- Hyperparameter

- **max_new_tokens: 512, 768, 1024 → 1024**
- **temperature: 0.0~0.7 → 0.0**
- **top_p: 0.8, 0.9, 1.0 → 1.0**

```
● ● ●  
# 생성  
out = _PIPE(  
    final_prompt,  
    max_new_tokens=1024,  
    temperature=0.0, # 결정론적 추론  
    top_p=1.0,  
    do_sample=False,  
)
```



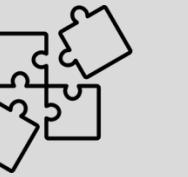
3. RAG

Documents



- 법령-시행령-시행규칙 3단 비교
 - 전자서명법/자본시장보호법/신용정보법/정보통신망법/개인정보보호법/전자금융거래법
 - 3단 비교: 법령을 기반으로 위임·인용한 시행령·시행규칙을 대응한 근거 사슬을 보여주는 표
- 감독규정
 - 전자금융감독규정
 - 신용정보업감독규정
- 시행령
 - 전자서명법/자본시장보호법/신용정보법/정보통신망법/개인정보보호법/전자금융거래법
- 시행규칙
 - 전자서명법/자본시장보호법/신용정보법/정보통신망법

STEP 1. Chunking



STEP 1
Chunking

STEP2

STEP3

STEP4

- 3단 비교 문서
 - 인용/위임 **HTML/XLS → CSV**
저장
 - 각 법령 조문별 **family csv** 생성
(중복 제거)
- 시행령/시행규칙/감독규정/운영
기준 **pdf**
 - 장/조문/부칙 **chunking**
 - 정규식 이용

STEP 2. Indexing



STEP 1
Chunking

STEP2
Indexing

STEP3

STEP4

- **Embedding**
 - 임베딩 모델: **BAAI/bge-m3**
 - chunking 한 내용을 벡터화
 - 벡터화 한 내용과
메타데이터를 저장
- **FAISS**
 - 질문과 유사도 검색이 가능

```
● ● ●  
embed_model: str = "BAAI/bge-m3"  
  
emb = HuggingFaceBgeEmbeddings(  
    model_name=embed_model,  
    model_kwarg={"device": device_for_embed},  
    encode_kwarg={"normalize_embeddings": True}  
)
```

STEP 3. Reranking



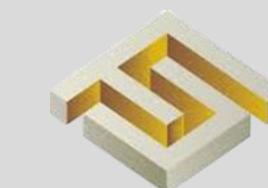
STEP 1
Chunking

STEP2
Indexing

STEP3
Reranking

STEP4

- 객/주관식 판별
- Case 1) 명시 법령 조문 0
 - “전자금융거래법 제1조에 따르면~“
 - FAISS 대신 바로 family csv에서 사용
 - accuracy ↑
- Case 2) Other (조문 미명시 or 법령 외 언급)
 - 질문을 bge-m3로 임베딩하여 FAISS에서 유사도 검색 후 문서 추출
 - Reranking: 힌트 가중치, 정규화, 중복제거 → 노이즈 감소
 - recall + cosine similarity



금융보안원
FINANCIAL SECURITY INSTITUTE



4. Prompt

STEP 4. Prompt Engineering



STEP 1
Chunking

STEP2
Indexing

STEP3
Reranking

STEP4
Prompt Engineering

4

STEP 1



Documents



Chunking



Indexing
(BAAI/bge-m3, FAISS)

STEP2



Question

개인정보보호법 제64조의2에 따라 과징금을 부과받은 금융기관이 과징금을 납부한 후, 보호위원회에 통보해야 하는 시점은 언제인가?
1 과징금을 납부한 다음 달 말일까지
2 과징금을 납부한 다음 분기 말일까지
3 과징금을 납부한 다음 해 말일까지
4 과징금을 납부한 즉시
5 과징금 부과 사실을 분기별로 일괄 통보할 때



객/주관식 및 Case 1,2 판별
→ 객관식 / Case 1



Question Embedding

STEP3



Reranking + 중복 제거
→ [참고할 문서] 생성

[참고할 문서]

[1] [개인정보보호법]
제64조의2
제64조의2(과징금의 부과) ①
보호위원회는 다음 각 호의 어느 하나에 해당하는 경우에는~
...

[5] [개인정보보호법시행령]
제60조의3

제60조의3(과징금의 부과 및 납부) ~ ④ 금융기관이 제2항에 따라 과징금을 수납한 때에는 지체 없이 그 사실을 보호위원회에 통보해야 한다.

STEP4



Prompt ([참고할 문서] + 미응답 방지)

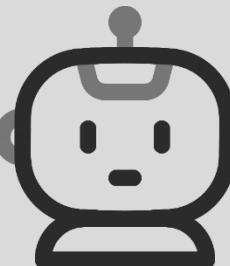
[참고할 문서]

당신은 금융보안 전문가입니다.
아래 [실제 질문]에 대해 가장 적절한 **정답 선택지 번호 한 개만 출력**하세요.
절대 미응답 하지 마세요.
[실제 질문]: 개인정보보호법 제64조의2에 따라 과징금을 부과받은 금융기관이 과징금을 납부한 후, 보호위원회에 통보해야 하는 시점은 언제인가?

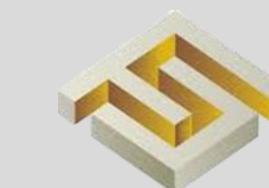
선택지:

- 1 과징금을 납부한 다음 달 말일까지
- 2 과징금을 납부한 다음 분기 말일까지
- 3 과징금을 납부한 다음 해 말일까지
- 4 과징금을 납부한 즉시
- 5 과징금 부과 사실을 분기별로 일괄 통보할 때

답변:



LLM
(KT Mi:dm)



금융보안원
FINANCIAL SECURITY INSTITUTE



5. Result

Result





Thank You