

경제금융기업 AI 활용, 취업 지원 플랫폼

신입 구직자 중심 기업분석 및 챗봇 서비스 제공

TEAM 영연윤영용 강영혜, 승연수, 박지윤, 박찬영, 변성용

CONTENTS

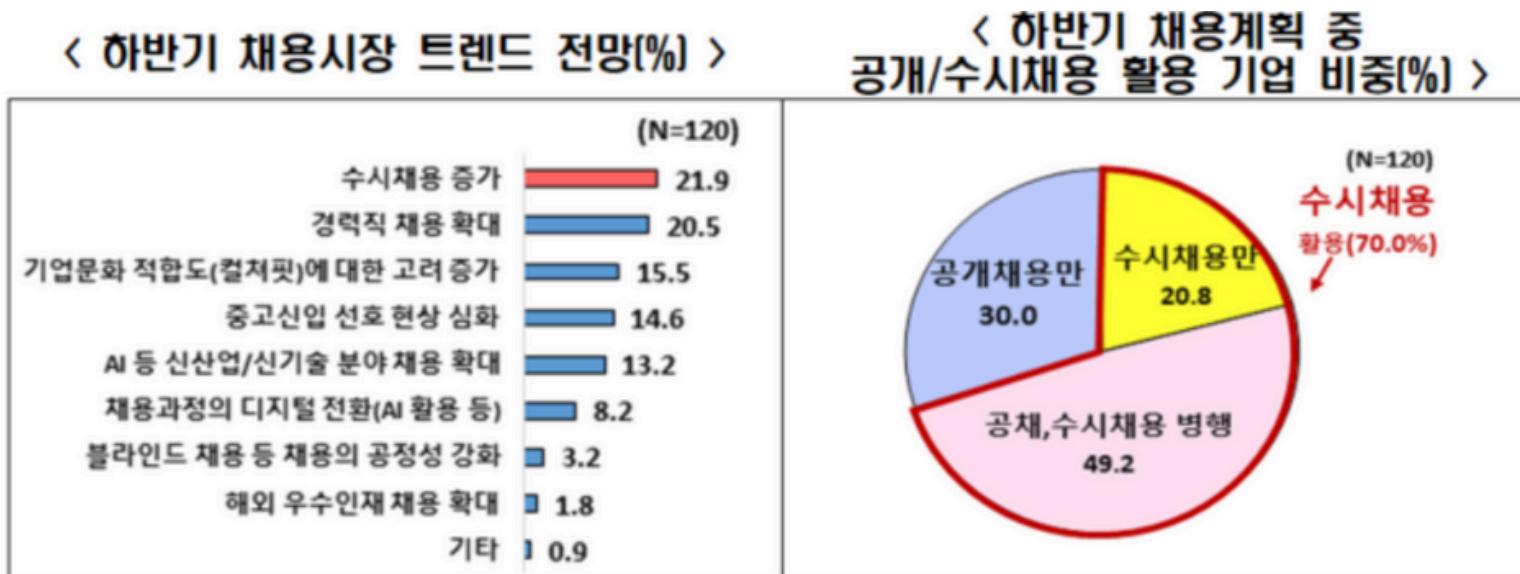
- 1 Introduction**
- 2 Outline**
- 3 Method**
- 4 Result**

- 5 Expected effects**
- 6 Future work**
- 7 Project Insight**
- 8 Q&A**

01 Introduction

최근 채용시장 트렌드

대기업 10곳 중 7곳(70.0%), 수시채용 활용
작년(55.9%)보다 **14.1%p** 증가



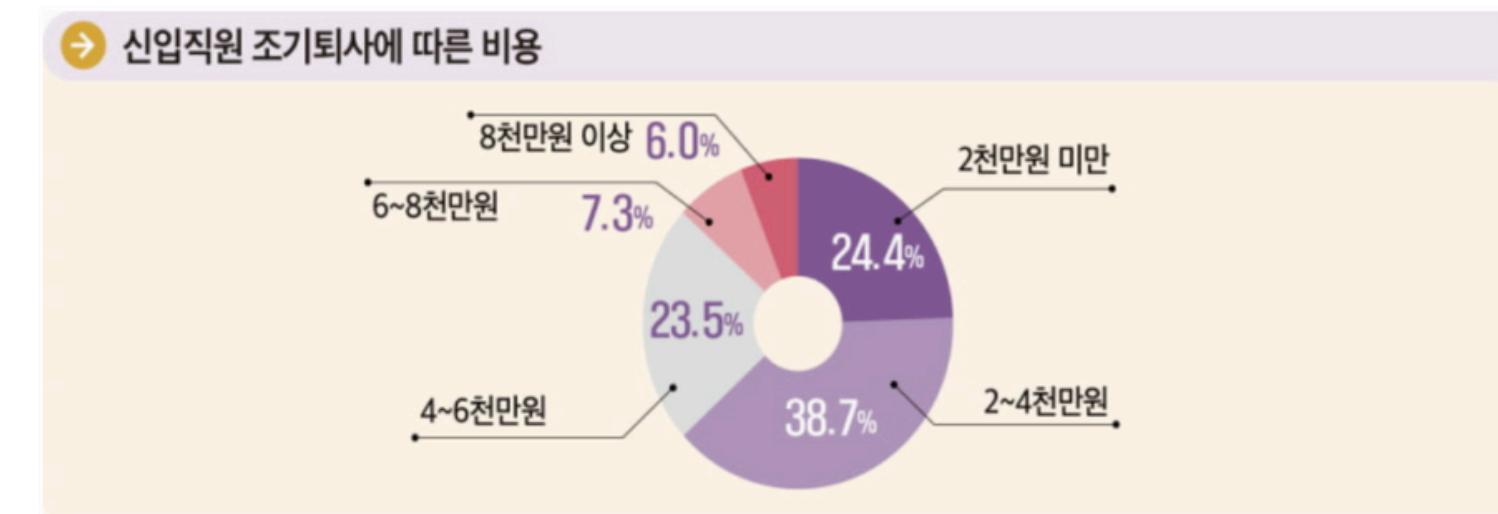
[출처] FKI 한국경제인협회 | 2024.08.29

수시 채용 증가 이유

“신입사원 조기퇴사율 81%”

[출처] 한국경영자총협회 | 2023.03.20

“조기 퇴사에 따른 비용 증가”



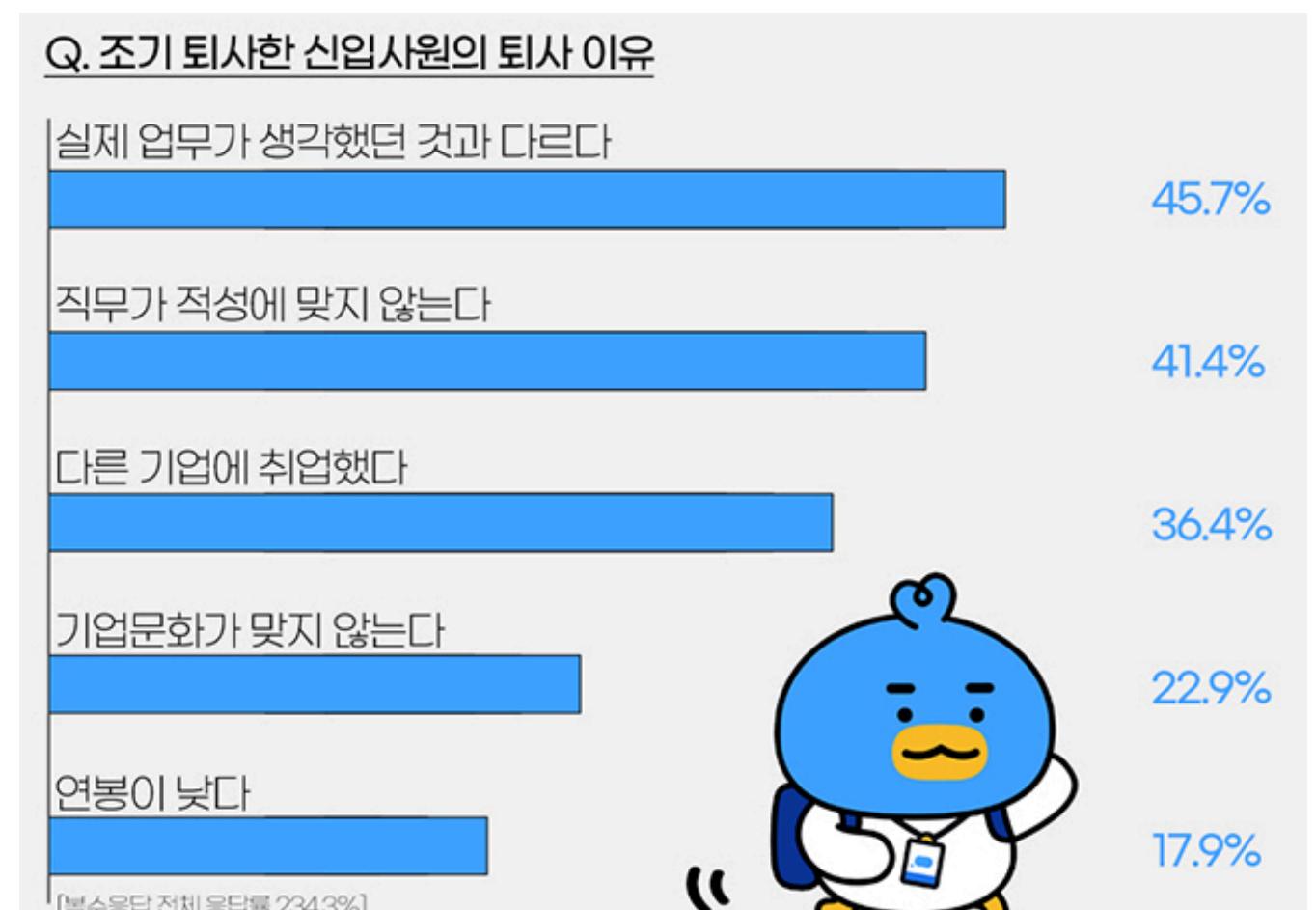
[출처] 고용노동부 | 2024.03.24

기업의 결론:
수시 채용으로 준비된 인재를 채용해
교육 비용을 줄이자

01 Introduction

Why 퇴사?

직무 이해도 부족, 기업 분석의 실패



Why. 왜 직무에 관한 이해도가 부족할까?
입사자와 기업간 직무 이해도에 관한 간극이 있다.

Why. 왜 직무 이해도에 관한 간극이 생겼을까?
구직 과정에서 기업에 관한 정보를 취득하기 어렵다.

Why. 왜 기업에 관한 정보 획득이 어려울까?
정보가 너무 많고, 시대의 흐름에 따른 이슈 파악이 어렵다.

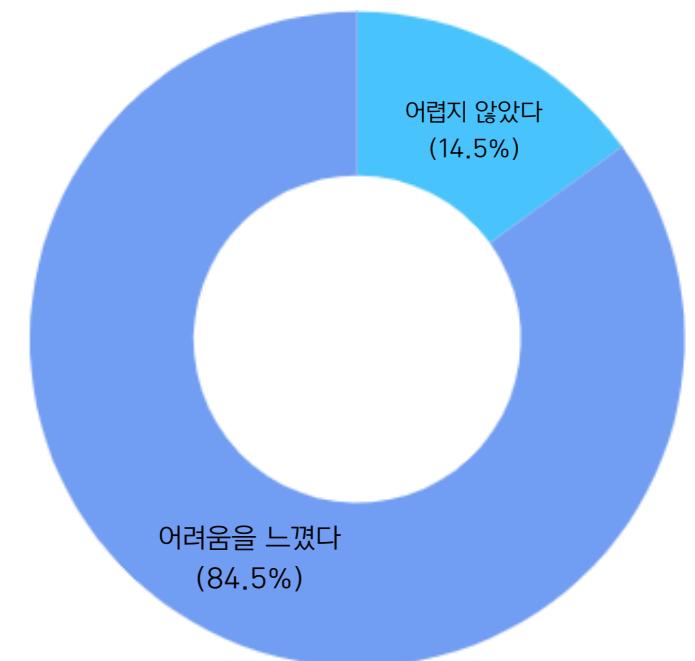
Why. 왜 파악이 어려울까?
산업이 빠르게 변화하고, 그에 따른 정보를 한번에 제공하는 서비스가 없다.

01 Introduction

정보 접근의 어려움

01

84.5%가 취업 정보 수집에
어려움을 느꼈다



02

구직자 3명 중 1명은
정보 부족으로 인해
막막함을 느낀 적이 있다



03

정보 수집에
어려움을 느끼는 이유

- 어떤 정보를 준비해야 할지 몰라서, 38.4%
- 정보를 얻을 만한 선배가 없어서, 36.8%
- 취업 스터디를 하지 않아서, 36.1%
- 지역별 정보 편차 때문에, 25.7%
- 취업 사교육을 하지 않아서, 16.6%

[출처] 잡코리아

AI 활용 취업 지원 플랫폼

(Proto: KDB산업은행)

02 Outline

경제금융기업 구직자를 위한 취업지원

경제금융기업 취업을 준비하는 구직자들에게 체계적이고 효율적인 지원을 제공하는 플랫폼입니다. 이 플랫폼은 경제 시장 분석과 기업 분석에 중점을 두어, 사용자가 보다 적은 시간과 노력으로 경제 및 기업 동향을 파악할 수 있도록 돕습니다. 이를 통해 자기소개서 작성과 면접 준비를 효과적으로 지원하며, 입사 후 업무에 대한 혼란을 최소화합니다. 또한, 사용자 맞춤형 정보를 제공함으로써 개인화된 취업 전략 수립이 가능하도록 설계되었습니다.



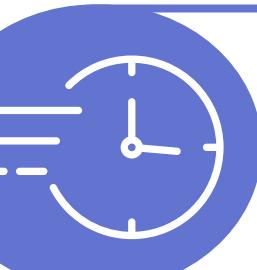
02 Outline



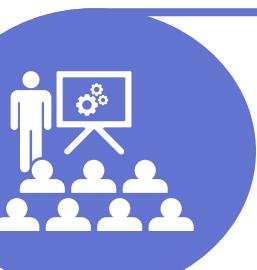
월별 경제·금융 정보 통합 제공



구직자 중심의 데이터 분석 및 활용



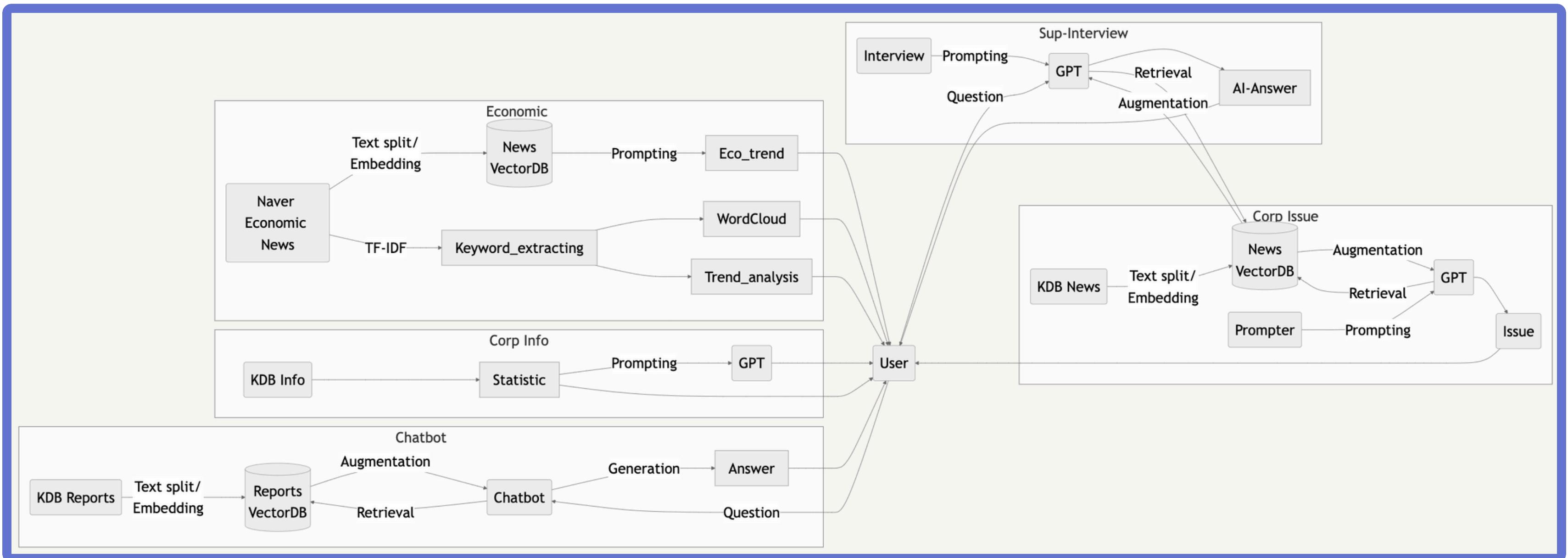
챗봇을 이용한 구직 시간 단축



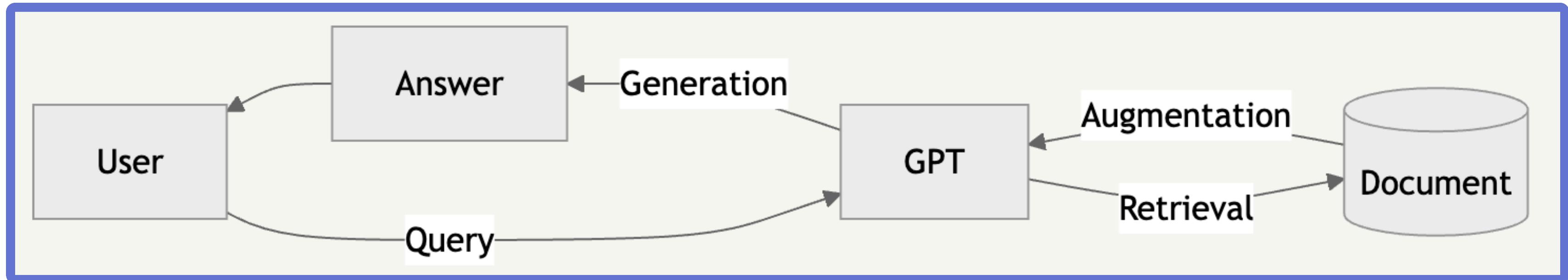
확장 가능한 플랫폼 설계 및 적용

03 Method

Framework



03 Method



Why RAG?

문맥 확장

외부 문서를 활용해 사용자의 질문에 대한 깊이 있는 정보 제공

데이터 기반 응답

실시간 데이터 검색을 통해 신뢰성 높은 답변 생성

학습 비용 절감

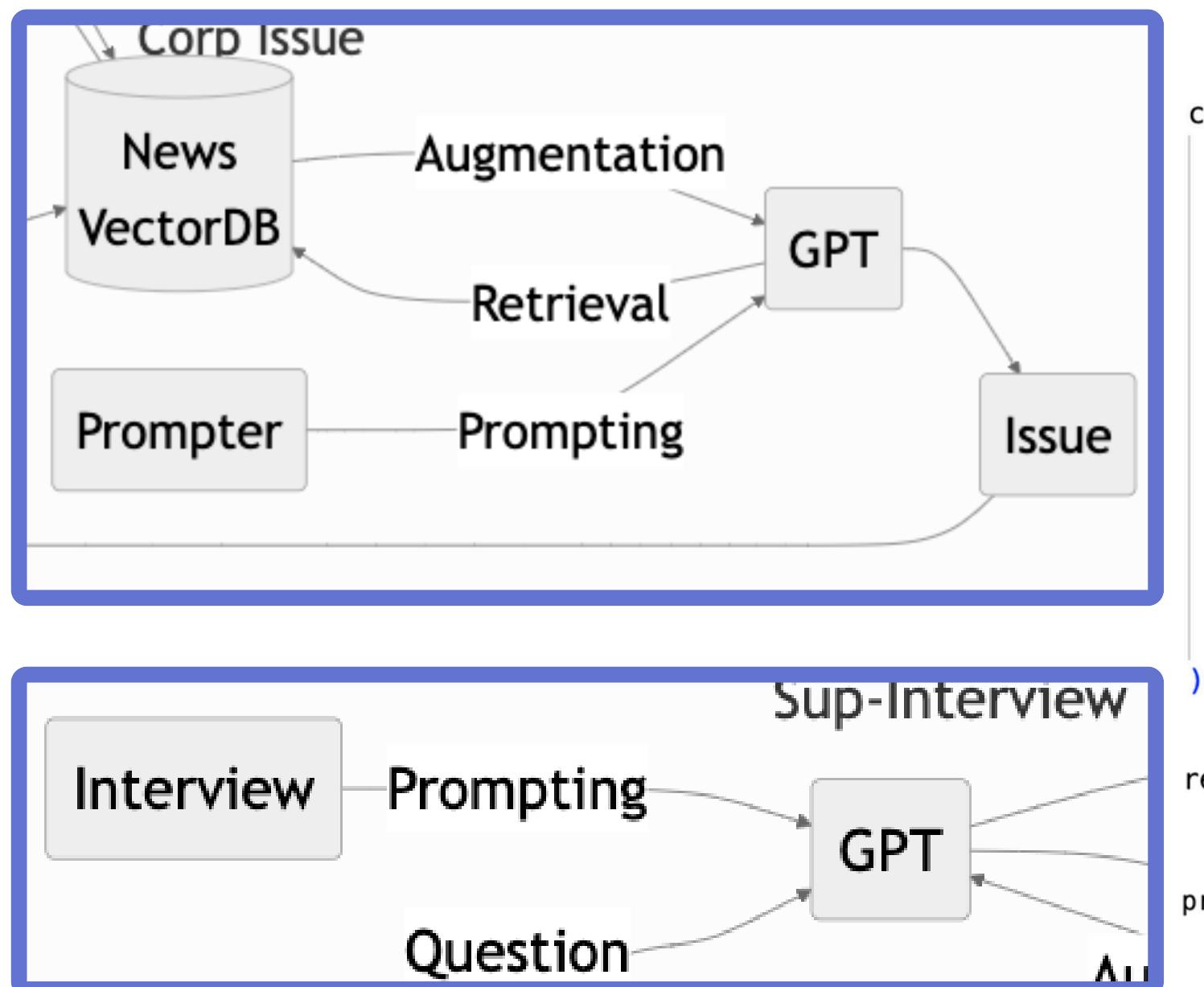
특정 데이터를 모델에 직접 학습시키지 않고 실시간 검색으로 해결

응답 품질 향상

GPT 모델의 기본 능력에 외부 데이터를 추가해 응답의 정확성 및 전문성 향상

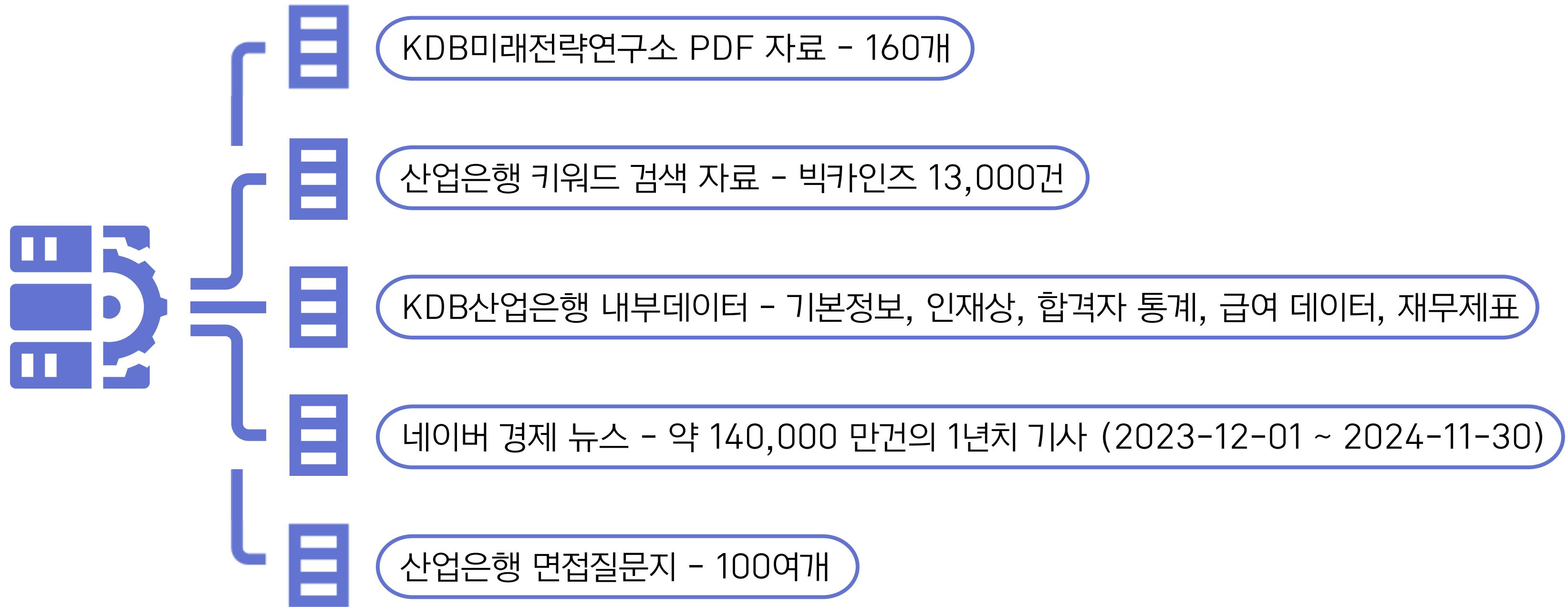
03 Method

Prompt Engineering



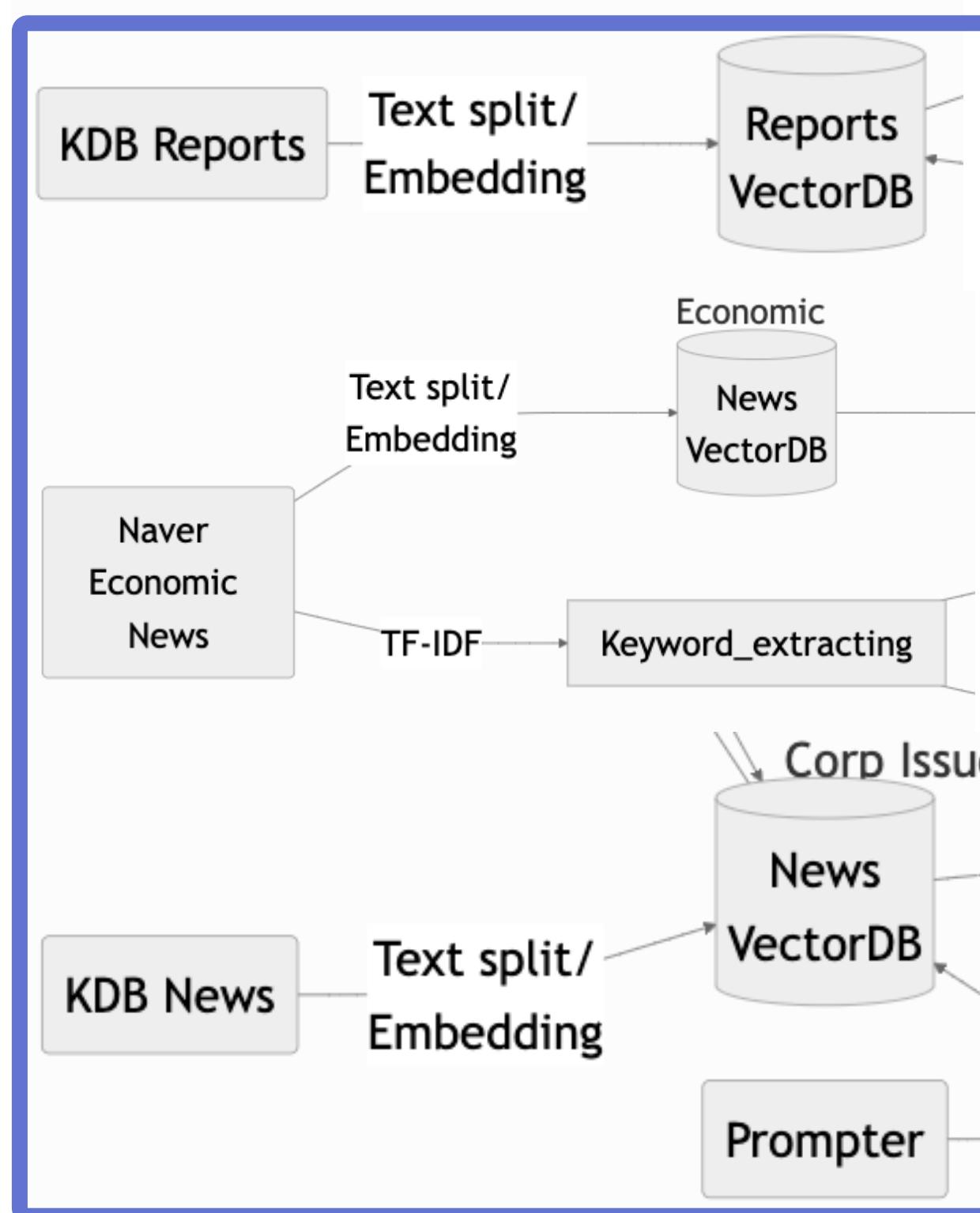
03 Method

Data Set



03 Method

Data Preprocessing



- **Text Split** : RecursiveCharacterTextSplitter
 - chunk_size =1000
 - chunk_overlap = 100
- **Embedding Method** : OpenAIEmbedding
 - Vector_size : 1536
- **TF-IDF**
 - top 20 keywords daily -> top 20 keywords monthly
 - top 50 keyword of a year – wordcloud
- **VectorDB** – FAISS
 - Similarity index – Cosine similarity

04 Result

<http://192.168.0.27:8501>

04 Result 경제 이슈 분석

```
@st.cache_data
def get_news_content(data, date):
    """선택한 날짜에 해당하는 뉴스 기사 내용 가져오기"""
    news_content = data[data['date'] == date]
    news_content_str = ' '.join(news_content['information'].dropna())
    if not news_content_str:
        st.error("해당 날짜의 뉴스 기사 내용이 없습니다.")
    return news_content_str

def get_headline(news_content):
    """LangChain을 통해 헤드라인 생성하기 (캐시하지 않음)"""
    llm = ChatGoogleGenerativeAI(model="gemini-1.5-flash", temperature=0.5)
```

```
@st.cache_data
def get_answer(news_content):
    """LangChain을 통해 헤드라인 생성"""
    prompt = PromptTemplate(
        input_variables=["news_content"],
        template="""다음 뉴스 기사들의 핵심 내용을 바탕으로 간결하고 매력적인 헤드라인 4개를 생성해주세요.
각 헤드라인은 고유하고 서로 다른 주요 포인트를 강조해야 합니다. 헤드라인만 표시합니다.
글머리 기호를 사용합니다.

뉴스 기사 내용:
{news_content}
지침:
- 간결하고 명확한 헤드라인 작성
- 중복 없이 고유한 내용 강조
- 한국어로 작성
- 각 헤드라인은 다른 관점이나 측면을 다뤄야 함
"""

    )
    # LLMChain 설정
    chain = prompt | llm

    # LangChain을 통해 헤드라인 추출
    headlines = chain.invoke({'news_content': news_content})

    return headlines

return get_answer(news_content)
```



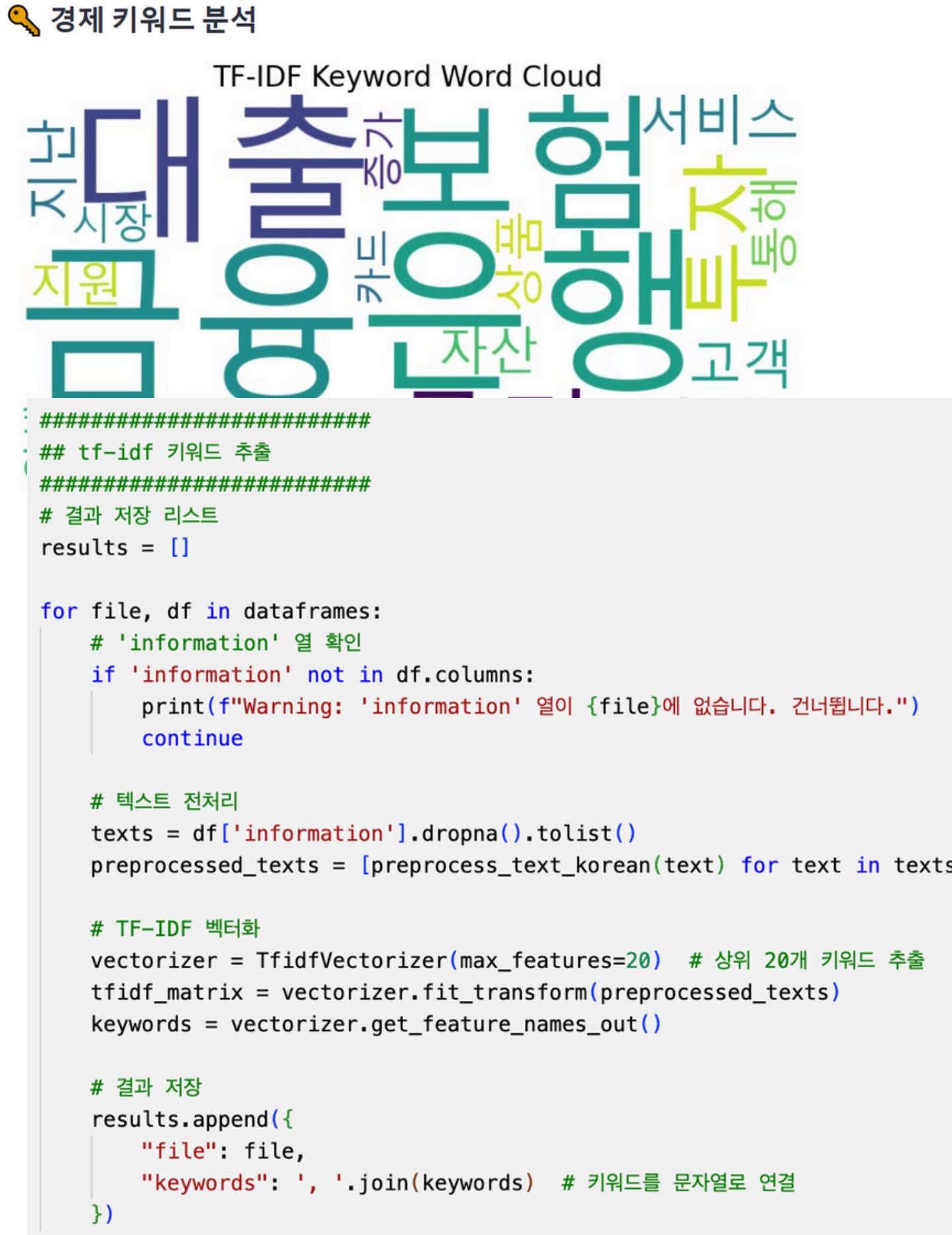
2024년 11월 30일 헤드라인

- 반도체 위기, 트럼프 리스크와 겨울나기 전략
- 엔화 예금 감소, 엔테크 열기 식나?
- 연말정산 절세 꿀팁: 연금계좌 활용으로 최대 148만원 환급
- 비트코인 10만 달러 돌파 임박, 가상자산 시장 활황 지속될까?

- 선택된 날짜의 기사 데이터 가져오기
- Langchain을 통해 기사 데이터를 LLM 이용해서 헤드라인 작성
- 캐시 기능을 통해 동일 기능 반복 호출 방지



04 Result 경제 키워드 분석 (워드 클라우드)



04 Result 기업 정보 및 채용 통계

금융 공기업 정보



KDB 산업은행

기업 정보

인재상

산업: 산업은행
주소: 서울특별시 영등포구 여의도
설립일: 1954.1.1
주력 사업: 기업 금융 지원 및 구조 조정
자본금: 70조 원
매출액: 15조 원
대표자: 강석훈

고객 중심: 고객의 니즈를 최우선으로 생각하며, 항상 고객의 요구에 부응합니다.
도전과 변화: 익숙함에 머무르지 않고 계산된 도전을 통해 새로운 변화를 수용합니다.
소통과 협력: 주도적으로 소통하여 협력을 통해 더 나은 길을 찾습니다.
현장 중심: 문제를 현장에서 해결하며, 실행력을 갖춘 인재를 선호합니다.
책임감과 신뢰: 책임을 완수하여 사회적 신뢰를 얻고자 합니다.
미래 지향: 미래를 생각하며 지속 가능한 성장을 추구합니다.
전문성: 각 분야에서 전문성을 갖추어 대안을 제시합니다.
디지털 마인드: 디지털 역량과 혁신적 사고를 겸비합니다.

최근 이슈

1. "AI 코리아 펀드 워터워너로 최종 4개 운용사 선정"
2. "혁신성장펀드 4개 운용사 통과"
3. "상생금융 지원방안에 2조원 부담 예상"
4. "4대 금융지주 '역대급' 실적 경신에도 어두운 표정"

합격자 통계



RecruitInfo:

```
class RecruitInfo:  
    def __init__(self, json_file_path):  
        self.json_file_path = json_file_path  
        self.data = self._load_data()  
        self.llm = ChatGoogleGenerativeAI(model="gemini-1.5-pro")  
    def _load_data(self):  
        try:  
            with open(self.json_file_path, 'r', encoding='utf-8') as f:  
                json_data = json.load(f)  
                return json_data.get('recruit', {})  
        except Exception as e:  
            st.error(f"파일을 읽는 중 오류가 발생했습니다: {e}")  
            return {}  
    def _analysis(self, info, column, title, bar=True):  
        st.write(title)  
        df = pd.DataFrame(self.data).set_index("년도")[column]  
        if bar:  
            st.bar_chart(df)  
        else:  
            st.line_chart(df)  
    with st.popover(f"💡 {info} AI 분석"):  
        if st.button(f"AI 답변 생성", key=info):  
            with st.spinner(f"AI가 데이터를 분석 중입니다..."):  
                prompt = f"""  
                취업 준비생에게 도움이 될 수 있도록 {column} 데이터를 분석하고  
                주요 패턴과 인사이트를 두 문장으로 요약하세요:\n                """ + df.to_string(index=False)  
                st.session_state[f"{column}_analysis"] = self.llm.predict(prompt)  
            if f'{column}_analysis' in st.session_state:  
                st.write(st.session_state[f'{column}_analysis'])
```

- 기업정보 통계에 관한 tabular data를 input으로 받아 LLM으로 설명 생성

04 Result 재무제표 시각화

재무제표 시각화

시각화할 항목을 선택하세요

포괄손익계산서

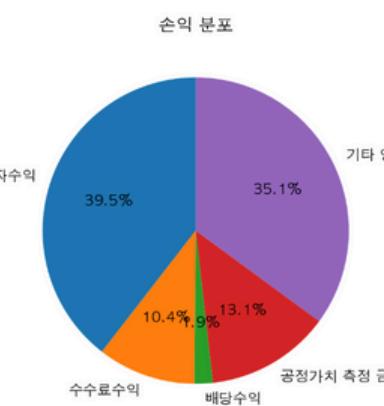
포괄손익계산서 (Statements of Comprehensive Income)	
제 09 기 2020년 1월 01일부터 2020년 12월 31일까지	
운영계정 (Operating Account)	
포괄손익계산서	
0. 이자수익	1,561,519
1. 수수료수익	409,836
2. 배당수익	76,370
3. 공정가치 측정 금융자산(FVTPL) 순손익	518,808
4. 기타 영업수익	138,6462
5. 비자연기재 손상 및 손익	-1,561,519
6. 세금 및 세금 부과금	-409,836
7. 당기 손익 - 공정가치 측정 유가 증권 관리 손익	518,808
8. 당기 손익 - 공정가치 측정 금융부채 관리 손익	-148,104
9. 기타 손익 - 투자증권 및 투자부채 관리 손익	-1,430
10. 외화환율 손익 및 손실	202,365
11. 기타 외화 손익 및 손실	-138,6462
12. 선불손실 충당금 및 손익	-882,644
13. 일반 관리비	623,951
14. 영업 손익 (손실)	2,946,855
15. 영업외 손	264,118
16. 재무제표 손익	562,550
17. 기타 영업 손	16,524
18. 영업 손익 세전	-3,241,273
19. 영업 손익 세후	752,746
20. 영업 손익 세후 (손실)	2,508,837
21. 기타 영업 손	2,643,342
22. 기타 영업 손	131,400
23. 주식으로 당기 손익으로 세부분의 영업 손익	388,753
24. 주식으로 당기 손익으로 세부분의 영업 손익	220,009
25. 주식으로 당기 손익으로 세부분의 영업 손익	-2,352
26. 환율변동 손익 및 손실	-1,312
27. 외화환율 손익 및 손실	-327,283
28. 기타 손익 및 손실	-173,433
29. 기타 손익 및 손실	-37,722
30. 당기 손익 - 투자증권 및 투자부채 관리 손익	-18,158
31. 당기 손익 - 투자증권 및 투자부채 관리 손익	2,640,357
32. 주당 영업 손익 (손실)	531
33. 기본 및 최우선 영업 손익 (손실)	531

포괄손익계산서 이미지

포괄손익계산서 데이터

항목	금액 (백만원)
0. 이자수익	1,561,519
1. 수수료수익	409,836
2. 배당수익	76,370
3. 공정가치 측정 금융자산(FVTPL) 순손익	518,808
4. 기타 영업수익	138,6462

포괄손익계산서 그래프



포괄손익계산서

```

class FinancialVisualization:
    def __init__(self, json_path, image_path):
        self.json_path = json_path
        self.image_path = image_path
        self.financial_data = self._load_financial_data()

    def _load_financial_data(self):
        try:
            with open(self.json_path, 'r', encoding='utf-8') as f:
                json_data = json.load(f)
                return json_data.get('financial', {})
        except Exception as e:
            st.error(f"파일을 읽는 중 오류가 발생했습니다: {e}")
            return {}

    def display_graph(self, data, financial_option):
        fig, ax = plt.subplots()
        if financial_option in ["재무상태표", "연결재무상태표"]:
            ax.bar(data["항목"], data["금액 (백만원)"], color="skyblue")
            ax.set_xlabel("항목")
            ax.set_ylabel("금액 (백만원)")
            ax.set_title("자산 분포")
            plt.xticks(rotation=45)
        else:
            ax.pie(data["금액 (백만원)"], labels=data["항목"], autopct='%1.1f%%', startangle=90)
            ax.set_title("손익 분포")
        st.pyplot(fig)

    def display_image(self, financial_option):
        img_path = os.path.join(self.image_path, f'{financial_option}.jpg')
        if os.path.exists(img_path):
            img = Image.open(img_path)
            st.image(img, caption=financial_option)

```

- 추후 API 연동을 통해 검색을 통한 기업 재무제표 시각화 템플릿화/자동화
- 재무제표 정보를 input으로 받아 LLM으로 설명 생성

04 Result 면접준비 Q&A

경제금융기업 AI 활용 취업 지원

산업은행 면접 준비 도우미

면접 질문을 선택하시면 AI가 답변을 제공해 드립니다.

금융 및 경제

ESG 금융의 활성화 방안

산업구조 고도화를 위한 투자전략

Get AI-generated answers

포스트 코로나 시대의 기업구조조정 방안

Get AI-generated answers

Answer: 포스트 코로나 시대 기업구조조정은 단순히 비용 절감이 아닌 민첩성과 회복탄력성 강화입니다. 데이터 기반 의사결정을 통해 핵심 사업 역량에 집중하고, 디지털 전환을 가속화하여 새로운 것이 중요합니다. 동시에, 변화에 유연하게 대응할 수 있는 조직 문화를 구축하고, 인재 재배치 지속 가능한 경쟁력을 확보해야 합니다.

중소기업 금융지원 확대 전략

```
# 답변 캐싱 할수
@st.cache_data
def get_answer(question: str) -> str:
    """Generate and cache answers for the given question."""
    prompt = PromptTemplate(
        input_variables=["question"],
        template="You are an expert in interviews. Provide a concise yet insightful answer to the following question: {question}"
    ).format(question=question)
    return llm.predict(prompt)

# Streamlit 카테고리 선택
st.subheader("💼 산업은행 면접 준비 도우미")
selected_category = st.selectbox("면접 질문을 선택하시면 AI가 답변을 제공해 드립니다.", ["금융 및 경제", "ESG 금융의 활성화 방안", "산업구조 고도화를 위한 투자전략", "포스트 코로나 시대의 기업구조조정 방안", "중소기업 금융지원 확대 전략"])

# 선택된 카테고리의 질문 가져오기
for question in questions[selected_category]:
    with st.expander(question):
        # 버튼을 눌렸을 때 답변 생성
        if st.button(f"Get AI-generated answers", key=f"button_{question}"):
            with st.spinner(f"Generating answers with AI..."):
                answer = get_answer(question)
            st.session_state[f"{question}"] = f"**Answer:** {answer}"

# 세션 상태에 저장된 답변 표시
if f"{question}" in st.session_state:
    st.write(st.session_state[f"{question}"])

• 답변 평가 추후 예정
```

04 Result AI 챗봇

경제금융기업 AI 활용 취업 지원 서비스

경제 전문가 AI 챗봇

안녕하세요! 경제 전문가 AI 챗봇입니다. 질문을 입력하시면 도움을 드리겠습니다.

요즘 KDB 산업은행 동향 알려줘

KDB 산업은행의 최근 동향은 크게 정책금융 역할 강화, 혁신성장 지원, 그리고 경영효율화 추진으로 요약될 수 있습니다. 특히, 제공해주신 정보를 바탕으로 방위산업에 대한 관심이 높아지고 있음을 알 수 있습니다. 아래에서 각 동향에 대해 자세히 설명하겠습니다.

1. 정책금융기관으로서의 역할 강화:

- 산업구조 고도화 지원: 전통적인 제조업 중심에서 벗어나 4차 산업혁명 시대에 대비한 미래 신 산업 육성에 중점을 두고 있습니다. 이는 인공지능, 빅데이터, 바이오헬스, 친환경 에너지 등 혁신성장 분야에 대한 투자 확대를 의미합니다. 산업은행은 단순 자금 공급을 넘어, 기업의 성장 단계별 맞춤형 지원, 컨설팅, 네트워킹 등 종합적인 지원 체계를 구축하고 있습니다.
- 중소·중견기업 지원 강화: 경제의 허리 역할을 하는 중소·중견기업의 성장과 경쟁력 강화를 위해 자금 지원, 기술 개발, 해외 진출 등 다양한 프로그램을 운영하고 있습니다. 특히, 벤처-스타트업 투자를 확대하여 혁신 생태계 조성에도 기여하고 있습니다.
- 지역균형발전 추진: 수도권 집중 현상 완화 및 지방 경제 활성화를 위해 지방 소재 기업에 대한 금융 지원을 확대하고, 지역 특화산업 육성에도 적극적으로 참여하고 있습니다.

2. 혁신성장 지원:

- 디지털 전환 지원: 기업들의 디지털 전환을 촉진하기 위해 관련 기술 개발, 인프라 구축, 인력 양성 등에 투자를 확대하고 있습니다. 클라우드, AI, 빅데이터 등 디지털 기술 도입을 지원하여 기업의 생산성 향상과 경쟁력 강화를 돋고 있습니다.
- ESG 경영 확산: 환경(Environment), 사회(Social), 지배구조(Governance)를 중시하는 ESG 경영이 기업의 지속 가능한 성장을 위한 필수 요소로 부상함에 따라, 산업은행은 ESG 챠팅 지원, ESG 컨설팅 제공 등을 통해 기업의 ESG 경영 도입을 적극 지원하고 있습니다. 특히, 친환경 사업 및 사회적 가치 창출 기업에 대한 투자를 확대하고 있습니다.
- 방위산업 육성: 제공해주신 정보처럼, 산업은행은 방위산업에 대한 전문가 간담회를 개최하는 등 관련 분야에 대한 관심을 높이고 있습니다. 이는 방위산업의 성장 잠재력을 인지하고, 국가 안보 및 경제 성장에 기여할 수 있는 전략적 산업으로 육성하려는 의지를 보여주는 것입니다. 향후 방위산업 관련 기업에 대한 금융 지원, 기술 개발 협력 등이 예상됩니다.

3. 경영 효율화 추진:

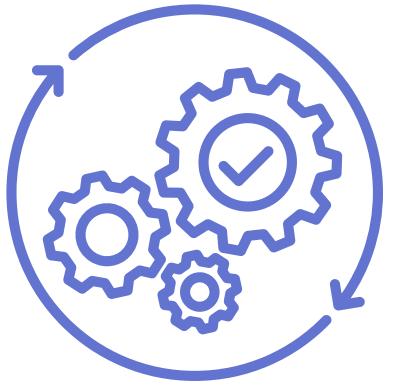
- 디지털 기반 업무 프로세스 혁신: 업무 효율성 향상 및 고객 편의 증대를 위해 디지털 기술을 적극 활용하고 있습니다. 온라인 플랫폼 강화, 빅데이터 분석 기반 의사결정 시스템 구축 등을 통해 경영 효율성을 높이고 있습니다.

질문을 입력하세요: >

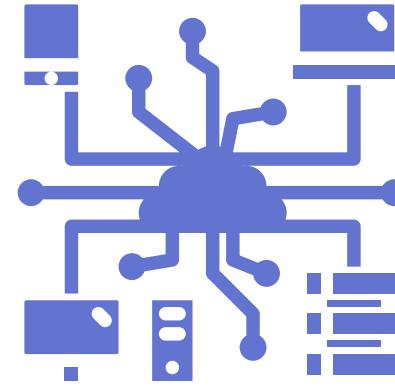
```
if user_query:  
    vector_store = st.session_state.vector_store  
    qa_chain = st.session_state.qa_chain  
    history_manager = st.session_state.history_manager  
  
    st.session_state.messages.append({"role": "user", "content": user_query})  
    with st.chat_message("user"):  
        st.markdown(user_query)  
  
    try:  
        retrieved_docs = vector_store.similarity_search(user_query, k=5)  
        documents = [  
            Document(  
                page_content=doc.page_content if hasattr(doc, 'page_content') else str(doc),  
                metadata=doc.metadata  
            )  
            for doc in retrieved_docs  
        ]  
        history_text = history_manager.to_text()  
  
        # 스트리밍된 메시지 처리  
        response = qa_chain.invoke({  
            "context": documents,  
            "question": user_query,  
            "history": history_text  
        })  
  
        st.session_state.messages.append({"role": "assistant", "content": response})  
        with st.chat_message("assistant"):  
            st.markdown(response)  
  
        history_manager.add_entry(user_query, response[-1]['text'] if isinstance(response, list) else response)  
        st.session_state.history_manager = history_manager
```

- query를 cosine similarity 기준으로 유사한 문서를 벡터DB에서 검색
- 검색된 내용을 바탕으로 답변 생성 챗봇 제공- RAG
- history 기능을 활용해 이전 대화 내용 저장 및 이후 답변에 활용

05 Expected effects



기업별 맞춤 취업
인사이트 제공



구직자
정보 접근성 향상



구인구직간
정보 비대칭 해소



실시간 및 플랫폼
확장 가능성

06 Future Work

01

자동화 시스템 구축

뉴스 기사 업데이트 자동화

기업 분석 레포트 업데이트 자동화

실시간 분석 & 분석 결과 제공

02

서비스 확장

대상 기업 분야 확대

구직 플랫폼으로의 확장

AI 이력서 & 자소서 서비스 구축

03

커뮤니티

취업 정보 공유

면접 후기 공유

광고를 활용한 수익 창출

07 Project Insight

캡슐화 도입



코드 가독성과 유지보수 문제 해결

분석 기간 조정



단기 분석으로 환율 예측 정확성 개선

모델 활용 제안



Fine-tuning 또는 도메인 특화 모델 활용

THANK YOU

CODE.

<https://github.com/cyp-ark/project1>