[슬라이드 1: 금융 뉴스 이해 플랫폼 팀명 : STJ] 안녕하세요. "금융 뉴스 이해 플랫폼" 프로젝트 기획 팀 STJ에서 발표를 맡게 된 박세영입니다.

[슬라이드 2] 오늘 발표는 총 4개의 파트로 구성됩니다. 먼저 서버 구축 환경을 소개하고, 이어서 NLP 기반 크롤링, 기술 스택 설명, GPT Open API 활용 방식, 마지막으로 향후 구현 방향에 대해 말씀드리겠습니다.

[슬라이드 4] 먼저, 플랫폼 개발에 사용된 기술 스택입니다. 서버는 Spring Boot를 기반으로 개발되었고, REST API는 Spring MVC를 활용해 구조화했습니다. 데이터는 JPA와 Hibernate로 ORM 처리하며, 데이터베이스는 MySQL을 사용합니다. 이는 유지보수와 성능 측면 모두에서 유리한 선택이었습니다.

[슬라이드 5: Spring 프레임워크] 이어서 저희가 활용한 기술 스택의 선정 이유에 대해 설명드리겠습니다. Spring 프레임워크를 사용한 이유는 자바 기반의 안정성과 확장성 때문입니다. DI와 IoC 기능을 통해 의존성 관리가 편리하고, 계층 구조 설계로 협업 시 역할 분리가 용이합니다. 또한 Boot, JPA, Security 등 생태계가 통합되어 있어 생산성이 높습니다.

[슬라이드 6: Spring MVC] Spring MVC는 컨트롤러, 서비스, 리포지토리 구조를 통해 코드의 명확성과 테스트 용이성을 확보할 수 있습니다. RESTful API 설계에 적합하고, 다양한 요청 및 응답 처리 기능을 제공해 서버 개발의 복잡성을 줄여줍니다.

[슬라이드 7: JPA] JPA는 SQL을 직접 작성하지 않고도 객체 중심으로 DB를 조작할 수 있게 도와줍니다. 엔티티 간 연관관계 설정이 용이하며, 트랜잭션 관리와 변경 감지 기능도 자동화되어 유지보수가 쉬운 구조를 만들 수 있습니다.

[슬라이드 8: MYSQL] MySQL은 오픈소스이면서도 상용 수준의 안정성과 성능을 보장하며, Spring JPA와의 호환성이 뛰어납니다. 정규화된 관계 모델링에 적합하고, 커뮤니티도 활발해 개발 시 참고 자료가 풍부합니다.

[슬라이드 9: Table 구성 – users table] table구성에 대해서 살펴보겠습니다. users 테이블은 플랫폼의 핵심 사용자 정보를 관리하는 테이블입니다. 대부분의 기능, 예를 들어 뉴스 열람, 퀴즈 응시, 단어 학습 등과 직접적으로 연결되어 있으며, 사용자 중심의 맞춤형 학습 경험을 제공하기 위해 다양한 연관 테이블과 1:N 관계로 설계되어 있습니다.

사용자는 (user\_news\_log), (user\_vocabulary), (quiz\_result), (user\_point), (user\_badge), (user\_interest\_category) 등의 활동을 하게 됩니다. 이러한 활동들은 users 테이블과 1:N 관계를 이루며, 개인의 금융 학습 성향과 실력을 정량적으로 추적하고 분석할 수 있는 구조를 형성합니다.

또한 사용자의 관심 키워드를 기반으로 외부 뉴스 크롤러와 연동되어, 맞춤형 뉴스가 자동 수집되고 개인 피드에 반영됩니다. 이 모든 데이터는 퀴즈 자동 생성, 복습 단어 추천, 포인트 보상, 게임화 기능 등과 연결되어, 학습 몰입도와 지속성을 높이는 사용자 주도형 금융 교육 루프를 구현합니다.

[슬라이드 10: Table 구성 – terms table] terms 테이블은 금융 뉴스나 퀴즈에 등장하는 금융 용어를 저장하는 핵심 테이블입니다. 이 용어들은 학습, 퀴즈, 해설 등 다양한 기능의 중심이 되기 때문에 별도로 구성되었습니다. 용어별 등장 빈도나 카테고리 분류를 통해 콘텐츠 자동 추천이나 난이도 조절이 가능하도록 설계하였습니다.

Term은 뉴스 기사 본문에서 자연어 처리 기반의 (news\_keyword)을 통해 자동 등록되며, (keyword\_trend)를 통해 트렌딩 용어로 분류됩니다. 또한 (user\_vocabulary)이나 (quiz\_term)을 통해 학습 수준별 콘텐츠 및 평가 자료로 재활용됩니다.

각각의 term은 (glossaries) 테이블과 연결되어 뉴스 본문 하이라이팅 기능과도 직결되며, 뉴스-퀴즈-단어장 간 일관된 학습 콘텐츠 단위로써 전체 플랫폼의 정보 구조를 통합하는 역할을 합니다. 이와 같은 설계는 데이터 중복을 줄이면서도 용어 중심의 학습 흐름을 체계화하고, 실시간 수집된 뉴스 데이터와 사용자 학습 콘텐츠를 유기적으로 연결함으로써 자동화된 금융 문해력 교육 시스템을 실현하고자 하는 목적을 담고 있습니다.

[슬라이드 11] 지금 보시는 이 흐름도는 저희 플랫폼의 전체 동작 과정을 시각화한 것입니다.

사용자가 로그인하면 Spring 서버에서 JWT를 발급받아 인증이 이뤄지고, 이후 뉴스 피드를 요청하면 DB에서 뉴스와 키워드 목록을 조회해 JSON 형태로 응답합니다.

뉴스 화면에서 사용자가 특정 용어를 클릭하면, DB에 용어가 있는 경우엔 해설을 바로 보여주고 관련 뉴스를 추천합니다. 만약 DB에 없다면 GPT API를 호출해 문맥 기반 설명을 제공합니다.

또한, 사용자가 요약 버튼을 클릭하면 뉴스 본문이 AI 서버로 전달되어 GPT 기반 요약이 이뤄지고, 결과는 사용자 화면에 출력됩니다. 이후 사용자는 퀴즈로 이동해 관련 내용을 복습할 수 있습니다.

이처럼 본 시스템은 뉴스 → 해설 → 요약 → 퀴즈로 이어지는 흐름을 통해 자동화된 금융 학습 루프를 제공합니다.

[슬라이드 13: 네이버 뉴스 크롤링 자동화] 네이버 뉴스의 경제 섹션은 다양한 언론사의 기사를 포괄하고 있기 때문에, 별도로 경제 기사를 필터링하지 않아도 됩니다. 크롤링 과정에서는 requests와 BeautifulSoup을 활용하여 HTML을 파싱하고 기사 본문을 추출합니다. 또한 자동화된 URL 수집 기능을 구현하여 사용자가 수동으로 기사 주소를 입력할 필요 없이, 최신 뉴스 기사의 링크를 자동으로 확보할 수 있도록 구성하였습니다. 중복 제거와 5개 제한 선택도 함께 처리되어 효율적인 데이터 수집이 가능해졌습니다.

[슬라이드 14: 단어 분리 & 조사 제거] 뉴스 본문에서 단어를 추출하는 데 있어, 한국어의 조사와 결합된 단어 구조는 분석 정확도를 떨어뜨리는 요소였습니다. 예를 들어 "전산장애로"라는 단어는 실제 의미 단위인 "전산장애"와 조사 "로"로 나뉘어야 합니다. 이를 해결하기 위해 단순 토큰화가 아닌 정규표현식과 조사 리스트를 정의하여 끝부분 조사를 제거하는 전처리를 진행하였습니다.

[슬라이드 15: 용어 해설 제공 DB 설계] 용어 해설을 위해 공공데이터포털에서 제공하는 기획재정부 시사경제용어 데이터를 활용하였습니다. CSV 데이터를 SQLite로 변환하여 dictionary 테이블로 구축하였고, term과 description 중심으로 구성된 이 데이터는 기사 내 용어 해석 제공에 활용됩니다. 보시는 것과 같이 단일 명사 용어는 높은 정확도로 해석이 가능했습니다.

[슬라이드 16: 복합 명사 처리의 한계와 개선] 하지만 "4차 산업혁명"이나 "BIS 자기자본비율"과 같은 복합어의 경우, 단순 단어 분리 방식으로는 인식이 어려웠습니다. 이를 해결하기 위해 최대 5단어까지 가능한 조합을 생성하는 방식을 도입했으나, 실행 시간이 증가하고 정확도도 떨어지는 단점이 있었습니다. 이에 따라 더 효과적인 방법이 필요하다고 판단했습니다.

[슬라이드 17: Konlpy 기반 복합어 추출] 해결책으로는 Konlpy(코엔엘파이)의 형태소 분석기를 도입하였고, 특히 Okt의 phrases() 기능을 활용해 복합 명사를 효과적으로 추출할 수 있었습니다. 이는 불필요한 단어 조합 생성을 줄이고, 실제 기사 내 의미 있는 용어와의 매칭 정확도를 높이는 데 큰 도움이 되었습니다. 실험 결과, 단어 조합 방식 대비 실행 속도와 해석의 정확성 모두에서 개선 효과를 보였습니다. 현재 보시는 이미지를 통해 기존 방식과 Konlpy 방식의 차이를 확인할 수 있습니다. Konlpy는 복합 명사를 더 잘 추출해내며, 실제 용어와의 매칭률도 높습니다.

[슬라이드 19: GPT Open API 연동] 이 화면은 GPT Open AI의 전체적인 작동 흐름을 보여주는 구조입니다. 사용자는 뉴스 본문 전체나 특정 단어를 선택하여 요청을 시작하게 되고, 시스템은 해당 요청을 바탕으로 GPT prompt를 구성합니다. 이 프롬프트는 system과 user 메시지를 조합해 문맥을 보존하도록 설계됩니다.

그 다음 OpenAI API로 요청을 전송하면, GPT 모델이 해당 단어의 의미를 문맥에 맞게 해석하거나 뉴스 전체 내용을 요약해 응답을 반환합니다. 이 응답은 Python에서 JSON 형태로 파싱되며, 결과는 DB에 저장되거나 사용자 화면에 실시간으로 표시됩니다. 이렇게 OpenAI는 뉴스 이해를 위한 핵심 처리 모듈로 동작하게 됩니다

[슬라이드 20: 한계 & 개선점 & 고려사항] OpenAI를 사용할 때는 몇 가지 중요한 고려사항이 있습니다. 첫째, 요금 문제입니다. OpenAI는 토큰 단위로 과금되기 때문에 반복적인 호출 시 비용 부담이 커질 수 있습니다. 따라서 해설과 요약 요청을 분리하고, 캐싱을 통해 중복 호출을 줄이는 전략이 필요합니다. 둘째, 실시간 응답 속도입니다. 특히 사용자 요청에 즉각 반응해야 하는 경우에는 대기 시간이 사용자 경험에 영향을 줄 수 있습니다. 셋째, 문맥 판단의 한계입니다. 모델이 항상 정확한 의미를 판단하지 못할 수 있으므로 후처리와 검증 절차가 필요합니다.

[슬라이드 21: 임베딩 활용 및 고려사항] 임베딩은 단어를 벡터로 변환해 의미를 수치적으로 표현하는 기술입니다. GPT나 BERT(버트)와 같은 모델은 문맥을 반영한 임베딩을 생성하며, 이를 통해 단어 간 유사성, 문장 구조, 의미적 연결 등을 파악할 수 있습니다. 활용 예로는 개체명 인식(NER), 문서 간 의미 유사도 비교, 의미 기반 검색, 개인화 추천 시스템 등이 있습니다. 그러나 Out-of-Vocabulary 단어에 대한 처리, 서브워드 기반 토큰화, 문자 단위 임베딩 등의 고려가 필요하며, OpenAI에서는 추가 데이터로 임베딩을 재학습하는 기능은 제공되지 않기 때문에 보완이 필요합니다.

문맥적 임베딩은 같은 단어라도 문장에 따라 다른 의미를 가질 수 있도록 반영해주는 기술입니다. GPT나 BERT(버트)같은 모델은 이러한 문맥 기반의 벡터를 생성할 수 있어, 문장 내 단어의 정확한 의미를 포착하는 데 유리합니다. 예를 들어 텍스트 분류에서는 뉴스 기사를 벡터로 표현한 후 이를 기반으로 금융, 정치, 스포츠 등으로 분류할 수 있습니다. 이처럼 임베딩 기술은 뉴스 분석, 검색, 추천 등 다양한 기능에 확장 적용될 수 있습니다.

[슬라이드 23] :

보시는 화면은 저희 플랫폼의 IA를 구상한 것입니다. 핵심 기능 2가지 중 첫 번째 기능인 뉴스 플랫폼의 기술을 구현 중이며, 실행 속도와 효율성, 그리고 단어 추출의 정확성을 증가시키기 위해 노력하고 있습니다. 앞으로 저희 팀은 두 번째 핵심 기능인 교육 플랫폼 구현에 집중할 예정입니다 이 플랫폼은 사용자 맞춤형 퀴즈 제공과 반복 학습을 통해 학습 효과를 높이는 데 목적이 있습니다 구체적으로는 사용자 데이터를 기반으로 난이도별 퀴즈를 추천하고 퀴즈 정답 시 포인트 및 레벨업 기능을 제공하여 동기를 부여하며 틀린 문제는 반복 노출과 해설 제공을 통해 학습효과를 높일 예정입니다. 또한 뉴스와 교육 기능을 자연스럽게 연결하여 콘텐츠 간 이동과 학습 흐름을 끊김 없이 이어갈 수 있도록 설계할 계획입니다.