aws FOR GAMES

GAMES ON AWS 2026

GS Neotek | MEGAZONE CLOUD

GAMES ON AWS 2026

aws FOR GAMES

유저 데이터 기반 초개인화 소셜 추천 만들기

함께할 사람을 찾아서

송정우

솔루션즈 아키텍트

AWS

구지민

소셜메이킹팀 리더

NEXON KOREA

GS Neotek | MEGAZONE CLOUD

목차

- 벡터 검색 on AWS
- 넥슨의 유저 데이터 기반 소셜 추천
- 더 나은 초개인화를 향해
- Lessons Learned

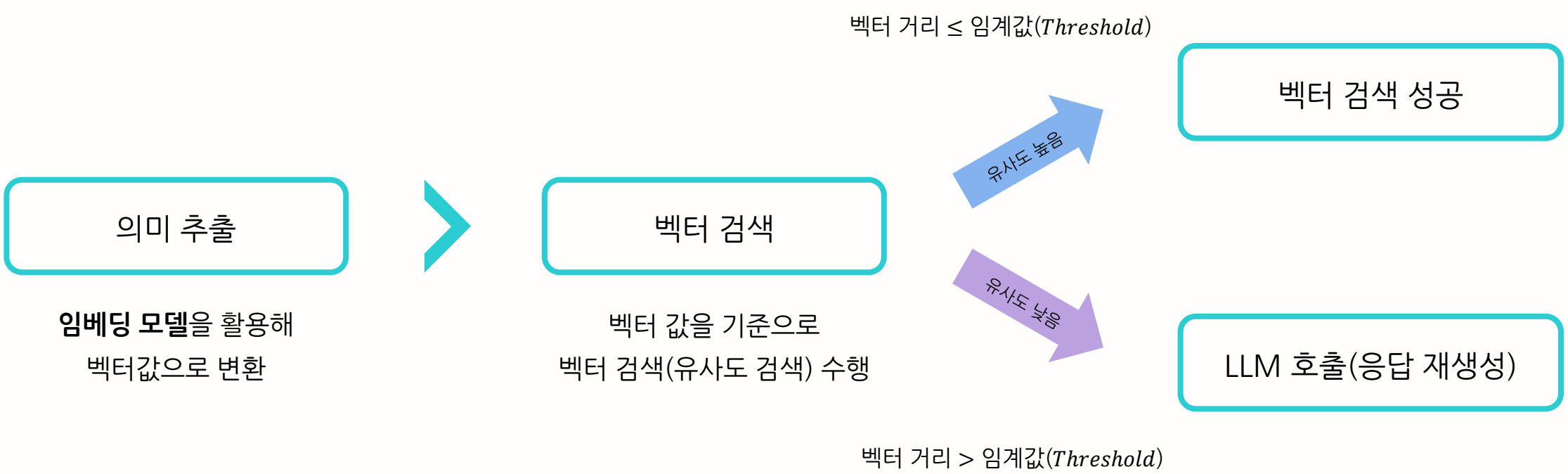
GAMES ON AWS 2026

aws FOR GAMES

벡터 검색 on AWS

GS Neotek | MEGAZONE CLOUD

벡터 검색이란?


의미 추출
임베딩 모델을 활용해
벡터값으로 변환
벡터 검색
벡터 값을 기준으로
벡터 검색(유사도 검색) 수행
벡터 거리 ≤ 임계값(Threshold)
유사도 높음
벡터 검색 성공
유사도 낮음
LLM 호출(응답 재생성)
벡터 거리 > 임계값(Threshold)

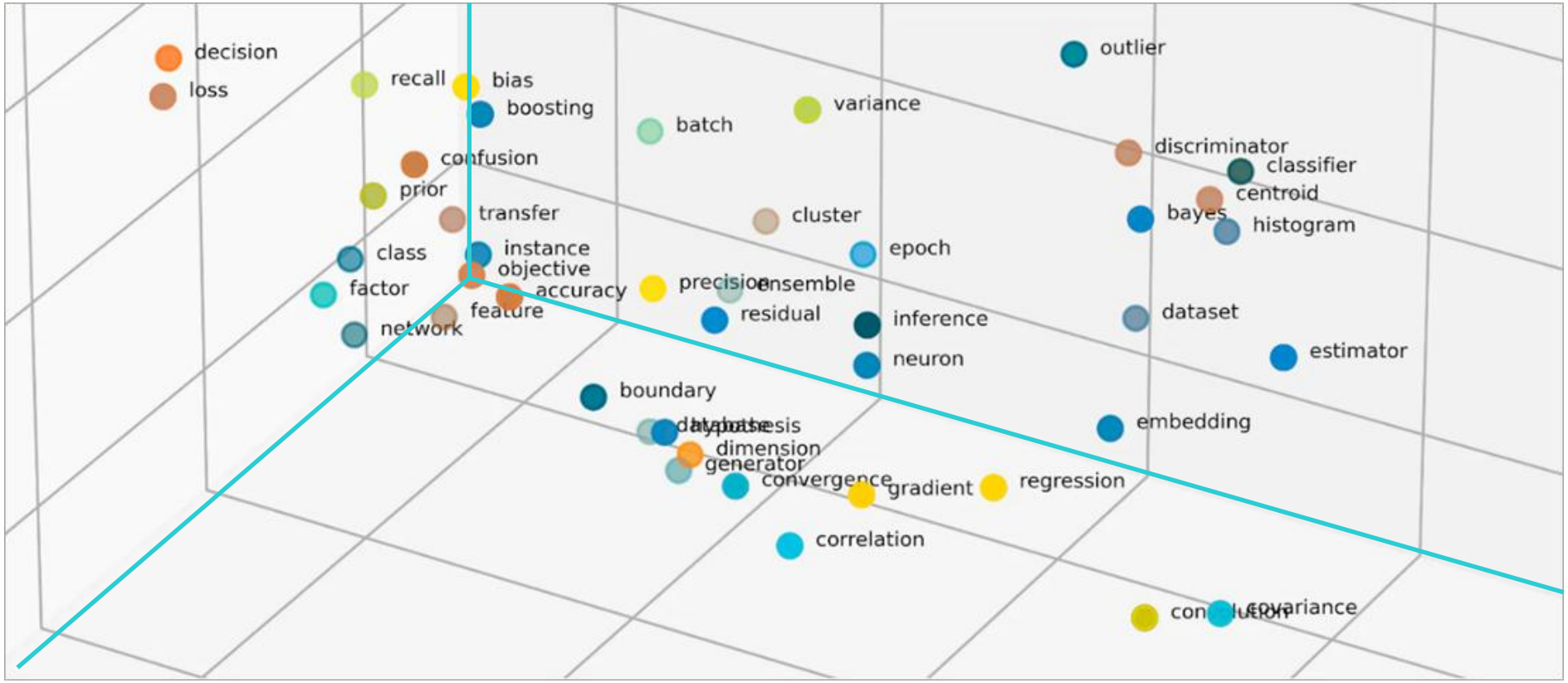
임베딩 모델

벡터 검색의 성능을 결정하는 모델

Amazon Bedrock

오픈소스 임베딩 모델

커스텀 임베딩 모델
(on Amazon SageMaker)



Amazon SageMaker AI와 Amazon Bedrock

Amazon SageMaker AI

AI 모델 개발

자체 모델을 빌드하고 커스터마이즈

자체 모델 배포 및 추론

MLOps

자체 모델을 SageMaker에서, Bedrock에서 다른 모델들을 함께

SageMaker에서 커스터마이징한 모델 혹은 기존 모델을 운영

Bedrock을 통해 에이전트 워크 플로우를 빌드하고 운영하기

Amazon Bedrock

애플리케이션 개발

에이전트 워크 플로우

실시간 추론 & 배치 추론

API 콜에 기반한 Claude, OpenAI, 3P 모델들

벡터 검색 on AWS



Amazon Aurora PostgreSQL

SQL-native vector search

관계형



Amazon Neptune

Knowledge graph + vectors

그래프



Amazon ElastiCache

Microsecond latency

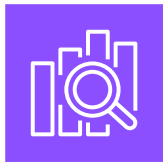
시맨틱 캐싱



Amazon S3 Vectors

Up to 90% cost savings

비용 최적화



Amazon OpenSearch Service

Lexical, semantic, hybrid, and agentic search, broadest capability set, including serverless

범용 목적



Amazon DynamoDB

Single-digit millisecond latency at any scale

서버리스

벡터 검색 on AWS



Amazon Aurora PostgreSQL

SQL-native vector search

관계형



Amazon Neptune

Knowledge graph + vectors

그래프



Amazon ElastiCache

Microsecond latency

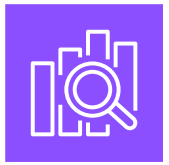
시맨틱 캐싱



Amazon S3 Vectors

Up to 90% cost savings

비용 최적화



Amazon OpenSearch Service

Lexical, semantic, hybrid, and agentic search, broadest capability set, including serverless

범용 목적



Amazon DynamoDB

Single-digit millisecond latency at any scale

서버리스

GAMES ON AWS 2026

aws FOR GAMES

넥슨의 유저 데이터 기반 소셜 추천

GS Neotek | MEGAZONE CLOUD

소셜메이킹팀은

게임 속에서 **함께할 사람**을 찾아주는 일을 합니다

함께할 사람 추천

친구 길드 파티

친구, 길드, 파티 등 게임 내에서
함께할 수 있는 유저를 추천하는 서비스를 운영

양방향 추천

유저 ⇄ 커뮤니티

유저에게 커뮤니티를 추천하거나,
커뮤니티와 잘 어울리는 유저를 추천하는 등
양방향의 서비스를 제공

몰입과 리텐션

게임 몰입도 리텐션

유저 간 관계 형성을 통해 게임 몰입도를 높이고,
리텐션 향상에 기여하는 역할을 수행



※ 이해를 돕기 위한 예시 화면입니다

관계가 만드는 변화 → 추천 → 관계 형성 → 몰입도 향상 → 리텐션 개선

소셜메이킹팀은

게임 속에서 **함께할 사람**을 찾아주는 일을 합니다

함께할 사람 추천

친구 · 길드 · 파티

친구, 길드, 파티 등 게임 내에서
함께할 수 있는 유저를 추천하는 서비스를 운영

양방향 추천

유저 ⇄ 커뮤니티

유저에게 커뮤니티를 추천하거나,
커뮤니티와 잘 어울리는 유저를 추천하는 등
양방향의 서비스를 제공

몰입과 리텐션

게임 몰입도 · 리텐션

유저 간 관계 형성을 통해 게임 몰입도를 높이고,
리텐션 향상에 기여하는 역할을 수행



※ 이해를 돕기 위한 예시 화면입니다

관계가 만드는 변화 : 추천 → 관계 형성 → 몰입도 향상 → 리텐션 개선

소셜 추천과 데이터

게임 별 플레이 이력과 넥슨 플랫폼 정보를 함께 활용

데이터	무엇을 보나	출처	반영 방식
플랫폼 정보 · 게임 성향	선호하는 게임 리스트 주 활동 시간대, 즐기는 콘텐츠, 플레이 스타일	넥슨 플랫폼 정보 게임별 수집 정보	배치 **주기적으로 피처 생성**
소셜 인터랙션	파티 플레이, 길드 · 클랜 활동, 같은 맵에서의 플레이	게임별 수집 정보	배치 **관계 그래프 주기 갱신**
커뮤니티 데이터	길드 홍보 · 파티 구인 글의 텍스트와 이미지  이미지 속 문구 오픈채팅 가능 자유로운 분위기	게임 공식 홈페이지	배치 **Amazon Bedrock 을 활용한 특성 추출**
실시간 상태	접속 상태, 최근 추천 이력	게임 서버 실시간 이벤트	실시간 **요청 시점에 바로 반영**

※ 길드원 모집 이미지는 이해를 돕기 위한 예시입니다

소셜 추천과 데이터

게임 별 플레이 이력과 넥슨 플랫폼 정보를 함께 활용

데이터	무엇을 보나	출처	반영 방식
플랫폼 정보 · 게임 성향	선호하는 게임 리스트 주 활동 시간대, 즐기는 콘텐츠, 플레이 스타일	넥슨 플랫폼 정보 게임별 수집 정보	배치 **주기적으로 피처 생성**
소셜 인터랙션	파티 플레이, 길드 · 클랜 활동, 같은 맵에서의 플레이	게임별 수집 정보	배치 **관계 그래프 주기 갱신**
커뮤니티 데이터	길드 홍보 · 파티 구인 글의 텍스트와 이미지  이미지 속 문구 오픈채팅 가능 자유로운 분위기	게임 공식 홈페이지	배치 **Amazon Bedrock 을 활용한 특성 추출**
실시간 상태	접속 상태, 최근 추천 이력	게임 서버 실시간 이벤트	실시간 **요청 시점에 바로 반영**

※ 길드원 모집 이미지는 이해를 돕기 위한 예시입니다

소셜 추천과 데이터

게임 별 플레이 이력과 넥슨 플랫폼 정보를 함께 활용

데이터	무엇을 보나	출처	반영 방식
플랫폼 정보 · 게임 성향	선호하는 게임 리스트 주 활동 시간대, 즐기는 콘텐츠, 플레이 스타일	넥슨 플랫폼 정보 게임별 수집 정보	배치 **주기적으로 피처 생성**
소셜 인터랙션	파티 플레이, 길드 · 클랜 활동, 같은 맵에서의 플레이	게임별 수집 정보	배치 **관계 그래프 주기 갱신**
커뮤니티 데이터	길드 홍보 · 파티 구인 글의 텍스트와 이미지  이미지 속 문구 오픈채팅 가능 자유로운 분위기	게임 공식 홈페이지	배치 **Amazon Bedrock 을 활용한 특성 추출**
실시간 상태	접속 상태, 최근 추천 이력	게임 서버 실시간 이벤트	실시간 **요청 시점에 바로 반영**

※ 길드원 모집 이미지는 이해를 돕기 위한 예시입니다

소셜 추천과 데이터

게임 별 플레이 이력과 넥슨 플랫폼 정보를 함께 활용

데이터	무엇을 보나	출처	반영 방식
플랫폼 정보 · 게임 성향	선호하는 게임 리스트 주 활동 시간대, 즐기는 콘텐츠, 플레이 스타일	넥슨 플랫폼 정보 게임별 수집 정보	배치 **주기적으로 피처 생성**
소셜 인터랙션	파티 플레이, 길드 · 클랜 활동, 같은 맵에서의 플레이	게임별 수집 정보	배치 **관계 그래프 주기 갱신**
커뮤니티 데이터	길드 홍보 · 파티 구인 글의 텍스트와 이미지  이미지 속 문구 오픈채팅 가능 자유로운 분위기	게임 공식 홈페이지	배치 **Amazon Bedrock 을 활용한 특성 추출**
실시간 상태	접속 상태, 최근 추천 이력	게임 서버 실시간 이벤트	실시간 **요청 시점에 바로 반영**

※ 길드원 모집 이미지는 이해를 돕기 위한 예시입니다

소셜 추천과 데이터

게임 별 플레이 이력과 넥슨 플랫폼 정보를 함께 활용

데이터	무엇을 보나	출처	반영 방식
플랫폼 정보 · 게임 성향	선호하는 게임 리스트 주 활동 시간대, 즐기는 콘텐츠, 플레이 스타일	넥슨 플랫폼 정보 게임별 수집 정보	배치 **주기적으로 피처 생성**
소셜 인터랙션	파티 플레이, 길드 · 클랜 활동, 같은 맵에서의 플레이	게임별 수집 정보	배치 **관계 그래프 주기 갱신**
커뮤니티 데이터	길드 홍보 · 파티 구인 글의 텍스트와 이미지  → 이미지 속 문구: 오픈채팅 가능, 자유로운 분위기	게임 공식 홈페이지	배치 **Amazon Bedrock 을 활용한 특성 추출**
실시간 상태	접속 상태, 최근 추천 이력	게임 서버 실시간 이벤트	실시간 **요청 시점에 바로 반영**

※ 길드원 모집 이미지는 이해를 돕기 위한 예시입니다

소셜 추천에서 풀어야 할 과제

모두가 만족하는 추천을 찾아서

01
여러 추천 방향을 고려해야 합니다

유저 → 유저	친구 · 멘토 · 멘티
유저 → 커뮤니티	길드 · 클랜 · 파티
커뮤니티 → 유저	길드원 · 클랜원 · 파티원
커뮤니티 → 커뮤니티	길드 연합 · 스쿼드 연합

과제

방향마다 따로 만들지 않고
하나의 구조로 처리하기

→ 양방향 추천의 범용화

02
비교해야 할 조합이 많습니다

수십~수백만 명 유저

×

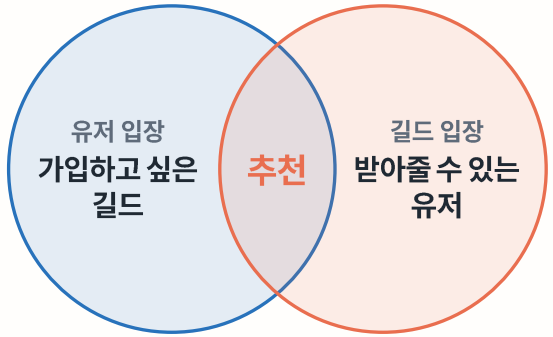
수백~수십만 개 커뮤니티

과제

전체를 비교하지 않고
빠르게 후보를 추리기

→ 임베딩 생성 시점 분리

03
양쪽이 모두 원해야 합니다



과제

수락 · 함께 활동할 가능성까지 보고
추천 순서를 정하기

→ 필터링과 재정렬

소셜 추천에서 풀어야 할 과제

모두가 만족하는 추천을 찾아서

01
여러 추천 방향을 고려해야 합니다

유저 → 유저	친구 · 멘토 · 멘티
유저 → 커뮤니티	길드 · 클랜 · 파티
커뮤니티 → 유저	길드원 · 클랜원 · 파티원
커뮤니티 → 커뮤니티	길드 연합 · 스쿼드 연합

과제

방향마다 따로 만들지 않고
하나의 구조로 처리하기

→ 양방향 추천의 범용화

02
비교해야 할 조합이 많습니다

수십~수백만 명 유저

×

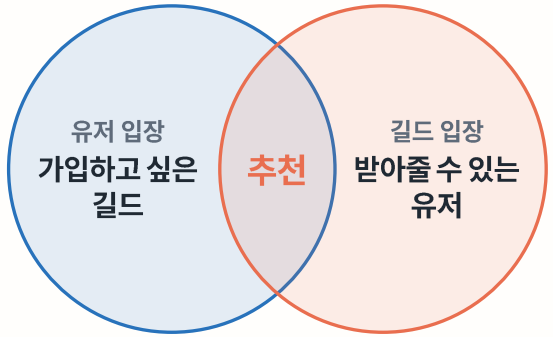
수백~수십만 개 커뮤니티

과제

전체를 비교하지 않고
빠르게 후보를 추리기

→ 임베딩 생성 시점 분리

03
양쪽이 모두 원해야 합니다



과제

수락 · 함께 활동할 가능성까지 보고
추천 순서를 정하기

→ 필터링과 재정렬

소셜 추천에서 풀어야 할 과제

모두가 만족하는 추천을 찾아서

01
여러 추천 방향을 고려해야 합니다

유저 → 유저	친구 · 멘토 · 멘티
유저 → 커뮤니티	길드 · 클랜 · 파티
커뮤니티 → 유저	길드원 · 클랜원 · 파티원
커뮤니티 → 커뮤니티	길드 연합 · 스쿼드 연합

과제

방향마다 따로 만들지 않고
하나의 구조로 처리하기

→ 양방향 추천의 범용화

02
비교해야 할 조합이 많습니다

수십~수백만 명 유저

×

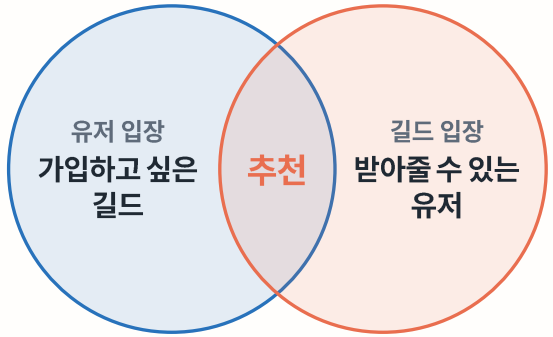
수백~수십만 개 커뮤니티

과제

전체를 비교하지 않고
빠르게 후보를 추리기

→ 임베딩 생성 시점 분리

03
양쪽이 모두 원해야 합니다



과제

수락 · 함께 활동할 가능성까지 보고
추천 순서를 정하기

→ 필터링과 재정렬

소셜 추천에서 풀어야 할 과제

모두가 만족하는 추천을 찾아서

01
여러 추천 방향을 고려해야 합니다

유저 → 유저	친구 · 멘토 · 멘티
유저 → 커뮤니티	길드 · 클랜 · 파티
커뮤니티 → 유저	길드원 · 클랜원 · 파티원
커뮤니티 → 커뮤니티	길드 연합 · 스쿼드 연합

과제

방향마다 따로 만들지 않고
하나의 구조로 처리하기

→ 양방향 추천의 범용화

02
비교해야 할 조합이 많습니다

수십~수백만 명 유저

×

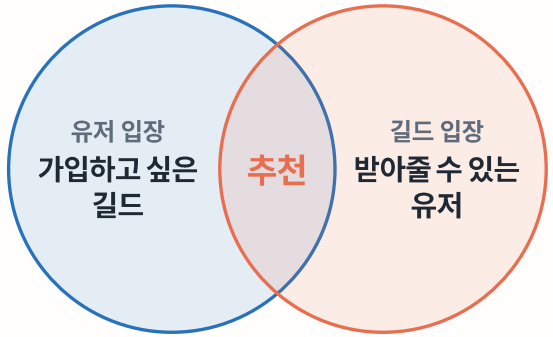
수백~수십만 개 커뮤니티

과제

전체를 비교하지 않고
빠르게 후보를 추리기

→ 임베딩 생성 시점 분리

03
양쪽이 모두 원해야 합니다



과제

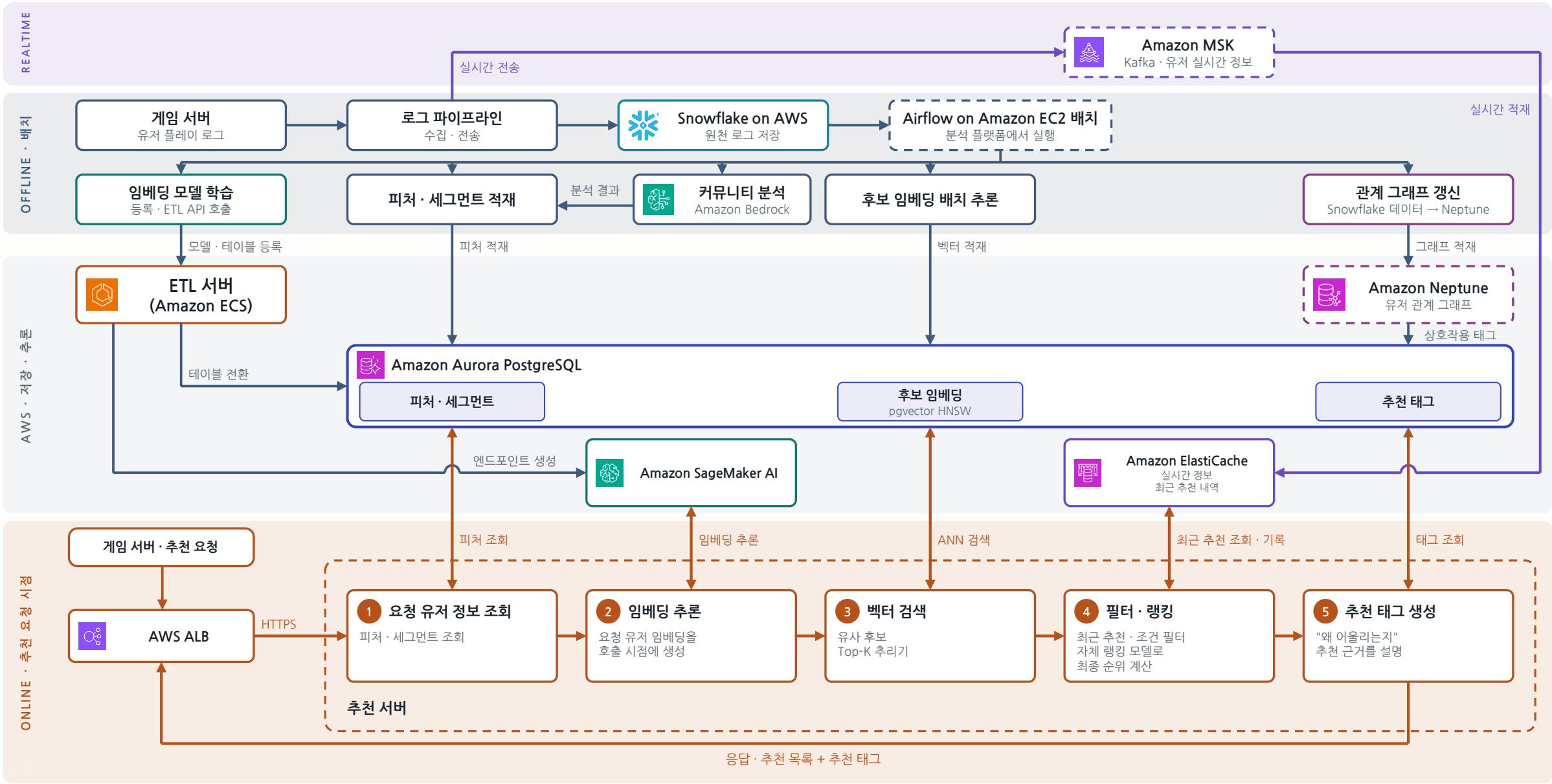
수락 · 함께 활동할 가능성까지 보고
추천 순서를 정하기

→ 필터링과 재정렬

aws FOR GAMES

GAMES ON AWS 2026

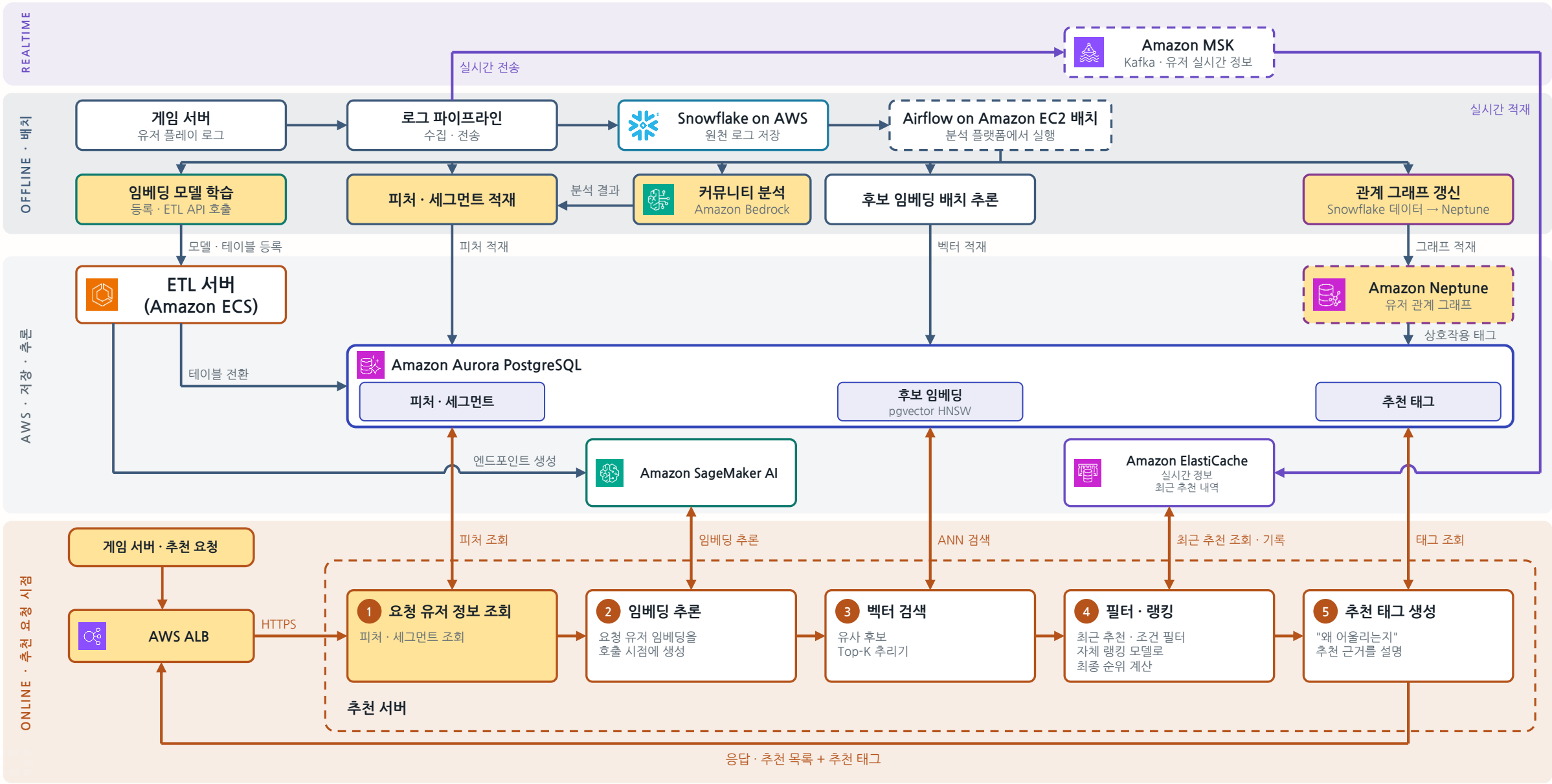
GS Neotek | MEGAZONE CLOUD



aws FOR GAMES

GAMES ON AWS 2026

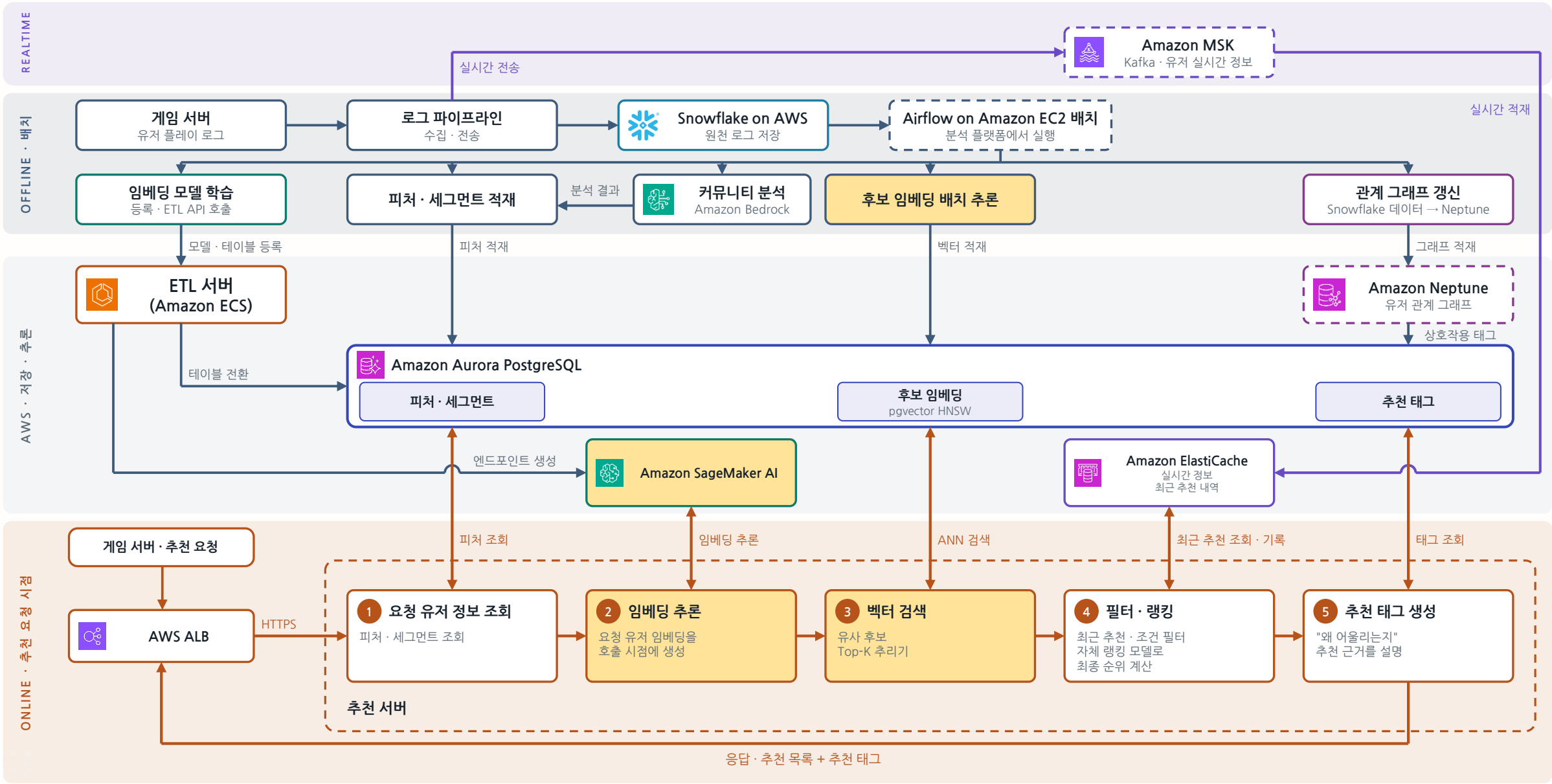
GS Neotek | MEGAZONE CLOUD



aws FOR GAMES

GAMES ON AWS 2026

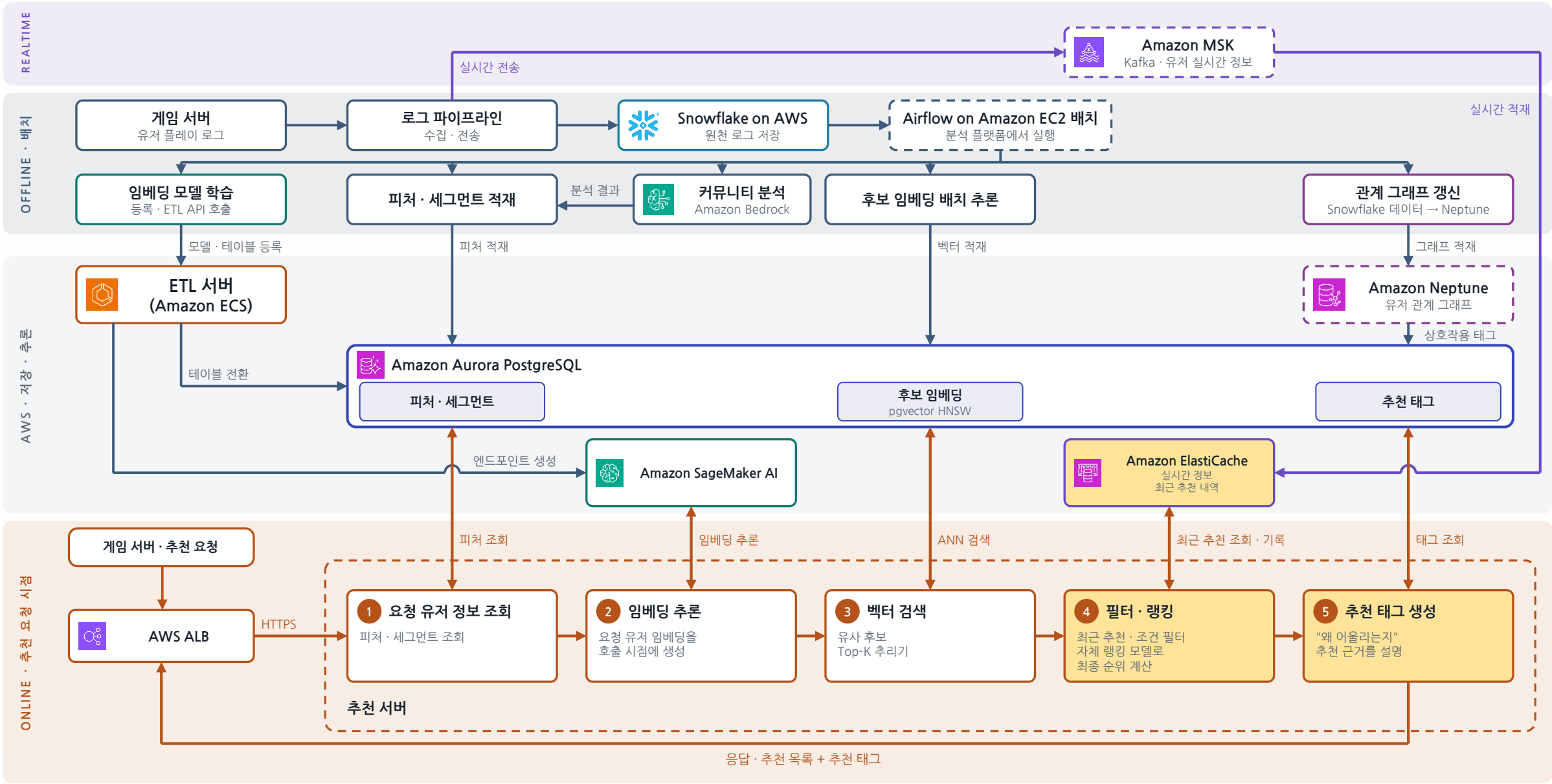
GS Neotek | MEGAZONE CLOUD



aws FOR GAMES

GAMES ON AWS 2026

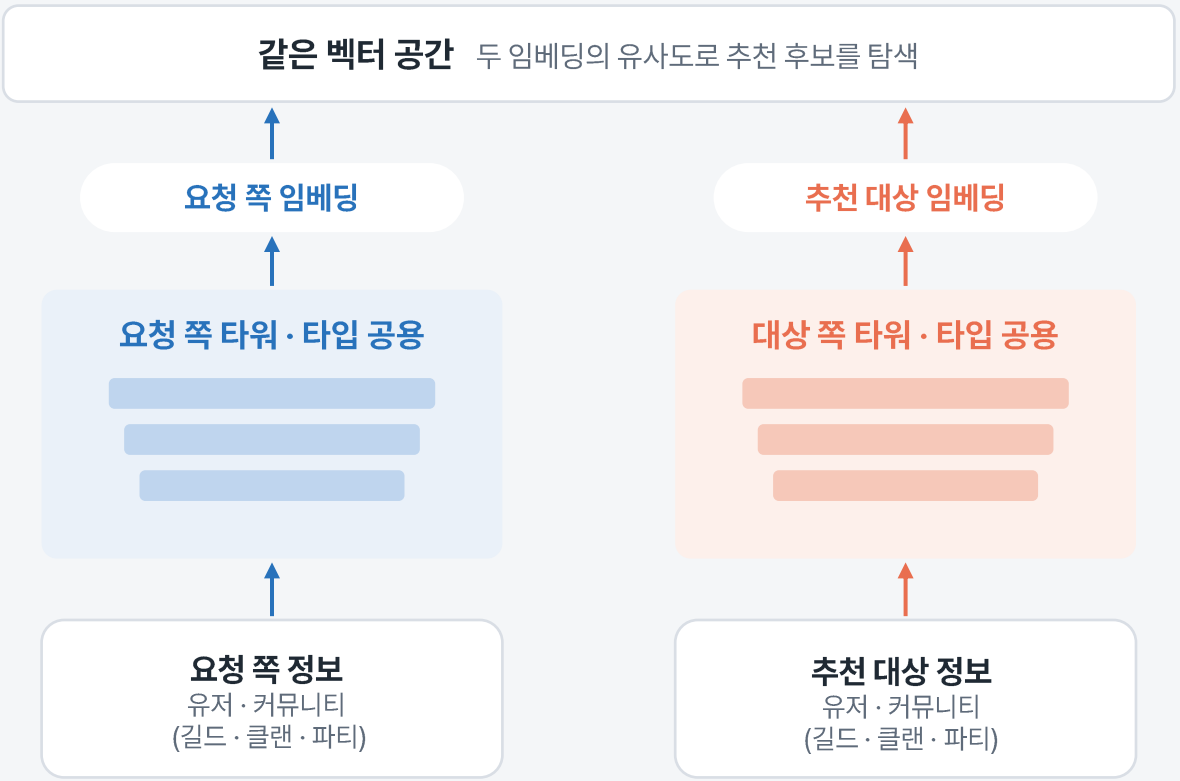
GS Neotek | MEGAZONE CLOUD



과제 1

양방향 추천의 범용화

유저와 커뮤니티, 어떤 방향의 조합이든 하나의 모델 구조로



양방향성의 추천 형태를 범용화할 수 있을까?

- two-tower 기반의 모델로 유저와 유저, 혹은 유저와 커뮤니티의 정보를 학습하도록 설계
- 추천 대상의 타입에 맞게 모델이 매번 늘어나지 않도록 제어

가능한 추천 방향

- 유저 → 유저
- 유저 → 커뮤니티
- 커뮤니티 → 유저
- 커뮤니티 → 커뮤니티

임베딩은 이렇게 만듭니다

유저와 커뮤니티를 같은 공간의 좌표로

① 입력

유저

- 플랫폼 정보
- 게임 내 성향

커뮤니티

- 활동 시간대
- 구성원 성향
- 홍보 글 분석 결과

→

② 학습

●● 가깝게 positive

- 길드 가입
- 친구 신청 수락

● ● 멀게 negative

- 길드 탈퇴
- 친구 해제

→

③ 출력

N차원 벡터

```
[ 0.21,  -0.53,  0.08,  …  ]
```

- 유저와 커뮤니티가 같은 공간의 좌표로 표현
- 가까울수록 함께할 가능성이 높은 대상

예시

달 **유저 : 달빛궁수** 저녁 접속 · 레이드 선호 → `[ 0.21,  -0.53,  0.08,  … ]`

별 **길드 : 별빛서약** 주 2회 레이드 · 저녁 활동 → `[ 0.19,  -0.48,  0.11,  … ]`

가까운 좌표 → 추천

※ 이해를 돕기 위한 예시 값입니다

임베딩은 이렇게 만듭니다

유저와 커뮤니티를 같은 공간의 좌표로

① 입력

유저

- 플랫폼 정보
- 게임 내 성향

커뮤니티

- 활동 시간대
- 구성원 성향
- 홍보 글 분석 결과

② 학습

●● 가깝게 positive

- 길드 가입
- 친구 신청 수락

● ● 멀게 negative

- 길드 탈퇴
- 친구 해제

③ 출력

N차원 벡터

```
[ 0.21,  -0.53,  0.08,  …  ]
```

- 유저와 커뮤니티가 같은 공간의 좌표로 표현
- 가까울수록 함께할 가능성이 높은 대상

예시

달 **유저 : 달빛궁수** 저녁 접속 · 레이드 선호 → `[ 0.21,  -0.53,  0.08,  … ]`

별 **길드 : 별빛서약** 주 2회 레이드 · 저녁 활동 → `[ 0.19,  -0.48,  0.11,  … ]`

가까운 좌표 → 추천

※ 이해를 돕기 위한 예시 값입니다

과제 2

임베딩 생성 시점 분리

추천 후보는 미리, 요청 유저는 실시간으로

사전 배치

추천 후보 데이터
Snowflake 가공 데이터

→

배치 추론
SageMaker ETL 엔드포인트

→

후보 임베딩 적재
추천 후보만 미리 계산

요청 시점

추천 요청
요청 유저

→

피처 조회
요청 유저 정보

→

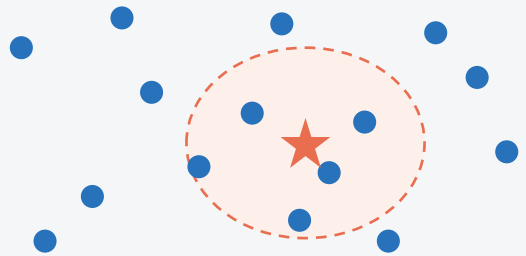
실시간 추론
SageMaker 엔드포인트

→

요청 유저 임베딩

→

Amazon Aurora PostgreSQL
같은 벡터 공간에서 ANN 검색



● 후보 임베딩 ★ 요청 유저 임베딩

가까운 후보 Top-K 추출 → 재정렬

- 추천 후보들의 임베딩 벡터는 사전 배치 작업으로 저장
- 추천을 요청하는 유저의 정보는 요청 시점에 SageMaker로 실시간 추론해 비교
- 모든 후보와 비교하지 않고, ANN 인덱스로 가까운 후보만 빠르게 검색

→ 추천을 요청하는 시점의 최근 정보를 활용하여 임베딩을 추출하기 위한 목적

과제 3

필터링과 재정렬

기획 요소를 반영하고, 양방향 모두가 만족할 수 있는 순서로 정렬

후보 Top-K
Aurora pgvector 검색 결과

→

① 필터 · 조건 반영

- **접속 중인 유저 반영** ElastiCache 실시간 상태
- **이전 추천 결과 제외** ElastiCache 추천 이력
- **기획 조건 반영** 서비스별 기획 요소

→

② 재정렬 (re-ranking)

양쪽 관점에서 다시 평가

- 상대가 받아줄 가능성 · 신청 수락률
- 가입 후 함께 활동할 가능성

학습된 모델 · 점수 규칙

→

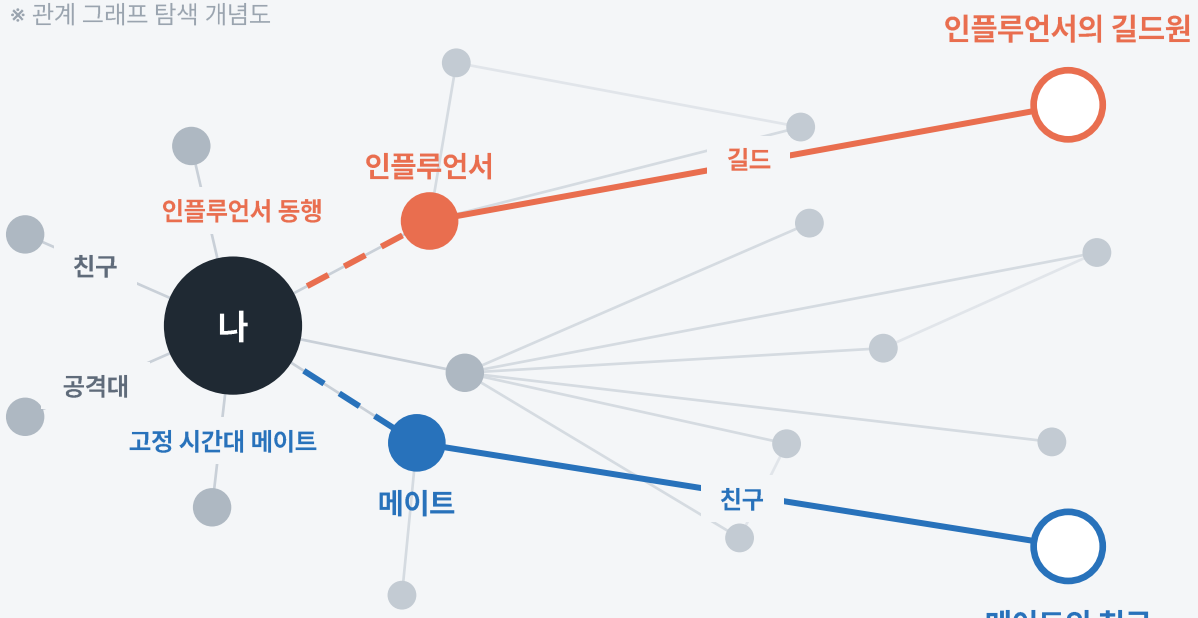
최종 추천
추천 목록
+ 추천 태그

- 임베딩 검색은 "어울리는 후보"를, 재정렬은 "양쪽이 모두 원하는 후보"를 선별하는 단계
- 접속 상태 & 추천 이력 & 기획 조건을 반영하여 추천 만족도를 높이는 전략

관계 그래프와 추천 태그

Amazon Neptune으로 유형 · 무형의 관계를 저장하고 탐색

※ 관계 그래프 탐색 개념도



유형의 관계 게임 콘텐츠에서 정의한 관계 (길드, 친구, 공격대 등)

무형의 관계 자체적으로 정의한 유저 간 관계 (인플루언서 동행, 고정 시간대 메이트)

관계를 그래프로 저장하는 이유

인플루언서의 길드원, 메이트의 친구 등
한 단계 건너 이어지는 관계를 빠르게 탐색

게임에서 정의한 유형의 관계뿐 아니라,
자체적으로 정의한 무형의 관계를 엣지로 추가하여
추천 품질을 개선

탐색 결과는 이렇게 추천 이유로 보여줍니다

달 **달빛궁수**
같은 시간대에 함께 플레이하는 새벽사냥꾼님의 친구예요

AWS 위에서 완성한 초개인화 소셜 추천

로그 수집부터 추천 응답까지

Snowflake on AWS
로그 저장

Amazon MSK
실시간 수집

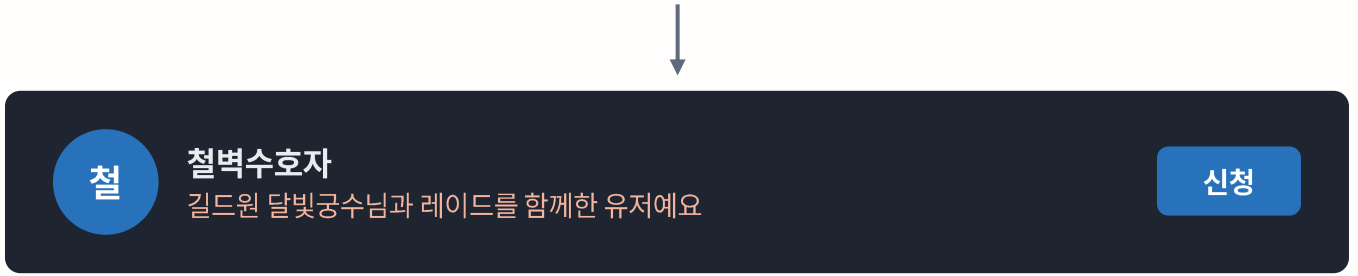
Amazon SageMaker AI
임베딩 학습 · 추론

Amazon Aurora PostgreSQL
후보 검색

Amazon Neptune
관계 · 추천 태그

Amazon ElastiCache
실시간 상태

Amazon ECS
추천 서버



※ 예시 화면

수집 · 학습 · 검색 · 관계 탐색 · 실시간 상태가 하나로 이어져, 왜 어울리는지까지 담긴 추천 결과를 완성

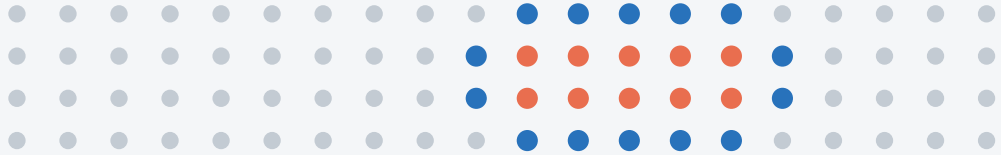
구조를 바꾸며 달라진 점

전체 탐색에서 후보 추출 방식으로

이전 요청마다 유저 전체와 비교



지금 벡터 검색으로 후보를 추린 뒤 재정렬



① 더 넓게 찾기

같은 응답 속도 수준에서 탐색하는 후보 규모

3~4만 개 → **500만 개**

기존 비교 범위 / 현재 검색 규모

② 더 오래 함께하기

추천을 통한 유저 몰입도 · 리텐션 개선

게임 A **리텐션**

약 **+4%p**

추천 경험 그룹

게임 B **게임 활동률**

+3%p 이상

추천 커뮤니티 가입 유저

게임 C **커뮤니티 이탈률**

약 **−7%p**

추천 커뮤니티 가입 유저

※ 미추천 그룹 대비 · 게임명 비공개

더 나은 초개인화를 향해

더 나은 초개인화를 위해

추천 태그 자동화: 추천의 이유를 더 자연스럽게

태그화할 수 있는 정보가 늘어남

- 같은 길드
- 파티 플레이 3회
- 같은 시간대 접속
- 레이드 선호
- 스승 관계
- 오픈채팅 가능
- 같은 맵 플레이
- + 새로운 상호작용

→

키워드 조합 · 문장 생성

태그를 엮어 추천 이유를
자연스러운 문장으로 만들기

같은 길드 + 파티 3회
+ 레이드 선호

→

추천 이유 노출 예시

지금 같은 길드 파티 3회

앞으로

"같은 길드에서 세 번 함께 파티했던 유저예요. 주로 같은 시간대에 레이드를 즐겨요"

- 다양한 유저 간 상호작용이 추가됨에 따라 태그화할 수 있는 유저 정보가 늘어날 예정
- 키워드 간의 조합이나 문장을 자연스럽게 생성하여 추천의 이유를 효과적으로 노출하는 방안을 도입

게임을 넘나드는 추천으로

공용 임베딩 모델 : 하나의 공간, 게임 별 특성은 그대로

AS-IS 게임별 독립 모델

게임 A	게임 B	게임 C
모델 A	**모델 B**	**모델 C**

- 게임이 늘면 학습 · 관리 부담도 함께 증가
- 한 게임에서 배운 성향을 다른 게임과 공유하지 못함
- 신규 게임은 데이터가 쌓일 때까지 추천 공백

→

TO-BE 공용 백본 + 게임별 어댑터

어댑터 A	어댑터 B	어댑터 C	범용 어댑터
게임 A	게임 B	게임 C	신규 게임 D

공용 백본 모든 게임이 공유 · 여러 게임에 걸친 공통 성향

신규 게임은 범용 어댑터로 바로 시작하고,
데이터가 쌓이면 게임 전용 어댑터로 자연스럽게 전환

운영 효율화
공통은 한 번만 학습하고, 게임별로는 어댑터만 관리

종합 성향 반영
여러 게임을 즐기는 유저의 성향을 플랫폼 단위로 반영

신규 게임 확장
출시 첫날부터 추천을 시작해 콜드 스타트 완화

Lessons Learned

Lessons Learned

소셜 추천 개발 과정에서 얻은 것들

데이터 성격에 맞는 경로

천천히 변하는 데이터는 배치로, 접속 상태는 실시간으로 반영

소셜 추천과 데이터

Lessons Learned

소셜 추천 개발 과정에서 얻은 것들

데이터 성격에 맞는 경로	천천히 변하는 데이터는 배치로, 접속 상태는 실시간으로 반영	소셜 추천과 데이터
방향이 바뀌어도 같은 구조	two-tower 모델 기반으로, 유연한 추천 확장이 가능	양방향 추천의 범용화

Lessons Learned

소셜 추천 개발 과정에서 얻은 것들

데이터 성격에 맞는 경로	천천히 변하는 데이터는 배치로, 접속 상태는 실시간으로 반영	소셜 추천과 데이터
방향이 바뀌어도 같은 구조	two-tower 모델 기반으로, 유연한 추천 확장이 가능	양방향 추천의 범용화
계산은 필요한 만큼만	추천 후보는 미리, 요청 유저는 호출할 때만 임베딩을 계산	임베딩 생성 시점 분리

Lessons Learned

소셜 추천 개발 과정에서 얻은 것들

데이터 성격에 맞는 경로	천천히 변하는 데이터는 배치로, 접속 상태는 실시간으로 반영	소셜 추천과 데이터
방향이 바뀌어도 같은 구조	two-tower 모델 기반으로, 유연한 추천 확장이 가능	양방향 추천의 범용화
계산은 필요한 만큼만	추천 후보는 미리, 요청 유저는 호출할 때만 임베딩을 계산	임베딩 생성 시점 분리
넓게 찾고, 정교하게 고르기	벡터 검색으로 빠르게 추리고, 필터링 · 재정렬로 순서를 결정	필터링과 재정렬

Lessons Learned

소셜 추천 개발 과정에서 얻은 것들

데이터 성격에 맞는 경로	천천히 변하는 데이터는 배치로, 접속 상태는 실시간으로 반영	소셜 추천과 데이터
방향이 바뀌어도 같은 구조	two-tower 모델 기반으로, 유연한 추천 확장이 가능	양방향 추천의 범용화
계산은 필요한 만큼만	추천 후보는 미리, 요청 유저는 호출할 때만 임베딩을 계산	임베딩 생성 시점 분리
넓게 찾고, 정교하게 고르기	벡터 검색으로 빠르게 추리고, 필터링 · 재정렬로 순서를 결정	필터링과 재정렬
관계 데이터를 활용하기	유형 · 무형의 관계를 그래프로 저장하고, 탐색 결과를 추천 이유로 활용	관계 그래프와 추천 태그

aws FOR GAMES

Thank you

GS Neotek | MEGAZONE CLOUD