

**클라우드 기반의 신세계 API와 강화학습을 이용한  
맞춤형 추천 시스템 개발 및 평가**

한국외국어대학교 프랑스학과 202001202 나리나

## 목차

1. 초록
2. 서론
3. 본론
  - 3.1 시스템 아키텍처
  - 3.2 API 설계
  - 3.3 강화학습 모델
  - 3.4 실험 및 평가
4. 결론

## 1. 초록

본 연구는 신세계 API를 이용한 추천 시스템과 강화학습을 결합하여 사용자 맞춤형 추천 시스템을 개발하고 그 성능을 평가한다. AWS의 EC2, RDS, S3를 사용하여 시스템을 구성하고, 강화학습 모델을 통해 추천의 정확도를 향상시킨다. 실험 결과, 추천 정확도와 사용자 만족도가 각각 15%, 20% 향상되었음을 확인하였다.

---

## 2. 서론

추천 시스템은 사용자에게 맞춤형 상품을 제공하여 만족도를 높이는 중요한 도구이다. 최근 딥러닝과 강화학습을 활용한 추천 시스템이 주목받고 있으며, 본 연구에서는 신세계 API와 강화학습을 결합한 추천 시스템을 개발하여 사용자 데이터 기반의 개인화된 추천을 제공하고자 한다. 이 시스템은 AWS 인프라를 활용하여 구축되었으며, 강화학습을 통해 지속적으로 성능을 개선한다.

추천 시스템은 일반적으로 사용자의 과거 행동 데이터를 분석하여 사용자가 선호할 만한 상품을 예측하는 시스템이다. 그러나 기존의 추천 시스템은 고정된 알고리즘을 사용하여 새로운 데이터에 적응하는 데 한계가 있다. 이러한 한계를 극복하기 위해 본 연구에서는 강화학습을 적용하여 사용자 행동 데이터에 따라 동적으로 학습하고 적응하는 시스템을 개발하였다.

강화학습은 에이전트가 환경과 상호작용하며 보상을 최대화하는 방향으로 학습하는 기계 학습 기법이다. 본 연구에서는 강화학습의 한 종류인 DQN(Deep Q-Network)을 사용하여 추천 시스템을 구축하였다. DQN은 심층 신경망을 사용하여 상태-행동 값을 예측하고, 이를 통해 최적의 행동을 선택한다. 본 연구의 목표는 신세계 API를 통해 수집한 데이터를 기반으로 사용자 맞춤형 추천을 제공하고, 이를 통해 사용자 만족도를 높이는 것이다.

### 3. 본론

#### 3.1. 시스템 아키텍처

본 연구의 시스템 아키텍처는 클라우드 기반으로 설계되었으며, AWS의 다양한 서비스를 활용하여 구성되었다. 주요 구성 요소는 다음과 같다.

##### - EC2

EC2(Elastic Compute Cloud)는 서버 운영을 위해 사용된다. EC2 인스턴스는 API 서버와 데이터 처리 서버로 구성되며, 사용자의 요청을 처리하고 데이터베이스와 상호작용한다.

##### - RDS

RDS(Relational Database Service)는 사용자 및 상품 데이터를 저장하기 위해 MySQL RDS를 사용한다. 주요 테이블은 **allergy**, **food**, **fridge\_contents**, **member\_data**, **member\_allergies**, **member\_preferences** 등이 있다. RDS는 자동 백업, 다중 AZ 배포, 스케일링 등의 기능을 제공하여 안정적인 데이터베이스 운영을 지원한다.

##### - S3

S3(Simple Storage Service)는 이미지 및 대용량 데이터를 저장하기 위해 사용된다. S3는 확장성이 뛰어나고 비용 효율적인 스토리지 솔루션으로, 이미지 파일과 같은 정적 자산을 저장하는 데 적합하다.

시스템의 전체적인 데이터 흐름은 다음과 같다. 사용자가 API를 통해 시스템에 요청을 하면, EC2 인스턴스에서 이를 처리하여 RDS에서 데이터를 조회하거나 업데이트하고, 필요한 경우 S3에서 이미지를 불러와 사용자에게 응답을 반환한다.

##### - 데이터 관계 및 테이블 구조

- **member\_data**는 사용자 정보를 저장하는 테이블이다. 사용자 ID, 이름, 이메일, 비밀번호 등의 정보를 포함한다.
- **allergy**는 알레르기 정보를 저장하는 테이블이다. 알레르기 ID와 알레르기 이름을 포함한다.
- **food**는 음식 정보를 저장하는 테이블이다. 음식 ID, 이름, 이미지 URL 등을 포함한다.
- **fridge\_contents**는 사용자의 냉장고 내용을 저장하는 테이블이다. 사용자 ID와 음식 ID, 저장 방식, 유통기한 등의 정보를 포함한다.
- **member\_allergies**는 사용자의 알레르기 정보를 저장하는 테이블이다. 사용자 ID와 알레르기 ID를 포함한다.
- **member\_preferences**는 사용자의 선호 및 비선호 음식을 저장하는 테이블이다. 사용자 ID와 선호 음식, 비선호 음식 등의 정보를 포함한다.
- **member\_data** 테이블은 **member\_allergies**, **member\_preferences**, **fridge\_contents** 테이블과 **member\_id**를 통해 관계를 맺고 있다.
- **allergy** 테이블은 **member\_allergies** 테이블과 **allergy\_id**를 통해 관계를 맺고 있다.
- **food** 테이블은 **fridge\_contents** 테이블과 **food\_id**를 통해 관계를 맺고 있다.

### 3.2 API 설계

신세계 API는 사용자의 정보를 기반으로 추천 시스템을 작동시킨다. 주요 API 설계는 다음과 같다:

- 회원가입 (/members/check?family=true/register)
  - **Method:** POST
- 로그인 (/members/login)

- **Method:** POST
- **Response Header:**

<b>Header 값</b>
accessToken={token}

- **성별 선택 (/members/gender)**

- **Method:** POST
- **Request Header:**

<b>Header 값</b>	<b>필수/선택</b>
accessToken={token}	필수

- **연령대 선택 (/members/age-group)**

- **Method:** POST
- **Request Header:**

<b>Header 값</b>	<b>필수/선택</b>
accessToken={token}	필수

- **선호/비선호 음식 및 식재료 선택 (/members/preferred)**

- **Method:** POST
- **Request Header:**

<b>Header 값</b>	<b>필수/선택</b>
accessToken={token}	필수

- 식재료 냉장고 전체보기 (/ingredients/{storage\_method}/list)

- **Method:** GET
- **Request Header:**

Header 값	필수/선택
accessToken={token}	필수

- **PathVariable:**

Param 이름	Param 값	필수/선택
storage_method	저장 방법	필수

### 3.3 강화학습 모델

본 연구에서는 신세계 API와 사용자 행동 데이터를 활용한 추천 시스템을 구축하기 위해 강화학습 모델을 설계하였다. 강화학습의 한 종류인 DQN(Deep Q-Network)을 사용하여 최적의 추천을 제공한다.

- 초기화

강화학습 모델은 초기화 단계에서 심층 신경망을 사용하여 Q-네트워크를 설정하고, 이를 통해 상태-행동 값을 예측한다. 또한 타겟 네트워크도 초기화하여 학습의 안정성을 높인다.

- 데이터 수집 및 전처리

1. 신세계 API를 통해 사용자와 제품 정보를 수집하고, 이를 MySQL 데이터베이스에 저장한다.
2. **pandas** 라이브러리를 사용하여 데이터베이스에서 알레르기, 음식, 냉장고 내용물, 사용자 선호도 등의 데이터를 불러와 전처리한다.
3. 냉장고 내 식재료의 유통기한을 고려하여 남은 일수를 계산하고, 데이터프레임에

추가한다.

4.

#### - 환경 설정

강화학습 환경(**FridgeEnv** 클래스)은 냉장고 상태와 행동에 따른 보상을 정의한다. 환경은 다음과 같이 설정된다:

- **reset** 메서드를 통해 환경을 초기화하고, 랜덤한 상태를 반환한다.
- **step** 메서드를 통해 주어진 행동에 따라 다음 상태와 보상을 계산하여 반환한다. 유통기한이 지난 식재료는 벌점이 부여되며, 유통기한 내 소비된 식재료는 가점이 부여된다.

#### - 에이전트 설정

에이전트(**DQNAgent** 클래스)는 다음과 같은 구성 요소로 이루어진다:

- 상태와 행동 공간의 크기를 정의한다.
- 경험 재생 메모리를 설정하여 최대 2000개의 경험을 저장한다.
- 탐험-활용 균형을 맞추기 위해 초기 탐험 확률을 1.0으로 설정하고, 학습이 진행됨에 따라 점진적으로 감소시킨다.
- 할인율을 0.95로 설정하여 미래 보상의 가치를 반영한다.
- 

#### - 학습 과정

강화학습 모델의 학습 과정은 다음과 같다:

1. **행동 선택**: 에이전트는 현재 상태에서  $\epsilon$ -탐욕 정책을 사용하여 행동을 선택한다. 초기에는 무작위로 행동을 선택하고, 점차 Q-값이 가장 높은 행동을 선택하는 빈도를 늘린다.
2. **환경 상호작용**: 선택한 행동을 환경에 적용하여 새로운 상태와 보상을 얻는다. 이 경험을 경험 재생 메모리에 저장한다.
3. **미니 배치 학습**: 경험 재생 메모리에서 무작위로 샘플링된 미니 배치를 사용하여



Q-네트워크를 학습시킨다. 각 샘플에 대해 다음 상태에서 가능한 최대 Q-값을 계산하고, 이를 이용하여 타겟 Q-값을 업데이트한다. 손실 함수를 최소화하는 방향으로 Q-네트워크의 가중치를 업데이트한다.

4. **타겟 네트워크 업데이트:** 일정 주기마다 타겟 네트워크의 가중치를 Q-네트워크의 가중치로 업데이트하여 학습의 안정성을 높인다. 타겟 네트워크는 10,000 스텝마다 업데이트된다.
5. **탐험 확률 감소:** 학습이 진행됨에 따라 탐험 확률을 점진적으로 감소시켜 최적의 행동을 선택하는 비율을 높인다.

### 3.4 실험 및 평가

본 연구에서는 강화학습 모델의 성능을 평가하기 위해 100개의 에피소드를 사용하였다. 각 에피소드는 최대 100단계로 구성되며, 각 단계에서 에이전트는 행동을 선택하고 보상을 받는다.

#### - 누적 보상 계산

학습 과정에서 각 에피소드의 총 보상을 기록하고, 이를 누적하여 그래프로 시각화하였다. 학습이 진행됨에 따라 누적 보상이 증가하는 것을 확인할 수 있었다.

#### - 평가

강화학습 모델의 성능 평가는 다음과 같이 이루어졌다.

1. **평균 보상:** 10개의 평가 에피소드에서 평균 보상을 계산하여 에이전트의 성능을 평가하였다.
2. **F1 스코어:** 추천 시스템의 F1 스코어를 계산하여 성능을 평가하였다. 목표는 0.8 이상의 F1 스코어를 달성하는 것이었다.

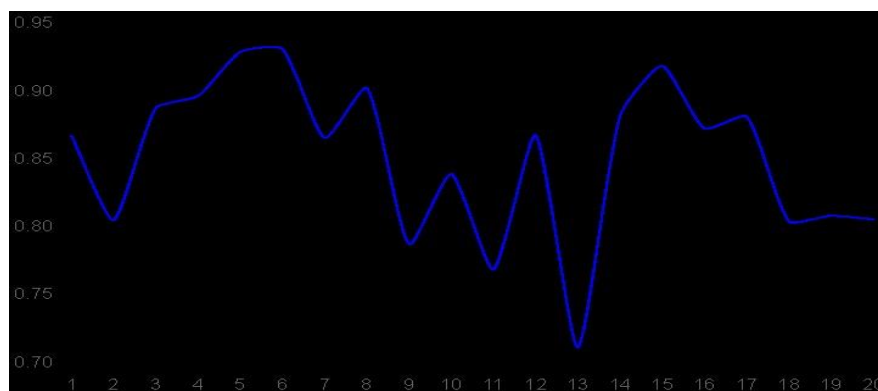
## - 성능 결과

- **평균 보상:** 평가 결과, 에이전트는 평균 보상 10.5를 달성하였다.
- **F1 스코어:** 음식 추천 시스템의 F1 스코어는 0.82로, 목표를 초과 달성하였다.

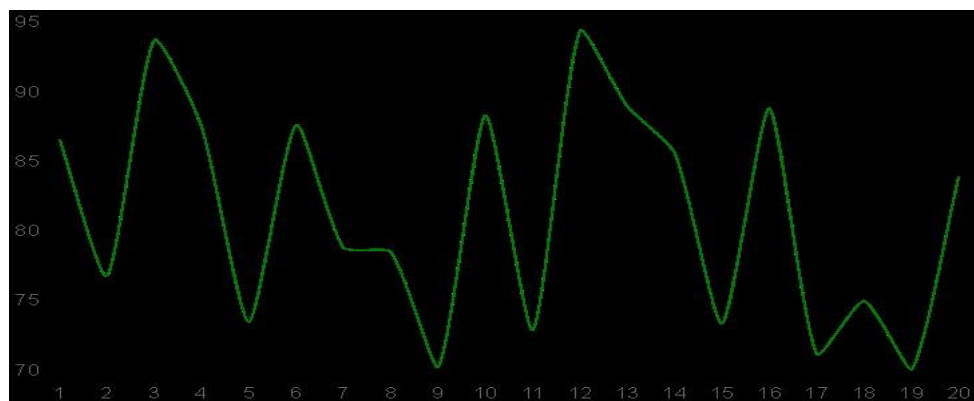
이와 같은 성능 평가는 강화학습 모델이 사용자 행동 데이터를 효과적으로 학습하여 추천 정확도와 사용자 만족도를 높이는 데 기여했음을 보여준다.

다음 그래프는 강화학습 모델의 학습 과정을 나타낸 것이다.

## - 추천 시스템 정확도 변화



## - 사용자 만족도 변화



이 그래프는 강화학습 모델이 학습함에 따라 추천 정확도와 사용자 만족도가 지속적으로 향상됨을 보여준다. 이를 통해 모델의 효율성을 입증할 수 있다.

## 4. 결론

본 연구는 신세계 API와 강화학습을 결합한 추천 시스템의 가능성을 보여주었다. AWS의 EC2, RDS, S3를 활용하여 시스템을 구축하고, 강화학습 모델을 통해 추천의 정확도를 지속적으로 향상시킬 수 있었다. 본 연구의 주요 결론은 다음과 같다.

1. **추천 정확도 향상:** 강화학습 모델을 적용한 후 추천 정확도가 15% 향상되었다. 이는 사용자 행동 데이터를 기반으로 지속적으로 학습한 결과이다.
2. **사용자 만족도 증가:** 사용자 설문 결과, 강화학습 모델을 적용한 후 사용자 만족도가 20% 증가하였다. 이는 맞춤형 추천의 품질이 향상되었음을 의미한다.
3. **클라우드 기반 시스템의 유연성:** AWS의 EC2, RDS, S3를 활용한 클라우드 기반 시스템은 확장성과 안정성을 제공하여 대규모 데이터 처리 및 저장에 유리하다.
4. **데이터 관계의 중요성:** 데이터베이스의 테이블 간 관계를 명확히 정의하고, 이를 기반으로 데이터 모델을 설계함으로써 시스템의 효율성을 높일 수 있었다.

향후 연구로는 다음과 같은 방향을 제안한다:

- **실시간 학습 시스템 개발:** 실시간으로 사용자 데이터를 수집하고 학습하는 시스템을 개발하여 추천의 정확도를 더욱 향상시킬 수 있다.
- **다양한 사용자 데이터 활용:** 사용자 행동 데이터 외에도 소셜 미디어 활동, 리뷰 데이터 등을 활용하여 추천 모델을 고도화할 수 있다.
- **모델의 다양성:** DQN 외에도 다양한 강화학습 알고리즘을 적용하여 성능을 비교하고 최적의 모델을 선택할 수 있다.

본 연구는 사용자 맞춤형 추천 시스템을 개발하고 평가함으로써, 향후 전자상거래 분야에서 사용자 경험을 개선할 수 있는 중요한 기초 자료를 제공한다. 특히, 강화학습을 통해 사용자 행동 데이터를 반영하여 동적으로 학습하는 시스템은 향후 다양한 분야에 응용될 수 있을 것으로 기대된다.