

Implementación de un Sistema de Recomendación

Usando IA para Recomendaciones Personalizadas

Introducción

En este documento, explicaremos paso a paso cómo implementar un sistema de recomendación utilizando técnicas de inteligencia artificial. Utilizaremos Python y la biblioteca 'surprise' para construir un modelo de recomendación basado en el algoritmo de matriz de factorización SVD (Singular Value Decomposition).

Datos y Preparación

Primero, definimos un conjunto de datos de ejemplo que contiene las calificaciones que los usuarios han dado a varias películas. Convertimos estos datos en un DataFrame de pandas para facilitar su manipulación.

Código:

```
```python
```

```
import pandas as pd
```

```
data = {
```

```
 'user_id': ['user1', 'user1', 'user1', 'user2', 'user2', 'user2', 'user3', 'user3', 'user3', 'user4', 'user4',
 'user4', 'user5', 'user5', 'user5'],
```

```
 'item_id': ['movie1', 'movie2', 'movie3', 'movie1', 'movie2', 'movie4', 'movie2', 'movie3', 'movie5',
 'movie1', 'movie3', 'movie6', 'movie4', 'movie5', 'movie6'],
```

```
 'rating': [5, 3, 4, 4, 2, 5, 5, 2, 3, 2, 3, 4, 3, 5, 4]
```

```
}
```

```
df = pd.DataFrame(data)
```

## Implementación de un Sistema de Recomendación

### Usando IA para Recomendaciones Personalizadas

```
print('Datos de calificaciones de usuarios y películas:')

print(df)

...
```

Explicación:

- ``import pandas as pd``: Importa la biblioteca pandas para manipular los datos.
- ``data``: Define un diccionario con los datos de ejemplo.
- ``pd.DataFrame(data)``: Convierte el diccionario en un DataFrame de pandas.
- ``print(df)``: Muestra los datos en formato de tabla.

### Biblioteca ``surprise``

Usamos la biblioteca ``surprise`` para construir el modelo de recomendación. Primero, cargamos los datos en un formato que ``surprise`` pueda entender.

Código:

```
```python  
  
from surprise import Dataset, Reader  
  
reader = Reader(rating_scale=(1, 5))  
  
data = Dataset.load_from_df(df[['user_id', 'item_id', 'rating']], reader)  
  
...
```

Explicación:

- ``from surprise import Dataset, Reader``: Importa las clases necesarias de la biblioteca ``surprise``.

Implementación de un Sistema de Recomendación

Usando IA para Recomendaciones Personalizadas

- ``Reader(rating_scale=(1, 5))``: Define la escala de calificaciones.
- ``Dataset.load_from_df(...)``: Carga los datos del DataFrame en el formato requerido por ``surprise``.

Entrenamiento del Modelo

Entrenamos un modelo de recomendación utilizando el algoritmo SVD. Primero, construimos el conjunto de entrenamiento completo y luego entrenamos el modelo. También realizamos una validación cruzada para evaluar el rendimiento del modelo.

Código:

```
```python
from surprise import SVD

from surprise.model_selection import cross_validate

trainset = data.build_full_trainset()

algo = SVD()

cross_validate(algo, data, measures=['RMSE', 'MAE'], cv=5, verbose=True)

algo.fit(trainset)
```
```

Explicación:

- ``from surprise import SVD``: Importa el algoritmo SVD.
- ``from surprise.model_selection import cross_validate``: Importa la función de validación cruzada.
- ``data.build_full_trainset()``: Construye el conjunto de entrenamiento completo.
- ``algo = SVD()``: Crea una instancia del algoritmo SVD.

Implementación de un Sistema de Recomendación

Usando IA para Recomendaciones Personalizadas

- `cross_validate(...)`: Realiza la validación cruzada para evaluar el modelo.
- `algo.fit(trainset)`: Entrena el modelo con el conjunto de entrenamiento.

Predicciones

Una vez entrenado el modelo, podemos hacer predicciones de calificaciones para usuarios y películas específicos. Aquí, predecimos la calificación que el usuario 'user1' daría a la película 'movie4'.

Código:

```
```python
user_id = 'user1'
item_id = 'movie4'

pred = algo.predict(user_id, item_id)

print(f'Predicción de calificación para {user_id} y {item_id}: {pred.est:.2f}')
```
```

Explicación:

- `algo.predict(user_id, item_id)`: Predice la calificación que el usuario daría a la película.
- `pred.est`: Obtiene la calificación predicha.

Generación de Recomendaciones

Podemos generar recomendaciones para un usuario específico filtrando las películas que el usuario

Implementación de un Sistema de Recomendación

Usando IA para Recomendaciones Personalizadas

no ha calificado y prediciendo las calificaciones para esas películas. Luego, ordenamos las predicciones y seleccionamos las mejores recomendaciones.

Código:

```
```python
```

```
def get_recommendations(user_id, df, algo, n=5):
 all_items = df['item_id'].unique()
 user_items = df[df['user_id'] == user_id]['item_id'].unique()
 items_to_predict = [item for item in all_items if item not in user_items]
 predictions = [algo.predict(user_id, item).est for item in items_to_predict]
 recommendations = pd.DataFrame({'item_id': items_to_predict, 'predicted_rating': predictions})
 recommendations = recommendations.sort_values(by='predicted_rating', ascending=False)
 return recommendations.head(n)

recommendations_for_user1 = get_recommendations('user1', df, algo, n=5)
print(f'Recomendaciones para user1:')
print(recommendations_for_user1)
...
```

Explicación:

- `get\_recommendations(...)` : Función para generar recomendaciones.
- `all\_items` : Lista de todas las películas únicas.
- `user\_items` : Lista de películas que el usuario ya ha calificado.

## **Implementación de un Sistema de Recomendación**

### **Usando IA para Recomendaciones Personalizadas**

- ``items_to_predict``: Películas que el usuario no ha calificado.
- ``algo.predict(...)``: Predice las calificaciones para las películas no calificadas.
- ``recommendations.sort_values(...)``: Ordena las predicciones de mayor a menor.
- ``recommendations.head(n)``: Retorna las `n` mejores recomendaciones.

### **Conclusión**

Este código implementa un sistema de recomendación utilizando técnicas de inteligencia artificial, específicamente el algoritmo de matriz de factorización SVD. El proceso incluye la preparación de los datos, la carga en el formato adecuado, el entrenamiento del modelo, la validación cruzada para evaluar el rendimiento y la generación de recomendaciones personalizadas.

Este enfoque es similar a lo que utilizan plataformas como Netflix para proporcionar recomendaciones personalizadas a sus usuarios, mejorando significativamente la experiencia del usuario.