

```

In [8]: import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.metrics import roc_auc_score
import random

# Загрузка данных
ratings = pd.read_csv('ratings_df_sample_2.csv')

# Добавление негативных примеров (для целей демонстрации)
negatives = ratings[['userId', 'movieId']].drop_duplicates().sample(frac=
negatives['rating'] = 0 # Предполагаем, что рейтинг 0 означает не просмо
negatives['watched'] = 0

# Добавляем негативные примеры в исходные данные
ratings['watched'] = 1
ratings = pd.concat([ratings, negatives])

# Подготовка тренировочных и тестовых данных
train, test = train_test_split(ratings, test_size=0.2, random_state=42, s

# Обучение dummy model
class DummyModel:
    def fit(self, X, y):
        pass

    def predict_proba(self, X):
        return np.array([[random.random(), random.random()] for _ in rang

dummy_model = DummyModel()
dummy_model.fit(train[['userId', 'movieId']], train['watched'])
dummy_pred = dummy_model.predict_proba(test[['userId', 'movieId']])

# Оценка качества dummy model
roc_auc = roc_auc_score(test['watched'], [p[1] for p in dummy_pred])
print('ROC AUC for Dummy Model:', roc_auc)

```

ROC AUC for Dummy Model: 0.5006155219533498

In []: