Final_tt_model

May 22, 2025

1 RMSE

1.1 Tuning for RMSE & Plot test RMSE

[ ]:
```
import tensorflow as tf
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from itertools import product
from sklearn.metrics import mean_squared_error
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
import time

# ===============================
# 1. Set Random Seed
# ===============================
tf.random.set_seed(42)
np.random.seed(42)

# ===============================
# 2. Load Data
# ===============================
ratings = pd.read_csv('u.data', sep='\t', names=['user_id', 'item_id',␣
↪'rating', 'timestamp'])

movies = pd.read_csv(
    'u.item',
    sep='|',
    encoding='latin-1',
    names=['item_id', 'movie_title', 'release_date', 'video_release_date',␣
↪'IMDb_URL'] + [f'genre_{i}' for i in range(19)],
    usecols=range(24)
)

users = pd.read_csv(
    'u.user',
    sep='|',
```

```
    names=['user_id', 'age', 'gender', 'occupation', 'zip_code']
)

# ================================
# 3. Preprocessing
# ================================
ratings = ratings.sort_values('timestamp').drop_duplicates(subset=['user_id',␣
↪'item_id'], keep='last')
ratings = ratings.merge(movies, on='item_id')
ratings = ratings.merge(users[['user_id', 'age', 'gender', 'occupation']],␣
↪on='user_id')

# --- Mean Center and Normalize Ratings per User ---
user_mean = ratings.groupby('user_id')['rating'].mean()
ratings['user_mean'] = ratings['user_id'].map(user_mean)
ratings['mean_centered_rating'] = ratings['rating'] - ratings['user_mean']

# Normalize mean-centered ratings to [-1, 1] using max absolute deviation per␣
↪user
user_max_dev = ratings.groupby('user_id')['mean_centered_rating'].apply(lambda␣
↪x: x.abs().max())
ratings['user_max_dev'] = ratings['user_id'].map(user_max_dev)

# Avoid division by zero for constant ratings
ratings['user_max_dev'] = ratings['user_max_dev'].replace(0, 1)

ratings['normalized_rating'] = ratings['mean_centered_rating'] /␣
↪ratings['user_max_dev']

# --- Encode Features ---
genre_cols = [f'genre_{i}' for i in range(19)]
user_encoder = LabelEncoder()
movie_encoder = LabelEncoder()
ratings['user'] = user_encoder.fit_transform(ratings['user_id'])
ratings['movie'] = movie_encoder.fit_transform(ratings['item_id'])
ratings['gender'] = ratings['gender'].map({'M': 0, 'F': 1})
ratings['occupation'] = ratings['occupation'].astype('category').cat.codes

n_users = ratings['user'].nunique()
n_movies = ratings['movie'].nunique()
n_occupations = len(ratings['occupation'].unique())


# ================================
# 4. Train-Test Split by User
# ================================
train_rows, test_rows = [], []
```

```python
for _, user_ratings in ratings.groupby('user_id'):
    if len(user_ratings) < 5:
        train_rows.append(user_ratings)
    else:
        train, test = train_test_split(user_ratings, test_size=0.2, random_state=42)
        train_rows.append(train)
        test_rows.append(test)

train_df = pd.concat(train_rows).reset_index(drop=True)
test_df = pd.concat(test_rows).reset_index(drop=True)
# ================================
# 5. Dataset Builder
# ================================
def build_dataset(df):
    return tf.data.Dataset.from_tensor_slices((
        {
            "user_id": df['user'].values,
            "movie_id": df['movie'].values,
            "age": df['age'].values.astype(np.float32).reshape(-1, 1),
            "gender": df['gender'].values,
            "occupation": df['occupation'].values,
            "genres": df[genre_cols].values.astype(np.float32)
        },
        df['normalized_rating'].values.astype(np.float32)
    )).batch(1024).prefetch(tf.data.AUTOTUNE)

train_dataset = build_dataset(train_df)
test_dataset = build_dataset(test_df)

# ================================
# 6. Model: genre_dense_size tied to embedding_size
# ================================
class RecommenderNet(tf.keras.Model):
    def __init__(self, n_users, n_movies, embedding_size, user_dense_size, movie_dense_size, final_dim):
        super(RecommenderNet, self).__init__()

        # User tower
        self.user_embedding = tf.keras.layers.Embedding(n_users, embedding_size)
        self.gender_embedding = tf.keras.layers.Embedding(2, embedding_size // 2)
        self.occupation_embedding = tf.keras.layers.Embedding(n_occupations, embedding_size // 2)
        self.age_normalization = tf.keras.layers.Normalization()
        self.age_dense = tf.keras.layers.Dense(embedding_size // 2, activation="relu")
```

```python
        self.user_dense = tf.keras.layers.Dense(user_dense_size,␣
↪activation="relu")
        self.user_final = tf.keras.layers.Dense(final_dim)

        # Movie tower
        self.movie_embedding = tf.keras.layers.Embedding(n_movies,␣
↪embedding_size)
        self.genre_dense = tf.keras.layers.Dense(embedding_size,␣
↪activation="relu")  # Tied to embedding_size
        self.movie_dense = tf.keras.layers.Dense(movie_dense_size,␣
↪activation="relu")
        self.movie_final = tf.keras.layers.Dense(final_dim)

    def call(self, inputs):
        user_vec = self.user_embedding(inputs["user_id"])
        gender_vec = self.gender_embedding(inputs["gender"])
        occupation_vec = self.occupation_embedding(inputs["occupation"])
        age_vec = self.age_dense(self.age_normalization(inputs["age"]))

        user_features = tf.concat([user_vec, gender_vec, occupation_vec,␣
↪age_vec], axis=1)
        user_features = self.user_dense(user_features)
        user_features = self.user_final(user_features)

        movie_vec = self.movie_embedding(inputs["movie_id"])
        genre_vec = self.genre_dense(inputs["genres"])
        movie_features = tf.concat([movie_vec, genre_vec], axis=1)
        movie_features = self.movie_dense(movie_features)
        movie_features = self.movie_final(movie_features)

        user_norm = tf.math.l2_normalize(user_features, axis=1)
        movie_norm = tf.math.l2_normalize(movie_features, axis=1)
        return tf.reduce_sum(user_norm * movie_norm, axis=1)

# ================================
# 7. Train/Evaluate
# ================================
def train_and_evaluate_model(embedding_size, user_dense_size, movie_dense_size,␣
↪final_dim):
    tf.keras.backend.clear_session()
    model = RecommenderNet(n_users, n_movies, embedding_size, user_dense_size,␣
↪movie_dense_size, final_dim)
    model.age_normalization.adapt(train_df['age'].values.astype(np.float32).
↪reshape(-1, 1))
```

```python
    model.compile(optimizer=tf.keras.optimizers.Adam(0.001), loss=tf.keras.losses.MeanSquaredError())
    history = model.fit(train_dataset, validation_data=test_dataset, epochs=10, verbose=0)

    train_preds = model.predict(train_dataset)
    test_preds = model.predict(test_dataset)

    return {
        'train_rmse': np.sqrt(mean_squared_error(train_df['normalized_rating'], train_preds)),
        'test_rmse': np.sqrt(mean_squared_error(test_df['normalized_rating'], test_preds)),
        'val_loss_curve': history.history['val_loss'],
    }

# ================================
# 8. Experiment Grid (no genre_dense_size)
# ================================
embedding_sizes = [16, 32, 64]
user_dense_sizes = [8, 16, 32, 64, 128]
movie_dense_sizes = [8, 16, 32, 64, 128]
final_dims = [64, 128]

experiments = [
    {
        'embedding_size': emb,
        'user_dense_size': uds,
        'movie_dense_size': mds,
        'final_dim': fd
    }
    for emb, uds, mds, fd in product(
        embedding_sizes, user_dense_sizes, movie_dense_sizes, final_dims
    )
]

# ================================
# 9. Run Experiments
# ================================
results = {}
start_time = time.time()
for config in experiments:
    key = f"emb{config['embedding_size']}_user{config['user_dense_size']}_movie{config['movie_dense_s
    print(f"Running: {key}")
    results[key] = train_and_evaluate_model(**config)
end_time = time.time()
```

```python
print(f"Total time taken: {(end_time - start_time):.2f} seconds.")

# ===============================
# 10. Plot Test RMSE
# ===============================
def plot_test_rmse(results):
    keys = list(results.keys())
    rmse_values = [results[k]['test_rmse'] for k in keys]

    plt.figure(figsize=(12, 5))
    plt.plot(rmse_values, marker='o', label='Test RMSE')
    plt.xticks(ticks=range(len(keys)), labels=keys, rotation=90)
    plt.ylabel("RMSE")
    plt.title("Test RMSE across parameter settings")
    plt.grid(True)
    plt.tight_layout()
    plt.show()

plot_test_rmse(results)

# ===============================
# 11. Top/Bottom 5
# ===============================
def print_top_bottom(results):
    sorted_by_rmse = sorted(results.items(), key=lambda x: x[1]['test_rmse'])

    print("\nTop 5 by RMSE:")
    for k, v in sorted_by_rmse[:5]:
        print(f"{k}: Test RMSE={v['test_rmse']:.4f}, Train␣
↪RMSE={v['train_rmse']:.4f}")

    print("\nBottom 5 by RMSE:")
    for k, v in sorted_by_rmse[-5:]:
        print(f"{k}: Test RMSE={v['test_rmse']:.4f}, Train␣
↪RMSE={v['train_rmse']:.4f}")

print_top_bottom(results)
```

```
Running: emb16_user8_movie8_final64
78/78                 0s 2ms/step
20/20                 0s 2ms/step
Running: emb16_user8_movie8_final128
78/78                 0s 2ms/step
20/20                 0s 2ms/step
Running: emb16_user8_movie16_final64
78/78                 0s 2ms/step
20/20                 0s 2ms/step
```

```
Running: emb16_user8_movie16_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user8_movie32_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user8_movie32_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user8_movie64_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user8_movie64_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user8_movie128_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user8_movie128_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user16_movie8_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user16_movie8_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user16_movie16_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user16_movie16_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user16_movie32_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user16_movie32_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user16_movie64_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user16_movie64_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user16_movie128_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
```

```
Running: emb16_user16_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user32_movie8_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user32_movie8_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user32_movie16_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user32_movie16_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user32_movie32_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user32_movie32_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user32_movie64_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user32_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user32_movie128_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user32_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user64_movie8_final64
78/78              0s 3ms/step
20/20              0s 2ms/step
Running: emb16_user64_movie8_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user64_movie16_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user64_movie16_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user64_movie32_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
```

```
Running: emb16_user64_movie32_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user64_movie64_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user64_movie64_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user64_movie128_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user64_movie128_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user128_movie8_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user128_movie8_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user128_movie16_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user128_movie16_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user128_movie32_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user128_movie32_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user128_movie64_final64
78/78            0s 4ms/step
20/20            0s 2ms/step
Running: emb16_user128_movie64_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb16_user128_movie128_final64
78/78            0s 2ms/step
20/20            0s 6ms/step
Running: emb16_user128_movie128_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb32_user8_movie8_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
```

```
Running: emb32_user8_movie8_final128
78/78             0s 2ms/step
20/20             0s 6ms/step
Running: emb32_user8_movie16_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb32_user8_movie16_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb32_user8_movie32_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb32_user8_movie32_final128
78/78             0s 3ms/step
20/20             0s 2ms/step
Running: emb32_user8_movie64_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb32_user8_movie64_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb32_user8_movie128_final64
78/78             0s 2ms/step
20/20             0s 5ms/step
Running: emb32_user8_movie128_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb32_user16_movie8_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb32_user16_movie8_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb32_user16_movie16_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb32_user16_movie16_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb32_user16_movie32_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb32_user16_movie32_final128
78/78             0s 3ms/step
20/20             0s 2ms/step
Running: emb32_user16_movie64_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
```

```
Running: emb32_user16_movie64_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb32_user16_movie128_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb32_user16_movie128_final128
78/78            0s 3ms/step
20/20            0s 2ms/step
Running: emb32_user32_movie8_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb32_user32_movie8_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb32_user32_movie16_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb32_user32_movie16_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb32_user32_movie32_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb32_user32_movie32_final128
78/78            0s 3ms/step
20/20            0s 2ms/step
Running: emb32_user32_movie64_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb32_user32_movie64_final128
78/78            0s 3ms/step
20/20            0s 2ms/step
Running: emb32_user32_movie128_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb32_user32_movie128_final128
78/78            0s 2ms/step
20/20            0s 6ms/step
Running: emb32_user64_movie8_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb32_user64_movie8_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb32_user64_movie16_final64
78/78            0s 3ms/step
20/20            0s 2ms/step
```

```
Running: emb32_user64_movie16_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie32_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie32_final128
78/78              0s 3ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie64_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie128_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie128_final128
78/78              0s 3ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie8_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie8_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie16_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie16_final128
78/78              0s 3ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie32_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie32_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie64_final64
78/78              0s 3ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie128_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
```

```
Running: emb32_user128_movie128_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user8_movie8_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user8_movie8_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user8_movie16_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user8_movie16_final128
78/78            0s 2ms/step
20/20            0s 5ms/step
Running: emb64_user8_movie32_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user8_movie32_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user8_movie64_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user8_movie64_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user8_movie128_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user8_movie128_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user16_movie8_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user16_movie8_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user16_movie16_final64
78/78            0s 2ms/step
20/20            0s 6ms/step
Running: emb64_user16_movie16_final128
78/78            0s 3ms/step
20/20            0s 2ms/step
Running: emb64_user16_movie32_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
```

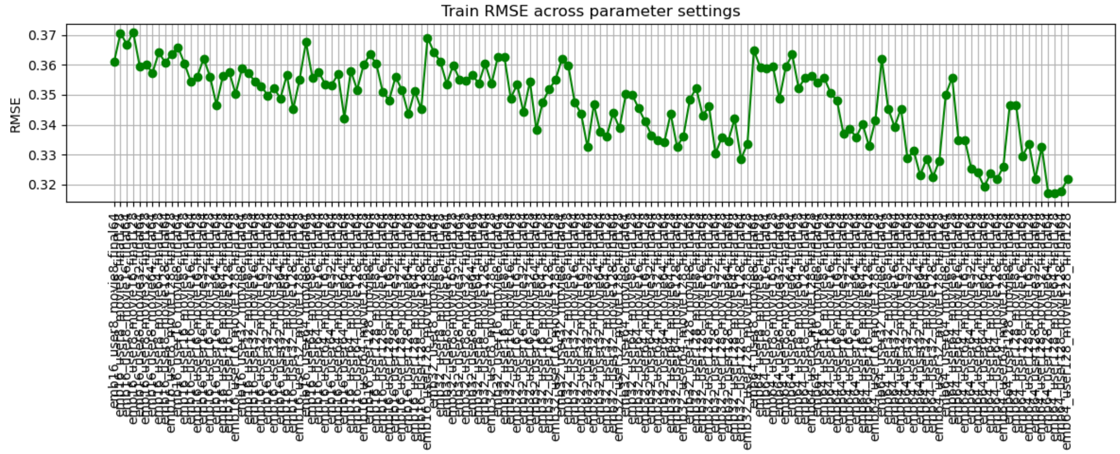
```
Running: emb64_user16_movie32_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user16_movie64_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user16_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user16_movie128_final64
78/78              0s 2ms/step
20/20              0s 5ms/step
Running: emb64_user16_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie8_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie8_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie16_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie16_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie32_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie32_final128
78/78              0s 2ms/step
20/20              0s 5ms/step
Running: emb64_user32_movie64_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie128_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user64_movie8_final64
78/78              0s 2ms/step
20/20              0s 5ms/step
```

```
Running: emb64_user64_movie8_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie16_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie16_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie32_final64
78/78            0s 3ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie32_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie64_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie64_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie128_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie128_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user128_movie8_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user128_movie8_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user128_movie16_final64
78/78            0s 2ms/step
20/20            0s 5ms/step
Running: emb64_user128_movie16_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user128_movie32_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user128_movie32_final128
78/78            0s 2ms/step
20/20            0s 6ms/step
Running: emb64_user128_movie64_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
```

```
Running: emb64_user128_movie64_final128
78/78               0s 2ms/step
20/20               0s 2ms/step
Running: emb64_user128_movie128_final64
78/78               0s 2ms/step
20/20               0s 4ms/step
Running: emb64_user128_movie128_final128
```

1.2 Plot train RMSE [for information]

```
[ ]: # ===============================
# 10. Plot Train RMSE
# ===============================
def plot_train_rmse(results):
    keys = list(results.keys())
    rmse_values = [results[k]['train_rmse'] for k in keys]

    plt.figure(figsize=(12, 5))
    plt.plot(rmse_values, marker='o', color='green', label='Train RMSE')
    plt.xticks(ticks=range(len(keys)), labels=keys, rotation=90)
    plt.ylabel("RMSE")
    plt.title("Train RMSE across parameter settings")
    plt.grid(True)
    plt.tight_layout()
    plt.show()

plot_train_rmse(results)

# ===============================
# 11. Print Top/Bottom 5 by Train RMSE
# ===============================
def print_top_bottom_train(results):
    sorted_by_train_rmse = sorted(results.items(), key=lambda x:␣
 ↪x[1]['train_rmse'])

    print("\nTop 5 by Train RMSE:")
    for k, v in sorted_by_train_rmse[:5]:
        print(f"{k}: Train RMSE={v['train_rmse']:.4f}, Test␣
 ↪RMSE={v['test_rmse']:.4f}")

    print("\nBottom 5 by Train RMSE:")
    for k, v in sorted_by_train_rmse[-5:]:
        print(f"{k}: Train RMSE={v['train_rmse']:.4f}, Test␣
 ↪RMSE={v['test_rmse']:.4f}")

print_top_bottom_train(results)
```




```
Top 5 by Train RMSE:
emb64_user128_movie64_final128: Train RMSE=0.3170, Test RMSE=0.3967
emb64_user128_movie64_final64: Train RMSE=0.3172, Test RMSE=0.3997
emb64_user128_movie128_final64: Train RMSE=0.3177, Test RMSE=0.3994
emb64_user64_movie64_final64: Train RMSE=0.3192, Test RMSE=0.3968
emb64_user128_movie128_final128: Train RMSE=0.3217, Test RMSE=0.3987

Bottom 5 by Train RMSE:
emb16_user8_movie16_final64: Train RMSE=0.3668, Test RMSE=0.3963
emb16_user64_movie8_final64: Train RMSE=0.3676, Test RMSE=0.3961
emb16_user128_movie128_final128: Train RMSE=0.3691, Test RMSE=0.4044
emb16_user8_movie8_final128: Train RMSE=0.3706, Test RMSE=0.3958
emb16_user8_movie16_final128: Train RMSE=0.3709, Test RMSE=0.4009
```

1.3 MAE

1.4 Tuning with MAE

[ ]:
```python
from sklearn.metrics import mean_absolute_error
class RecommenderNet(tf.keras.Model):
    def __init__(self, n_users, n_movies, embedding_size, user_dense_size, movie_dense_size, final_dim):
        super(RecommenderNet, self).__init__()

        # User tower
        self.user_embedding = tf.keras.layers.Embedding(n_users, embedding_size)
        self.gender_embedding = tf.keras.layers.Embedding(2, embedding_size // 2)
        self.occupation_embedding = tf.keras.layers.Embedding(n_occupations, embedding_size // 2)
        self.age_normalization = tf.keras.layers.Normalization()
```

```
        self.age_dense = tf.keras.layers.Dense(embedding_size // 2,␣
↪activation="relu")
        self.user_dense = tf.keras.layers.Dense(user_dense_size,␣
↪activation="relu")
        self.user_final = tf.keras.layers.Dense(final_dim)

        # Movie tower
        self.movie_embedding = tf.keras.layers.Embedding(n_movies,␣
↪embedding_size)
        self.genre_dense = tf.keras.layers.Dense(embedding_size,␣
↪activation="relu")
        self.movie_dense = tf.keras.layers.Dense(movie_dense_size,␣
↪activation="relu")
        self.movie_final = tf.keras.layers.Dense(final_dim)

    def call(self, inputs):
        user_vec = self.user_embedding(inputs["user_id"])
        gender_vec = self.gender_embedding(inputs["gender"])
        occupation_vec = self.occupation_embedding(inputs["occupation"])
        age_vec = self.age_dense(self.age_normalization(inputs["age"]))

        user_features = tf.concat([user_vec, gender_vec, occupation_vec,␣
↪age_vec], axis=1)
        user_features = self.user_dense(user_features)
        user_features = self.user_final(user_features)

        movie_vec = self.movie_embedding(inputs["movie_id"])
        genre_vec = self.genre_dense(inputs["genres"])
        movie_features = tf.concat([movie_vec, genre_vec], axis=1)
        movie_features = self.movie_dense(movie_features)
        movie_features = self.movie_final(movie_features)

        user_norm = tf.math.l2_normalize(user_features, axis=1)
        movie_norm = tf.math.l2_normalize(movie_features, axis=1)
        return tf.reduce_sum(user_norm * movie_norm, axis=1)

# ===============================
# 7. Train/Evaluate using MAE
# ===============================
def train_and_evaluate_model(embedding_size, user_dense_size, movie_dense_size,␣
↪final_dim):
    tf.keras.backend.clear_session()
    model = RecommenderNet(n_users, n_movies, embedding_size, user_dense_size,␣
↪movie_dense_size, final_dim)
    model.age_normalization.adapt(train_df['age'].values.astype(np.float32).
↪reshape(-1, 1))
```

```
    model.compile(optimizer=tf.keras.optimizers.Adam(0.001), loss=tf.keras.
↪losses.MeanAbsoluteError())
    history = model.fit(train_dataset, validation_data=test_dataset, epochs=10,␣
↪verbose=0)

    train_preds = model.predict(train_dataset)
    test_preds = model.predict(test_dataset)

    return {
        'train_mae': mean_absolute_error(train_df['normalized_rating'],␣
↪train_preds),
        'test_mae': mean_absolute_error(test_df['normalized_rating'],␣
↪test_preds),
        'val_loss_curve': history.history['val_loss'],
    }

# ===============================
# 8. Experiment Grid
# ===============================
embedding_sizes = [16, 32, 64]
user_dense_sizes = [8, 16, 32, 64, 128]
movie_dense_sizes = [8, 16, 32, 64, 128]
final_dims = [64, 128]

experiments = [
    {
        'embedding_size': emb,
        'user_dense_size': uds,
        'movie_dense_size': mds,
        'final_dim': fd
    }
    for emb, uds, mds, fd in product(
        embedding_sizes, user_dense_sizes, movie_dense_sizes, final_dims
    )
]

# ===============================
# 9. Run Experiments
# ===============================
results = {}
start_time = time.time()
for config in experiments:
    key =␣
↪f"emb{config['embedding_size']}_user{config['user_dense_size']}_movie{config['movie_dense_s
    print(f"Running: {key}")
    results[key] = train_and_evaluate_model(**config)
```

```python
end_time = time.time()
print(f"Total time taken: {(end_time - start_time):.2f} seconds.")

# ================================
# 10. Plot Test MAE
# ================================
def plot_test_mae(results):
    keys = list(results.keys())
    mae_values = [results[k]['test_mae'] for k in keys]

    plt.figure(figsize=(12, 5))
    plt.plot(mae_values, marker='o', label='Test MAE')
    plt.xticks(ticks=range(len(keys)), labels=keys, rotation=90)
    plt.ylabel("MAE")
    plt.title("Test MAE across parameter settings")
    plt.grid(True)
    plt.tight_layout()
    plt.show()

plot_test_mae(results)

# ================================
# 11. Top/Bottom 5 by MAE
# ================================
def print_top_bottom(results):
    sorted_by_mae = sorted(results.items(), key=lambda x: x[1]['test_mae'])

    print("\nTop 5 by MAE:")
    for k, v in sorted_by_mae[:5]:
        print(f"{k}: Test MAE={v['test_mae']:.4f}, Train MAE={v['train_mae']:.
↪4f}")

    print("\nBottom 5 by MAE:")
    for k, v in sorted_by_mae[-5:]:
        print(f"{k}: Test MAE={v['test_mae']:.4f}, Train MAE={v['train_mae']:.
↪4f}")

print_top_bottom(results)
```

```
Running: emb16_user8_movie8_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user8_movie8_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user8_movie16_final64
78/78              0s 2ms/step
```

```
20/20             0s 2ms/step
Running: emb16_user8_movie16_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user8_movie32_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user8_movie32_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user8_movie64_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user8_movie64_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user8_movie128_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user8_movie128_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user16_movie8_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user16_movie8_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user16_movie16_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user16_movie16_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user16_movie32_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user16_movie32_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user16_movie64_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user16_movie64_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user16_movie128_final64
78/78             0s 2ms/step
```

```
20/20             0s 2ms/step
Running: emb16_user16_movie128_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user32_movie8_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user32_movie8_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user32_movie16_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user32_movie16_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user32_movie32_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user32_movie32_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user32_movie64_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user32_movie64_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user32_movie128_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user32_movie128_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user64_movie8_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user64_movie8_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user64_movie16_final64
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user64_movie16_final128
78/78             0s 2ms/step
20/20             0s 2ms/step
Running: emb16_user64_movie32_final64
78/78             0s 2ms/step
```

```
20/20              0s 2ms/step
Running: emb16_user64_movie32_final128
78/78              0s 3ms/step
20/20              0s 2ms/step
Running: emb16_user64_movie64_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user64_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user64_movie128_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user64_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user128_movie8_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user128_movie8_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user128_movie16_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user128_movie16_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user128_movie32_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user128_movie32_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user128_movie64_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user128_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user128_movie128_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb16_user128_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user8_movie8_final64
78/78              0s 2ms/step
```

```
20/20              0s 2ms/step
Running: emb32_user8_movie8_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user8_movie16_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user8_movie16_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user8_movie32_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user8_movie32_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user8_movie64_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user8_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user8_movie128_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user8_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user16_movie8_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user16_movie8_final128
78/78              0s 3ms/step
20/20              0s 2ms/step
Running: emb32_user16_movie16_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user16_movie16_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user16_movie32_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user16_movie32_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user16_movie64_final64
78/78              0s 2ms/step
```

```
20/20              0s 2ms/step
Running: emb32_user16_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user16_movie128_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user16_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user32_movie8_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user32_movie8_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user32_movie16_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user32_movie16_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user32_movie32_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user32_movie32_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user32_movie64_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user32_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user32_movie128_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user32_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie8_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie8_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie16_final64
78/78              0s 2ms/step
```

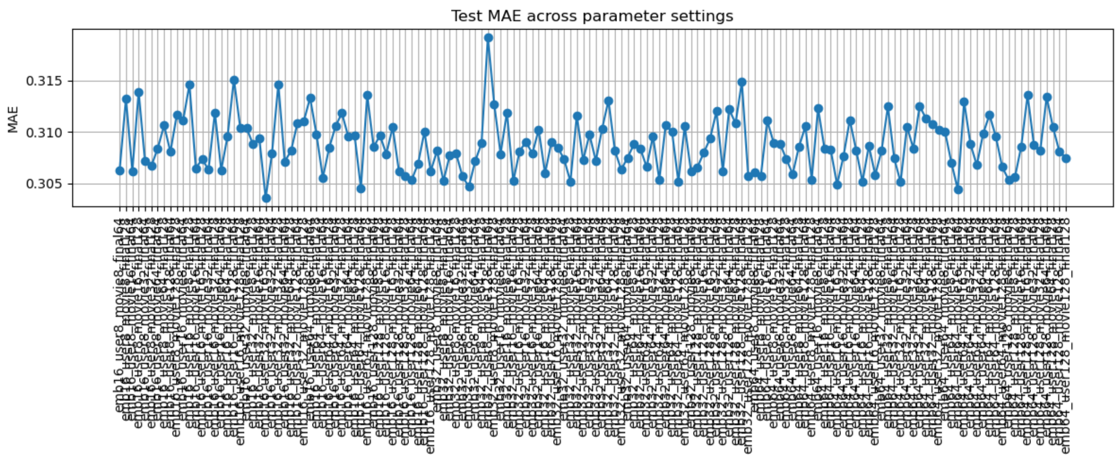
```
20/20              0s 2ms/step
Running: emb32_user64_movie16_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie32_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie32_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie64_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie128_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user64_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie8_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie8_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie16_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie16_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie32_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie32_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie64_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb32_user128_movie128_final64
78/78              0s 2ms/step
```

```
20/20              0s 2ms/step
Running: emb32_user128_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user8_movie8_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user8_movie8_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user8_movie16_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user8_movie16_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user8_movie32_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user8_movie32_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user8_movie64_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user8_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user8_movie128_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user8_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user16_movie8_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user16_movie8_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user16_movie16_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user16_movie16_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user16_movie32_final64
78/78              0s 2ms/step
```

```
20/20              0s 2ms/step
Running: emb64_user16_movie32_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user16_movie64_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user16_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user16_movie128_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user16_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie8_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie8_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie16_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie16_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie32_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie32_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie64_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie128_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user32_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user64_movie8_final64
78/78              0s 2ms/step
```

```
20/20            0s 2ms/step
Running: emb64_user64_movie8_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie16_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie16_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie32_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie32_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie64_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie64_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie128_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user64_movie128_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user128_movie8_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user128_movie8_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user128_movie16_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user128_movie16_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user128_movie32_final64
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user128_movie32_final128
78/78            0s 2ms/step
20/20            0s 2ms/step
Running: emb64_user128_movie64_final64
78/78            0s 2ms/step
```

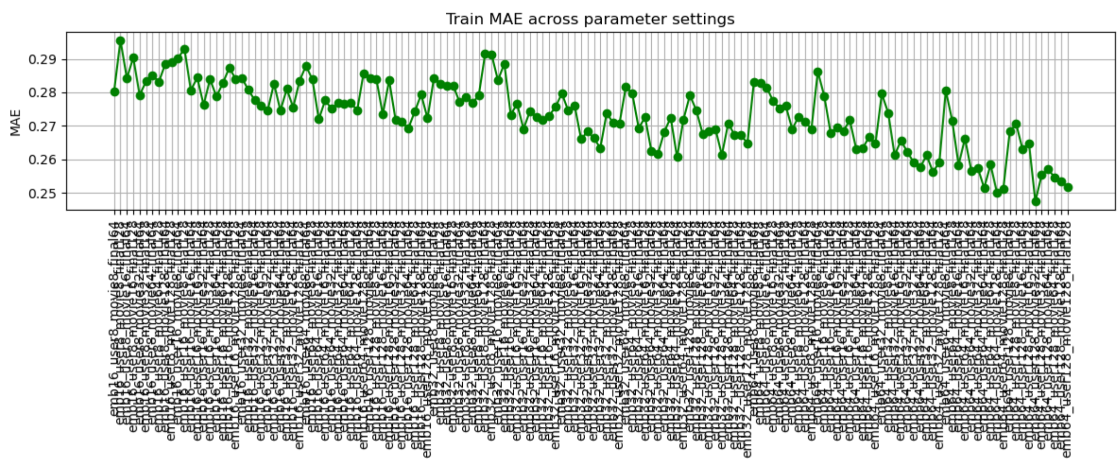
```
20/20              0s 2ms/step
Running: emb64_user128_movie64_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user128_movie128_final64
78/78              0s 2ms/step
20/20              0s 2ms/step
Running: emb64_user128_movie128_final128
78/78              0s 2ms/step
20/20              0s 2ms/step
Total time taken: 476.58 seconds.
```




```
Top 5 by MAE:
emb16_user32_movie16_final128: Test MAE=0.3036, Train MAE=0.2761
emb64_user64_movie16_final64: Test MAE=0.3044, Train MAE=0.2583
emb16_user64_movie128_final64: Test MAE=0.3045, Train MAE=0.2746
emb32_user8_movie32_final128: Test MAE=0.3047, Train MAE=0.2785
emb64_user16_movie16_final128: Test MAE=0.3049, Train MAE=0.2695

Bottom 5 by MAE:
emb16_user16_movie8_final128: Test MAE=0.3146, Train MAE=0.2931
emb16_user32_movie32_final128: Test MAE=0.3146, Train MAE=0.2827
emb32_user128_movie128_final64: Test MAE=0.3148, Train MAE=0.2672
emb16_user16_movie128_final64: Test MAE=0.3151, Train MAE=0.2873
emb32_user8_movie128_final64: Test MAE=0.3192, Train MAE=0.2916
```

1.5 Plotting training with MAE [For information]

[ ]:
```python
def plot_train_mae(results):
    keys = list(results.keys())
    mae_values = [results[k]['train_mae'] for k in keys]

    plt.figure(figsize=(12, 5))
    plt.plot(mae_values, marker='o', color='green', label='Train MAE')
    plt.xticks(ticks=range(len(keys)), labels=keys, rotation=90)
    plt.ylabel("MAE")
    plt.title("Train MAE across parameter settings")
    plt.grid(True)
    plt.tight_layout()
    plt.show()
plot_train_mae(results)
def print_top_bottom_train_mae(results):
    sorted_by_train_mae = sorted(results.items(), key=lambda x: x[1]['train_mae'])

    print("\nTop 5 by Train MAE:")
    for k, v in sorted_by_train_mae[:5]:
        print(f"{k}: Train MAE={v['train_mae']:.4f}, Test MAE={v['test_mae']:.4f}")

    print("\nBottom 5 by Train MAE:")
    for k, v in sorted_by_train_mae[-5:]:
        print(f"{k}: Train MAE={v['train_mae']:.4f}, Test MAE={v['test_mae']:.4f}")
print_top_bottom_train_mae(results)
```




```
Top 5 by Train MAE:
```

```
emb64_user128_movie32_final64: Train MAE=0.2474, Test MAE=0.3087
emb64_user64_movie128_final64: Train MAE=0.2500, Test MAE=0.3095
emb64_user64_movie128_final128: Train MAE=0.2512, Test MAE=0.3066
emb64_user64_movie64_final64: Train MAE=0.2514, Test MAE=0.3098
emb64_user128_movie128_final128: Train MAE=0.2518, Test MAE=0.3075

Bottom 5 by Train MAE:
emb16_user8_movie16_final128: Train MAE=0.2904, Test MAE=0.3139
emb32_user8_movie128_final128: Train MAE=0.2914, Test MAE=0.3127
emb32_user8_movie128_final64: Train MAE=0.2916, Test MAE=0.3192
emb16_user16_movie8_final128: Train MAE=0.2931, Test MAE=0.3146
emb16_user8_movie8_final128: Train MAE=0.2956, Test MAE=0.3132
```

1.6 Finalize Model

[ ]:
```python
import tensorflow as tf
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from itertools import product
from sklearn.metrics import mean_squared_error
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
import time

# ===============================
# 1. Set Random Seed
# ===============================
tf.random.set_seed(42)
np.random.seed(42)

# ===============================
# 2. Load Data
# ===============================
ratings = pd.read_csv('u.data', sep='\t', names=['user_id', 'item_id',␣
↪'rating', 'timestamp'])

movies = pd.read_csv(
    'u.item',
    sep='|',
    encoding='latin-1',
    names=['item_id', 'movie_title', 'release_date', 'video_release_date',␣
↪'IMDb_URL'] + [f'genre_{i}' for i in range(19)],
    usecols=range(24)
)

users = pd.read_csv(
```

```
    'u.user',
    sep='|',
    names=['user_id', 'age', 'gender', 'occupation', 'zip_code']
)

# ===============================
# 3. Preprocessing
# ===============================
ratings = ratings.sort_values('timestamp').drop_duplicates(subset=['user_id',␣
 ↪'item_id'], keep='last')
ratings = ratings.merge(movies, on='item_id')
ratings = ratings.merge(users[['user_id', 'age', 'gender', 'occupation']],␣
 ↪on='user_id')

# --- Mean Center and Normalize Ratings per User ---
user_mean = ratings.groupby('user_id')['rating'].mean()
ratings['user_mean'] = ratings['user_id'].map(user_mean)
ratings['mean_centered_rating'] = ratings['rating'] - ratings['user_mean']

# Normalize mean-centered ratings to [-1, 1] using max absolute deviation per␣
 ↪user
user_max_dev = ratings.groupby('user_id')['mean_centered_rating'].apply(lambda␣
 ↪x: x.abs().max())
ratings['user_max_dev'] = ratings['user_id'].map(user_max_dev)

# Avoid division by zero for constant ratings
ratings['user_max_dev'] = ratings['user_max_dev'].replace(0, 1)

ratings['normalized_rating'] = ratings['mean_centered_rating'] /␣
 ↪ratings['user_max_dev']

# --- Encode Features ---
genre_cols = [f'genre_{i}' for i in range(19)]
user_encoder = LabelEncoder()
movie_encoder = LabelEncoder()
ratings['user'] = user_encoder.fit_transform(ratings['user_id'])
ratings['movie'] = movie_encoder.fit_transform(ratings['item_id'])
ratings['gender'] = ratings['gender'].map({'M': 0, 'F': 1})
ratings['occupation'] = ratings['occupation'].astype('category').cat.codes

n_users = ratings['user'].nunique()
n_movies = ratings['movie'].nunique()
n_occupations = len(ratings['occupation'].unique())


# ===============================
# 4. Train-Test Split by User
```

```
# ================================
train_rows, test_rows = [], []
for _, user_ratings in ratings.groupby('user_id'):
    if len(user_ratings) < 5:
        train_rows.append(user_ratings)
    else:
        train, test = train_test_split(user_ratings, test_size=0.2,␣
↪random_state=42)
        train_rows.append(train)
        test_rows.append(test)

train_df = pd.concat(train_rows).reset_index(drop=True)
test_df = pd.concat(test_rows).reset_index(drop=True)
# ================================
# 5. Dataset Builder
# ================================
def build_dataset(df):
    return tf.data.Dataset.from_tensor_slices((
        {
            "user_id": df['user'].values,
            "movie_id": df['movie'].values,
            "age": df['age'].values.astype(np.float32).reshape(-1, 1),
            "gender": df['gender'].values,
            "occupation": df['occupation'].values,
            "genres": df[genre_cols].values.astype(np.float32)
        },
        df['normalized_rating'].values.astype(np.float32)
    )).batch(1024).prefetch(tf.data.AUTOTUNE)

train_dataset = build_dataset(train_df)
test_dataset = build_dataset(test_df)
```

[ ]:
```
import tensorflow as tf
import numpy as np
import time
from sklearn.metrics import mean_squared_error

class RecommenderNet(tf.keras.Model):
    def __init__(self, n_users, n_movies, n_occupations,
                 embedding_size=16,
                 user_dense_size=16,
                 genre_dense_size=16,  # tied to embedding_size
                 movie_dense_size=64,
                 final_dim=64):
        super(RecommenderNet, self).__init__()

        # User tower
```

```
        self.user_embedding = tf.keras.layers.Embedding(n_users, embedding_size)
        self.gender_embedding = tf.keras.layers.Embedding(2, embedding_size //␣
↪2)
        self.occupation_embedding = tf.keras.layers.Embedding(n_occupations,␣
↪embedding_size // 2)
        self.age_normalization = tf.keras.layers.Normalization()
        self.age_dense = tf.keras.layers.Dense(embedding_size // 2,␣
↪activation="relu")
        self.user_dense = tf.keras.layers.Dense(user_dense_size,␣
↪activation="relu")
        self.user_final = tf.keras.layers.Dense(final_dim)

        # Movie tower
        self.movie_embedding = tf.keras.layers.Embedding(n_movies,␣
↪embedding_size)
        self.genre_dense = tf.keras.layers.Dense(genre_dense_size,␣
↪activation="relu")
        self.movie_dense = tf.keras.layers.Dense(movie_dense_size,␣
↪activation="relu")
        self.movie_final = tf.keras.layers.Dense(final_dim)

    def call(self, inputs):
        user_vec = self.user_embedding(inputs["user_id"])
        gender_vec = self.gender_embedding(inputs["gender"])
        occupation_vec = self.occupation_embedding(inputs["occupation"])
        age_vec = self.age_dense(self.age_normalization(inputs["age"]))

        user_features = tf.concat([user_vec, gender_vec, occupation_vec,␣
↪age_vec], axis=1)
        user_features = self.user_dense(user_features)
        user_features = self.user_final(user_features)

        movie_vec = self.movie_embedding(inputs["movie_id"])
        genre_vec = self.genre_dense(inputs["genres"])
        movie_features = tf.concat([movie_vec, genre_vec], axis=1)
        movie_features = self.movie_dense(movie_features)
        movie_features = self.movie_final(movie_features)

        user_norm = tf.math.l2_normalize(user_features, axis=1)
        movie_norm = tf.math.l2_normalize(movie_features, axis=1)
        return tf.reduce_sum(user_norm * movie_norm, axis=1)


# Instantiate & Compile Model
model = RecommenderNet(
    n_users=n_users,
```

```
    n_movies=n_movies,
    n_occupations=len(ratings['occupation'].unique()),
    embedding_size=64,
    user_dense_size=128,
    genre_dense_size=64,
    movie_dense_size=32,
    final_dim=128
)

model.age_normalization.adapt(train_df['age'].values.astype(np.float32).
 ↪reshape(-1, 1))

model.compile(
    loss=tf.keras.losses.MeanSquaredError(),
    optimizer=tf.keras.optimizers.Adam(learning_rate=0.001)
)

# Train Model
start_time = time.time()
history = model.fit(train_dataset, validation_data=test_dataset, epochs=50)
end_time = time.time()
print(f"Training took {(end_time - start_time):.2f} seconds.")

# Evaluate Final Model (RMSE)

train_preds = model.predict(train_dataset)
train_rmse = np.sqrt(mean_squared_error(train_df['normalized_rating'].values,␣
 ↪train_preds))
print(f"Train RMSE: {train_rmse:.4f}")

test_preds = model.predict(test_dataset)
test_rmse = np.sqrt(mean_squared_error(test_df['normalized_rating'].values,␣
 ↪test_preds))
print(f"Test RMSE: {test_rmse:.4f}")
```

```
Epoch 1/50
78/78          10s 35ms/step -
loss: 0.1863 - val_loss: 0.1593
Epoch 2/50
78/78          2s 28ms/step -
loss: 0.1601 - val_loss: 0.1663
Epoch 3/50
78/78          2s 22ms/step -
loss: 0.1533 - val_loss: 0.1551
Epoch 4/50
78/78          2s 22ms/step -
loss: 0.1443 - val_loss: 0.1562
```

```
Epoch 5/50
78/78            3s 27ms/step -
loss: 0.1369 - val_loss: 0.1543
Epoch 6/50
78/78            4s 46ms/step -
loss: 0.1274 - val_loss: 0.1565
Epoch 7/50
78/78            3s 20ms/step -
loss: 0.1217 - val_loss: 0.1583
Epoch 8/50
78/78            2s 21ms/step -
loss: 0.1179 - val_loss: 0.1544
Epoch 9/50
78/78            2s 20ms/step -
loss: 0.1142 - val_loss: 0.1564
Epoch 10/50
78/78            3s 21ms/step -
loss: 0.1101 - val_loss: 0.1562
Epoch 11/50
78/78            2s 20ms/step -
loss: 0.1066 - val_loss: 0.1584
Epoch 12/50
78/78            3s 33ms/step -
loss: 0.1053 - val_loss: 0.1558
Epoch 13/50
78/78            2s 25ms/step -
loss: 0.1027 - val_loss: 0.1588
Epoch 14/50
78/78            2s 21ms/step -
loss: 0.1012 - val_loss: 0.1674
Epoch 15/50
78/78            2s 20ms/step -
loss: 0.1013 - val_loss: 0.1598
Epoch 16/50
78/78            3s 26ms/step -
loss: 0.1001 - val_loss: 0.1567
Epoch 17/50
78/78            3s 38ms/step -
loss: 0.0976 - val_loss: 0.1554
Epoch 18/50
78/78            4s 26ms/step -
loss: 0.0953 - val_loss: 0.1556
Epoch 19/50
78/78            2s 21ms/step -
loss: 0.0932 - val_loss: 0.1564
Epoch 20/50
78/78            3s 31ms/step -
loss: 0.0917 - val_loss: 0.1585
```

```
Epoch 21/50
78/78            3s 36ms/step -
loss: 0.0908 - val_loss: 0.1600
Epoch 22/50
78/78            3s 35ms/step -
loss: 0.0900 - val_loss: 0.1594
Epoch 23/50
78/78            2s 25ms/step -
loss: 0.0893 - val_loss: 0.1592
Epoch 24/50
78/78            2s 21ms/step -
loss: 0.0888 - val_loss: 0.1585
Epoch 25/50
78/78            2s 21ms/step -
loss: 0.0887 - val_loss: 0.1588
Epoch 26/50
78/78            2s 21ms/step -
loss: 0.0891 - val_loss: 0.1596
Epoch 27/50
78/78            2s 30ms/step -
loss: 0.0894 - val_loss: 0.1589
Epoch 28/50
78/78            3s 40ms/step -
loss: 0.0892 - val_loss: 0.1589
Epoch 29/50
78/78            2s 30ms/step -
loss: 0.0882 - val_loss: 0.1616
Epoch 30/50
78/78            2s 21ms/step -
loss: 0.0873 - val_loss: 0.1622
Epoch 31/50
78/78            2s 21ms/step -
loss: 0.0865 - val_loss: 0.1583
Epoch 32/50
78/78            3s 22ms/step -
loss: 0.0853 - val_loss: 0.1579
Epoch 33/50
78/78            2s 21ms/step -
loss: 0.0844 - val_loss: 0.1589
Epoch 34/50
78/78            3s 30ms/step -
loss: 0.0837 - val_loss: 0.1586
Epoch 35/50
78/78            2s 30ms/step -
loss: 0.0833 - val_loss: 0.1584
Epoch 36/50
78/78            2s 21ms/step -
loss: 0.0831 - val_loss: 0.1581
```

```
Epoch 37/50
78/78            2s 22ms/step -
loss: 0.0827 - val_loss: 0.1580
Epoch 38/50
78/78            2s 21ms/step -
loss: 0.0820 - val_loss: 0.1590
Epoch 39/50
78/78            3s 20ms/step -
loss: 0.0815 - val_loss: 0.1611
Epoch 40/50
78/78            2s 29ms/step -
loss: 0.0812 - val_loss: 0.1615
Epoch 41/50
78/78            3s 34ms/step -
loss: 0.0810 - val_loss: 0.1593
Epoch 42/50
78/78            2s 21ms/step -
loss: 0.0809 - val_loss: 0.1579
Epoch 43/50
78/78            2s 21ms/step -
loss: 0.0810 - val_loss: 0.1580
Epoch 44/50
78/78            2s 21ms/step -
loss: 0.0812 - val_loss: 0.1587
Epoch 45/50
78/78            3s 21ms/step -
loss: 0.0811 - val_loss: 0.1595
Epoch 46/50
78/78            2s 26ms/step -
loss: 0.0809 - val_loss: 0.1624
Epoch 47/50
78/78            3s 33ms/step -
loss: 0.0811 - val_loss: 0.1632
Epoch 48/50
78/78            4s 21ms/step -
loss: 0.0811 - val_loss: 0.1608
Epoch 49/50
78/78            2s 22ms/step -
loss: 0.0807 - val_loss: 0.1591
Epoch 50/50
78/78            2s 21ms/step -
loss: 0.0802 - val_loss: 0.1588
Training took 126.84 seconds.
78/78            2s 18ms/step
Train RMSE: 0.2758
20/20            0s 19ms/step
Test RMSE: 0.3985
```

```
[ ]: from sklearn.metrics import mean_absolute_error
test_mae = mean_absolute_error(test_df['normalized_rating'].values, test_preds)
print(f"Test MAE: {test_mae:.4f}")
```

```
Test MAE: 0.3113
```

1.7 Calculate unscaled metrics

```
[ ]: from sklearn.metrics import mean_absolute_error, mean_squared_error

# Step 1: Predict normalized ratings from model
train_preds = model.predict(train_dataset)
test_preds = model.predict(test_dataset)

# Step 2: Map back user mean and max deviation
train_user_mean = train_df['user_id'].map(user_mean)
train_user_dev = train_df['user_id'].map(user_max_dev)

test_user_mean = test_df['user_id'].map(user_mean)
test_user_dev = test_df['user_id'].map(user_max_dev)

# Step 3: Rescale predictions to original rating scale (1-5), without clipping
train_rescaled = train_preds * train_user_dev.values + train_user_mean.values
test_rescaled = test_preds * test_user_dev.values + test_user_mean.values

# Step 4: Compute RMSE and MAE on true ratings
train_rmse = np.sqrt(mean_squared_error(train_df['rating'], train_rescaled))
train_mae = mean_absolute_error(train_df['rating'], train_rescaled)

test_rmse = np.sqrt(mean_squared_error(test_df['rating'], test_rescaled))
test_mae = mean_absolute_error(test_df['rating'], test_rescaled)

# Step 5: Output
print(f"Train RMSE (1-5 scale): {train_rmse:.4f}")
print(f"Train MAE  (1-5 scale): {train_mae:.4f}")
print(f"Test RMSE  (1-5 scale): {test_rmse:.4f}")
print(f"Test MAE   (1-5 scale): {test_mae:.4f}")
```

```
78/78           1s 9ms/step
20/20           0s 9ms/step
Train RMSE (1-5 scale): 0.6607
Train MAE  (1-5 scale): 0.5135
Test RMSE  (1-5 scale): 0.9555
Test MAE   (1-5 scale): 0.7497
```

1.8 Full Unscaled matrix

[ ]:
```
import pandas as pd
import numpy as np
from tqdm import tqdm

# Get list of all user IDs and movie IDs
all_user_ids = ratings['user_id'].unique()
all_movie_ids = ratings['item_id'].unique()

# Create mapping from user_id/item_id to internal model indices (used during␣
↪training)
user_id_to_index = {uid: idx for idx, uid in enumerate(all_user_ids)}
movie_id_to_index = {mid: idx for idx, mid in enumerate(all_movie_ids)}

# Prepare auxiliary user info
users_sorted = users[users['user_id'].isin(all_user_ids)].sort_values('user_id')
user_age = users_sorted['age'].values.astype(np.float32)
user_gender = users_sorted['gender'].map({'M': 0, 'F': 1}).values
user_occupation = users_sorted['occupation'].astype('category').cat.codes.values
n_users = len(all_user_ids)

# Initialize predicted rating matrix
rating_matrix_pred = np.zeros((n_users, len(all_movie_ids)))

# Batch prediction (to avoid memory overload)
batch_size = 256
movie_id_list = list(all_movie_ids)
genre_cols = [col for col in movies.columns if col.startswith('genre_')]
n_movies = len(all_movie_ids)

for movie_batch_start in tqdm(range(0, n_movies, batch_size)):
    movie_batch_end = min(movie_batch_start + batch_size, n_movies)
    movie_batch_ids = movie_id_list[movie_batch_start:movie_batch_end]
    movie_batch_indices = [movie_id_to_index[mid] for mid in movie_batch_ids]

    genre_batch = movies.set_index('item_id').loc[movie_batch_ids][genre_cols].
↪values.astype(np.float32)

    # Repeat for all users
    user_input = {
        "user_id": np.repeat(np.arange(n_users), len(movie_batch_ids)),
        "movie_id": np.tile(movie_batch_indices, n_users),
        "age": np.repeat(user_age.reshape(-1, 1), len(movie_batch_ids), axis=0),
        "gender": np.repeat(user_gender, len(movie_batch_ids)),
        "occupation": np.repeat(user_occupation, len(movie_batch_ids)),
        "genres": np.tile(genre_batch, (n_users, 1))
```

```
    }

    # Predict
    preds = model.predict(user_input, verbose=0)
    preds = preds.reshape(n_users, len(movie_batch_ids))

    # Store in matrix
    rating_matrix_pred[:, movie_batch_start:movie_batch_end] = preds

# -------------------------------
# Unscale predictions to 1-5 scale
# -------------------------------
# Get per-user mean and max deviation used during normalization
user_mean_array = ratings.groupby('user_id')['user_mean'].first().
↪reindex(all_user_ids).values
user_dev_array = ratings.groupby('user_id')['user_max_dev'].first().
↪reindex(all_user_ids).values

# Handle any NaNs from reindexing
user_dev_array = np.where(user_dev_array == 0, 1, user_dev_array)

# Unnormalize
rating_matrix_unscaled = rating_matrix_pred * user_dev_array[:, None] +␣
↪user_mean_array[:, None]

# Optionally clip to [1, 5] range
# rating_matrix_unscaled = np.clip(rating_matrix_unscaled, 1, 5)
# Apply scaling from [-1, 1] → [1, 5]
rating_matrix_scaled_1_5 = 2 * rating_matrix_pred + 3

# Convert to DataFrame for readability
rating_df_scaled = pd.DataFrame(rating_matrix_scaled_1_5, index=all_user_ids,␣
↪columns=all_movie_ids)
```

```
100%|      | 7/7 [02:22<00:00, 20.39s/it]
```

1.9 Top K Predictions

[ ]:
```
top_k = 5
user_ids_to_check = [121, 31, 77]

# Invert index map if needed
index_to_user_id = {idx: uid for uid, idx in user_id_to_index.items()}
user_id_to_row = {uid: idx for idx, uid in enumerate(all_user_ids)}  # for␣
↪direct row lookup

for uid in user_ids_to_check:
```

```
    user_row = user_id_to_row[uid]
    predicted_scores = rating_matrix_scaled_1_5[user_row]

    # Get top k movie indices
    top_k_indices = np.argsort(predicted_scores)[::-1][:top_k]
    top_movie_ids = [all_movie_ids[i] for i in top_k_indices]
    top_scores = predicted_scores[top_k_indices]

    # Get movie titles
    titles = movies.set_index('item_id').loc[top_movie_ids]['movie_title'].
↪values

    print(f"\nTop {top_k} Recommendations for User {uid}:")
    for mid, title, score in zip(top_movie_ids, titles, top_scores):
        print(f"  Movie ID: {mid}, Title: {title}, Predicted Rating: {score:.
↪3f}")
```

```
Top 5 Recommendations for User 121:
  Movie ID: 1204, Title: To Be or Not to Be (1942), Predicted Rating: 4.289
  Movie ID: 175, Title: Brazil (1985), Predicted Rating: 4.242
  Movie ID: 493, Title: Thin Man, The (1934), Predicted Rating: 4.210
  Movie ID: 7, Title: Twelve Monkeys (1995), Predicted Rating: 4.175
  Movie ID: 134, Title: Citizen Kane (1941), Predicted Rating: 4.158

Top 5 Recommendations for User 31:
  Movie ID: 1086, Title: It's My Party (1995), Predicted Rating: 4.267
  Movie ID: 218, Title: Cape Fear (1991), Predicted Rating: 4.116
  Movie ID: 888, Title: One Night Stand (1997), Predicted Rating: 4.113
  Movie ID: 1227, Title: Awfully Big Adventure, An (1995), Predicted Rating:
4.041
  Movie ID: 1053, Title: Now and Then (1995), Predicted Rating: 4.001

Top 5 Recommendations for User 77:
  Movie ID: 1334, Title: Somebody to Love (1994), Predicted Rating: 4.317
  Movie ID: 750, Title: Amistad (1997), Predicted Rating: 4.316
  Movie ID: 1634, Title: Etz Hadomim Tafus (Under the Domin Tree) (1994),
Predicted Rating: 4.309
  Movie ID: 1053, Title: Now and Then (1995), Predicted Rating: 4.281
  Movie ID: 1397, Title: Of Human Bondage (1934), Predicted Rating: 4.273
```

1.10 Get ratings for users for given movie IDs

[ ]:
```
# Target movie IDs
target_movie_ids = [121, 31, 77]

# Subset the scaled prediction DataFrame
```

```
ratings_selected = rating_matrix_unscaled[target_movie_ids]␣
↪### rating_df_scaled

# Display the DataFrame (rows = user_id, columns = movie_id)
print(ratings_selected)
```

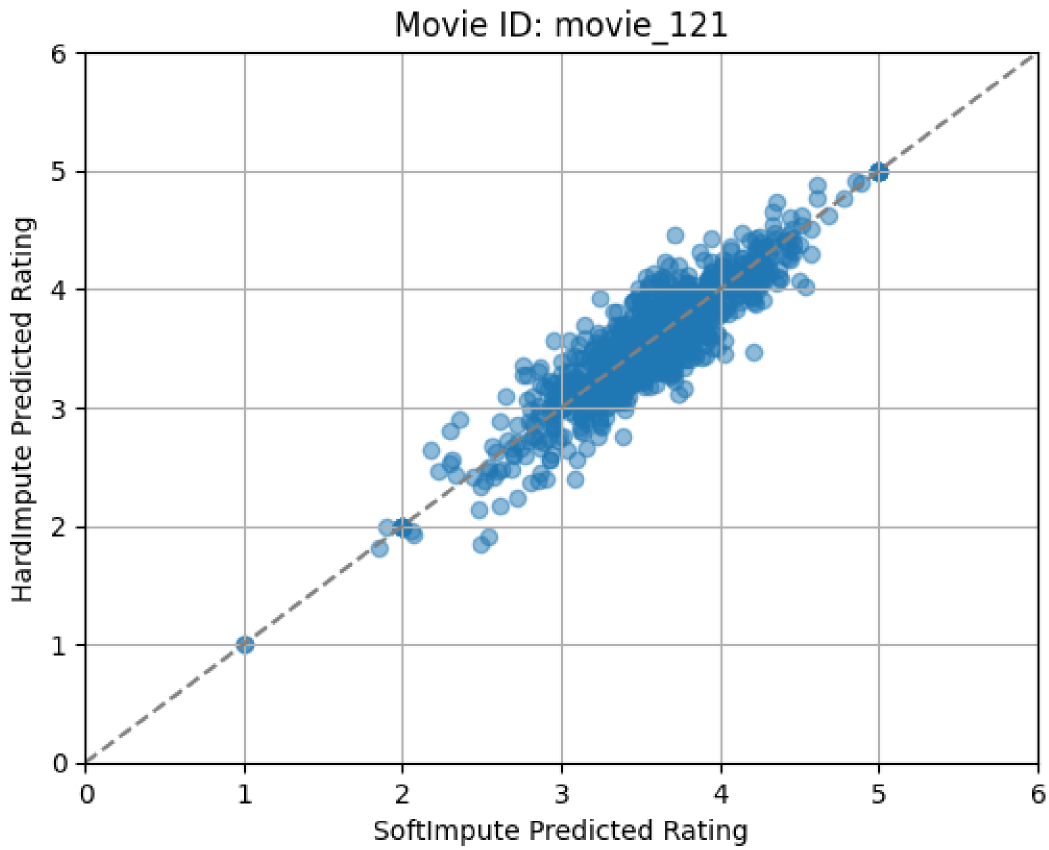
```
[[3.61337529 3.62286868 2.41613218 … 4.24206681 2.61512676 4.1925372 ]
 [4.69120807 3.45364671 4.3192492  … 3.49035481 4.3149947  3.63313635]
 [4.48688099 2.93901983 3.04648335 … 3.36250455 4.11983583 3.67295838]]
```

2 Comparitive analysis

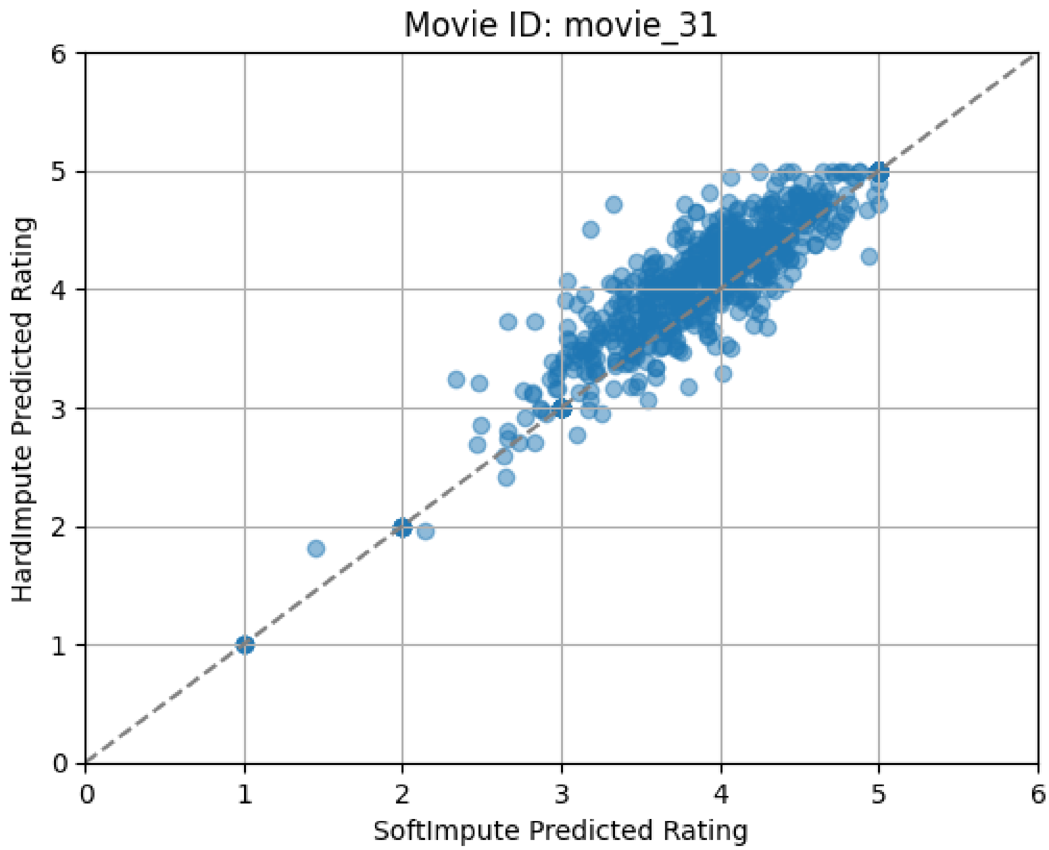
[ ]:
```
import pandas as pd
import matplotlib.pyplot as plt

# Step 1: Load predicted ratings
soft = pd.read_csv('softimpute_full_predictions.csv', index_col=0)
hard = pd.read_csv('hardimpute_full_predictions.csv', index_col=0)
```

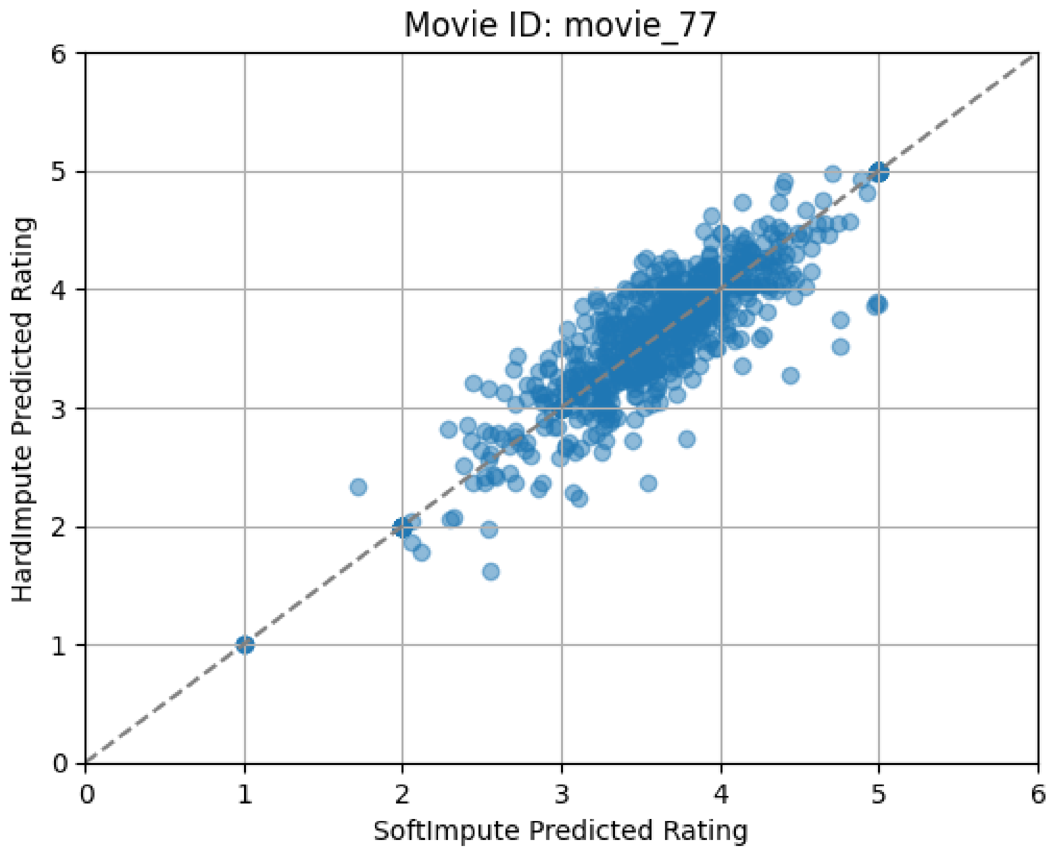
3 Plots S VS H

[ ]:
```
import matplotlib.pyplot as plt

# Step: Define movie columns to inspect
movie_ids = ['movie_121', 'movie_31', 'movie_77']

for movie_id in movie_ids:
    soft_movie_ratings = soft[movie_id].values
    hard_movie_ratings = hard[movie_id].values

    plt.figure()
    plt.scatter(soft_movie_ratings, hard_movie_ratings, alpha=0.5)
    plt.xlabel('SoftImpute Predicted Rating')
    plt.ylabel('HardImpute Predicted Rating')
    plt.title(f'Movie ID: {movie_id}')
    plt.grid(True)
    plt.plot([0, 6], [0, 6], color='gray', linestyle='--')  # identity line
    plt.xlim(0, 6)
    plt.ylim(0, 6)
    plt.show()
```



Movie ID: movie_121
HardImpute Predicted Rating
SoftImpute Predicted Rating
0
1
2
3
4
5
6


Movie ID: movie_31
HardImpute Predicted Rating
SoftImpute Predicted Rating
0
1
2
3
4
5
6

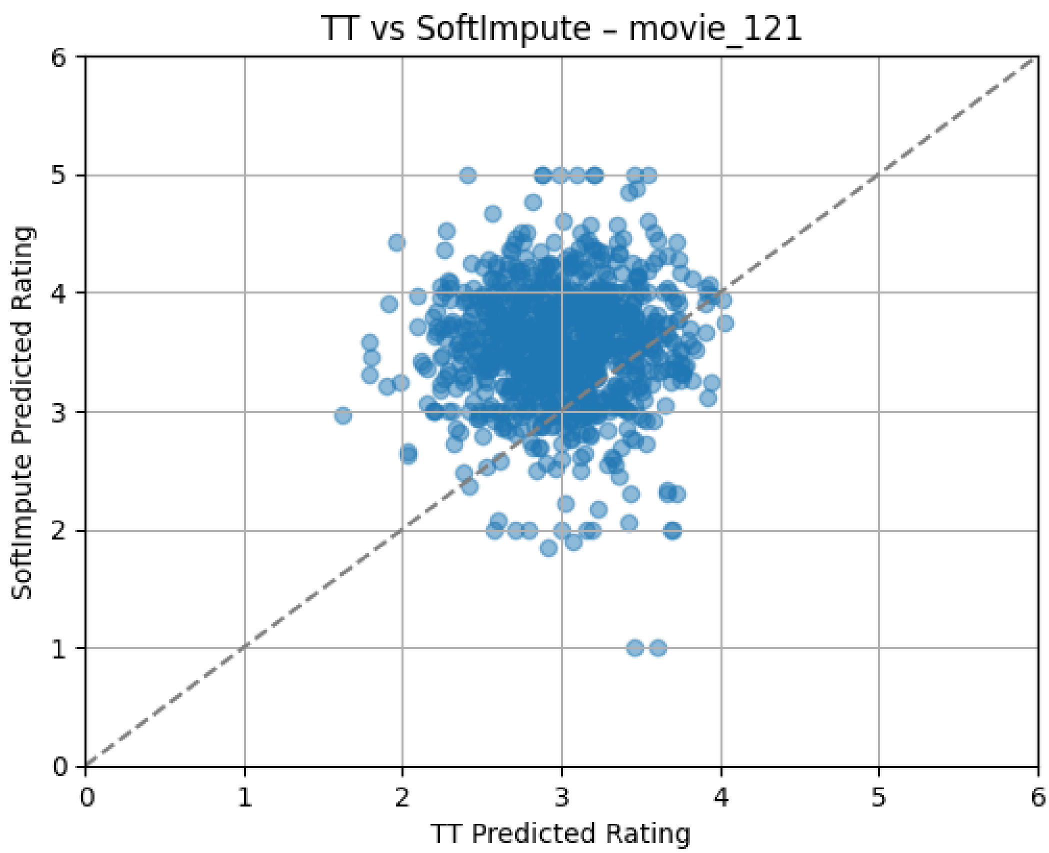


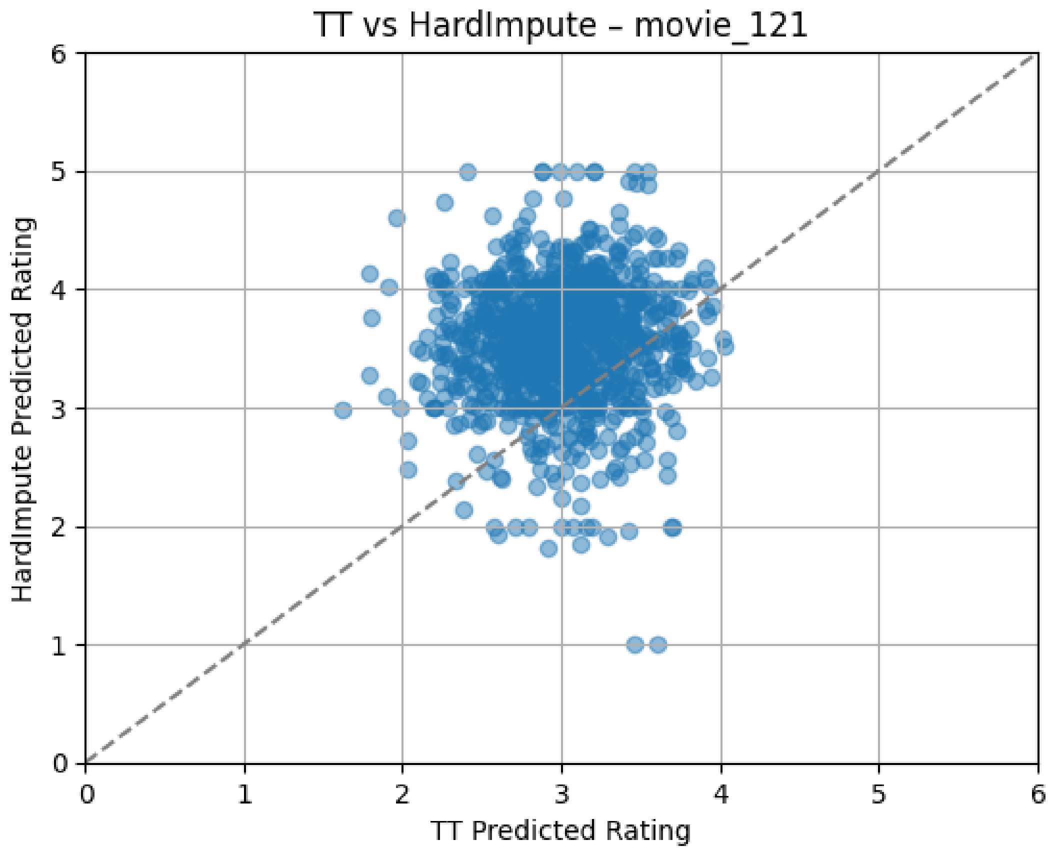
4 Plots SH vs NN

[ ]:
```
# Step 1: Ensure column names in TT match Soft/HardImpute
tt = rating_df_scaled.rename(columns={121: 'movie_121', 31: 'movie_31', 77:␣
↪'movie_77'})

# Step 2: Plot TT vs SoftImpute and TT vs HardImpute
for movie_id in ['movie_121', 'movie_31', 'movie_77']:
    tt_ratings = tt[movie_id].values
    soft_ratings = soft[movie_id].values
    hard_ratings = hard[movie_id].values
    print(tt_ratings.shape)
    print(soft_ratings.shape)

    # TT vs SoftImpute
    plt.figure()
    plt.scatter(tt_ratings, soft_ratings, alpha=0.5)
    plt.xlabel('TT Predicted Rating')
    plt.ylabel('SoftImpute Predicted Rating')
```

```
    plt.title(f'TT vs SoftImpute - {movie_id}')
    plt.grid(True)
    plt.plot([0, 6], [0, 6], color='gray', linestyle='--')  # identity line
    plt.xlim(0, 6)
    plt.ylim(0, 6)
    plt.show()

    # TT vs HardImpute
    plt.figure()
    plt.scatter(tt_ratings, hard_ratings, alpha=0.5)
    plt.xlabel('TT Predicted Rating')
    plt.ylabel('HardImpute Predicted Rating')
    plt.title(f'TT vs HardImpute - {movie_id}')
    plt.grid(True)
    plt.plot([0, 6], [0, 6], color='gray', linestyle='--')  # identity line
    plt.xlim(0, 6)
    plt.ylim(0, 6)
    plt.show()
```

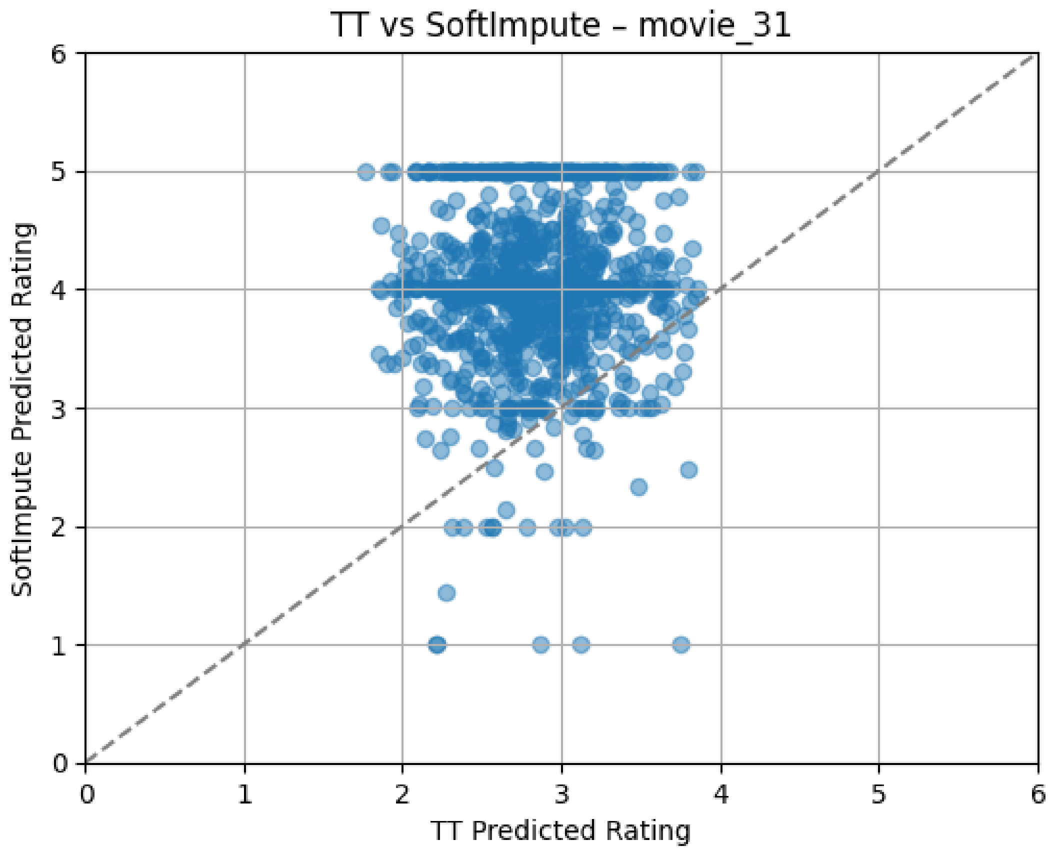
```
(943,)
(943,)
```

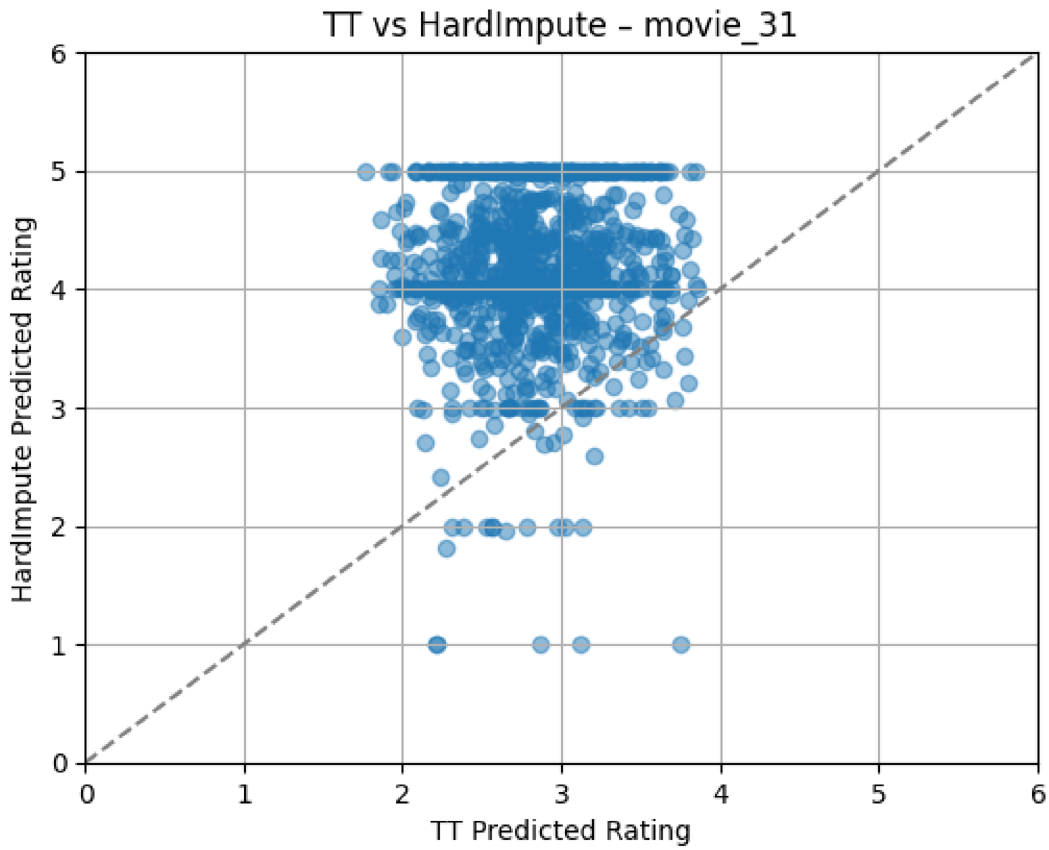




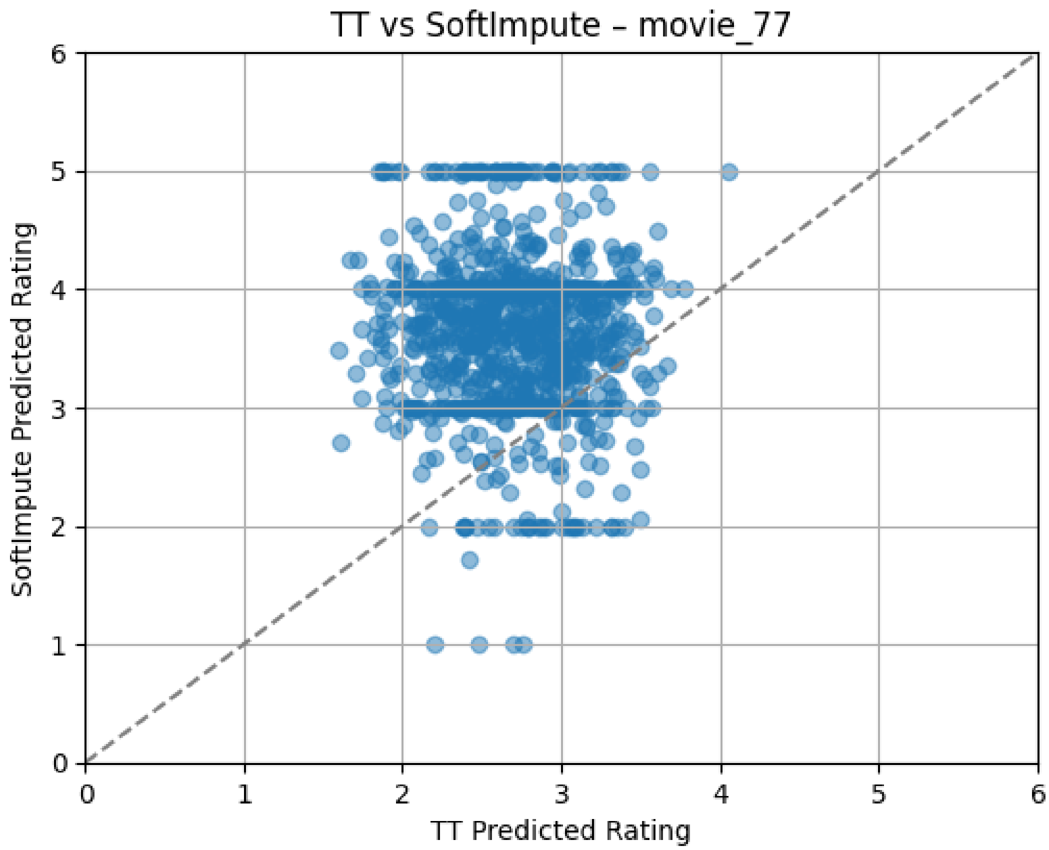
TT vs HardImpute – movie_121
HardImpute Predicted Rating
TT Predicted Rating
0
1
2
3
4
5
6

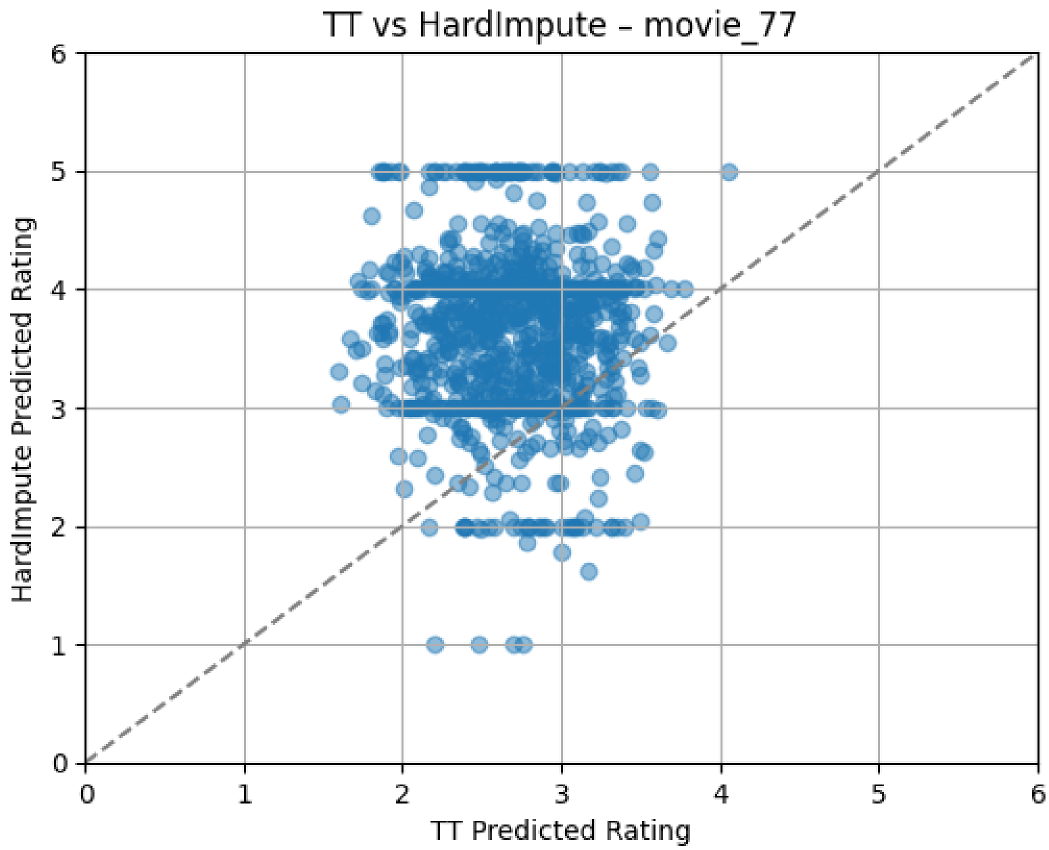

```
(943,)
(943,)
```


TT vs SoftImpute – movie_31
SoftImpute Predicted Rating
TT Predicted Rating
0
1
2
3
4
5
6


TT vs HardImpute – movie_31
HardImpute Predicted Rating
TT Predicted Rating
0
1
2
3
4
5
6
0
1
2
3
4
5
6

```
(943,)
(943,)
```


TT vs SoftImpute – movie_77
SoftImpute Predicted Rating
TT Predicted Rating
0
1
2
3
4
5
6



[ ]:
```
from scipy.stats import pearsonr

# Ensure TT movie column names match format in Soft/Hard
tt = rating_df_scaled.copy()
tt.columns = [f"movie_{mid}" for mid in tt.columns]

# Ensure all movies are shared
common_movies = set(tt.columns).intersection(soft.columns).intersection(hard.
 ↪columns)

# Initialize lists
corr_soft_hard = []
corr_soft_tt = []
corr_hard_tt = []

# Loop over all common movies
for movie_id in common_movies:
    s = soft[movie_id].values
    h = hard[movie_id].values
```

```
    t = tt[movie_id].values

    # Compute Pearson correlations
    corr_soft_hard.append(pearsonr(s, h)[0])
    corr_soft_tt.append(pearsonr(s, t)[0])
    corr_hard_tt.append(pearsonr(h, t)[0])

# Convert to arrays
corr_soft_hard = np.array(corr_soft_hard)
corr_soft_tt = np.array(corr_soft_tt)
corr_hard_tt = np.array(corr_hard_tt)

# Print summary statistics
print("Correlation Summary:")
print("Soft vs HardImpute:")
print(f"  Mean: {corr_soft_hard.mean():.4f}, Min: {corr_soft_hard.min():.4f},␣
↪Max: {corr_soft_hard.max():.4f}")

print("Soft vs Two-Tower:")
print(f"  Mean: {corr_soft_tt.mean():.4f}, Min: {corr_soft_tt.min():.4f}, Max:␣
↪{corr_soft_tt.max():.4f}")

print("Hard vs Two-Tower:")
print(f"  Mean: {corr_hard_tt.mean():.4f}, Min: {corr_hard_tt.min():.4f}, Max:␣
↪{corr_hard_tt.max():.4f}")
```

```
Correlation Summary:
Soft vs HardImpute:
  Mean: 0.9240, Min: 0.6211, Max: 1.0000
Soft vs Two-Tower:
  Mean: 0.0008, Min: -0.0903, Max: 0.0951
Hard vs Two-Tower:
  Mean: 0.0013, Min: -0.0925, Max: 0.1070
```

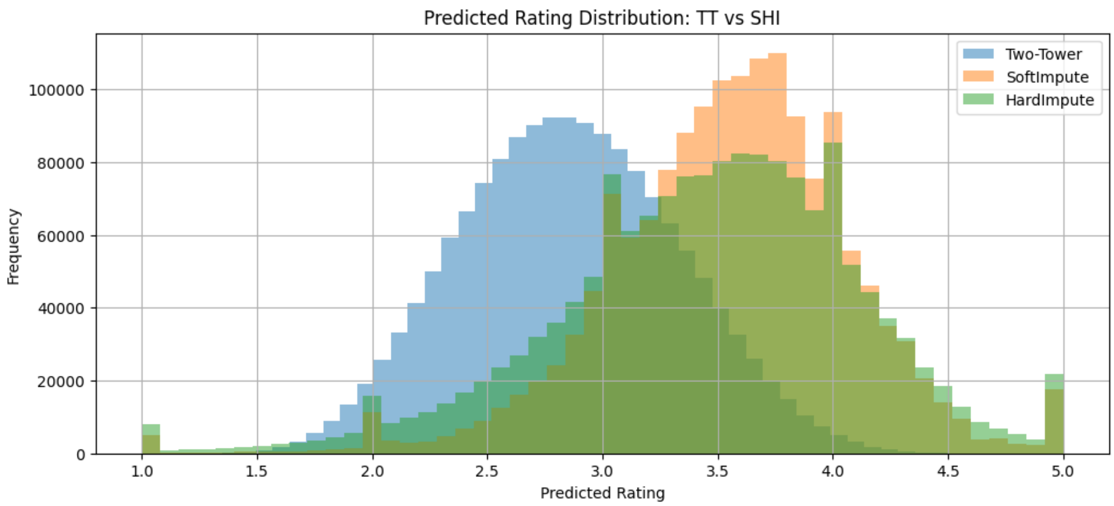
[ ]:

```

```

[ ]:

```
import matplotlib.pyplot as plt

plt.figure(figsize=(12, 5))
plt.hist(rating_df_scaled.values.flatten(), bins=50, alpha=0.5,␣
↪label='Two-Tower')
plt.hist(soft.values.flatten(), bins=50, alpha=0.5, label='SoftImpute')
plt.hist(hard.values.flatten(), bins=50, alpha=0.5, label='HardImpute')
plt.xlabel("Predicted Rating")
plt.ylabel("Frequency")
plt.title("Predicted Rating Distribution: TT vs SHI")
plt.legend()
```

```python
plt.grid(True)
plt.show()
```



[ ]:
```python

```

4.1 Average corellations

[ ]:
```python
from scipy.stats import pearsonr

# Ensure TT movie column names match format in Soft/Hard
tt = rating_df_scaled.copy()
tt.columns = [f"movie_{mid}" for mid in tt.columns]

# Ensure all movies are shared
common_movies = set(tt.columns).intersection(soft.columns).intersection(hard.
 ↪columns)

# Initialize lists
corr_soft_hard = []
corr_soft_tt = []
corr_hard_tt = []

# Loop over all common movies
for movie_id in common_movies:
    s = soft[movie_id].values
    h = hard[movie_id].values
    t = tt[movie_id].values

    # Compute Pearson correlations
```

```
    corr_soft_hard.append(pearsonr(s, h)[0])
    corr_soft_tt.append(pearsonr(s, t)[0])
    corr_hard_tt.append(pearsonr(h, t)[0])

# Convert to arrays
corr_soft_hard = np.array(corr_soft_hard)
corr_soft_tt = np.array(corr_soft_tt)
corr_hard_tt = np.array(corr_hard_tt)

# Print summary statistics
print("Correlation Summary:")
print("Soft vs HardImpute:")
print(f"  Mean: {corr_soft_hard.mean():.4f}, Min: {corr_soft_hard.min():.4f},␣
↪Max: {corr_soft_hard.max():.4f}")

print("Soft vs Two-Tower:")
print(f"  Mean: {corr_soft_tt.mean():.4f}, Min: {corr_soft_tt.min():.4f}, Max:␣
↪{corr_soft_tt.max():.4f}")

print("Hard vs Two-Tower:")
print(f"  Mean: {corr_hard_tt.mean():.4f}, Min: {corr_hard_tt.min():.4f}, Max:␣
↪{corr_hard_tt.max():.4f}")
```

```
Correlation Summary:
Soft vs HardImpute:
  Mean: 0.9240, Min: 0.6211, Max: 1.0000
Soft vs Two-Tower:
  Mean: 0.0027, Min: -0.0960, Max: 0.1125
Hard vs Two-Tower:
  Mean: 0.0031, Min: -0.1003, Max: 0.1078
```

4.2 Average Overlaps

[ ]:
```
# Step 1: Rename TT index to match 'user_X' format
tt.index = [f"user_{i}" for i in range(len(tt))]

# Step 2: Define function to get top-k movies per user
def top_k_movies(df, k=10):
    """Returns a dict mapping user_id -> set of top-k movie_ids."""
    return {
        user_id: set(row.sort_values(ascending=False).head(k).index)
        for user_id, row in df.iterrows()
    }

# Step 3: Generate top-10 recommendations for each model
k = 10
top_soft = top_k_movies(soft, k)
```

```
top_hard = top_k_movies(hard, k)
top_tt = top_k_movies(tt, k)

# Step 4: Function to compute overlap counts
def compute_overlap(top_a, top_b):
    overlaps = []
    for uid in top_a.keys():  # assumes all user IDs match
        overlap_count = len(top_a[uid].intersection(top_b[uid]))
        overlaps.append(overlap_count)
    return np.array(overlaps)

# Step 5: Compute overlaps
overlap_soft_hard = compute_overlap(top_soft, top_hard)
overlap_soft_tt = compute_overlap(top_soft, top_tt)
overlap_hard_tt = compute_overlap(top_hard, top_tt)

# Step 6: Print summary
print("Top-10 Overlap Summary (Number of Common Movies):")
print(f"Soft vs HardImpute:   Mean = {overlap_soft_hard.mean():.2f}, Min =␣
↪{overlap_soft_hard.min()}, Max = {overlap_soft_hard.max()}")
print(f"Soft vs Two-Tower:    Mean = {overlap_soft_tt.mean():.2f}, Min =␣
↪{overlap_soft_tt.min()}, Max = {overlap_soft_tt.max()}")
print(f"Hard vs Two-Tower:    Mean = {overlap_hard_tt.mean():.2f}, Min =␣
↪{overlap_hard_tt.min()}, Max = {overlap_hard_tt.max()}")
```

```
Top-10 Overlap Summary (Number of Common Movies):
Soft vs HardImpute:   Mean = 5.75, Min = 0, Max = 10
Soft vs Two-Tower:    Mean = 0.08, Min = 0, Max = 2
Hard vs Two-Tower:    Mean = 0.08, Min = 0, Max = 2
```

[ ]:
```
top_soft
```

[ ]:
```
{'user_0': {'movie_216',
  'movie_289',
  'movie_377',
  'movie_389',
  'movie_431',
  'movie_438',
  'movie_512',
  'movie_550',
  'movie_656',
  'movie_834'},
 'user_1': {'movie_156',
  'movie_172',
  'movie_31',
  'movie_476',
  'movie_506',
```

```
 'movie_51',
 'movie_543',
 'movie_746',
 'movie_87',
 'movie_970'},
'user_2': {'movie_101',
 'movie_158',
 'movie_329',
 'movie_34',
 'movie_357',
 'movie_359',
 'movie_367',
 'movie_52',
 'movie_561',
 'movie_58'},
'user_3': {'movie_103',
 'movie_111',
 'movie_1619',
 'movie_431',
 'movie_445',
 'movie_492',
 'movie_502',
 'movie_64',
 'movie_77',
 'movie_88'},
'user_4': {'movie_184',
 'movie_200',
 'movie_265',
 'movie_357',
 'movie_408',
 'movie_51',
 'movie_52',
 'movie_652',
 'movie_719',
 'movie_98'},
'user_5': {'movie_24',
 'movie_34',
 'movie_357',
 'movie_359',
 'movie_36',
 'movie_367',
 'movie_389',
 'movie_57',
 'movie_622',
 'movie_77'},
'user_6': {'movie_118',
 'movie_125',
```

```
 'movie_38',
 'movie_389',
 'movie_401',
 'movie_413',
 'movie_538',
 'movie_808',
 'movie_862',
 'movie_94'},
'user_7': {'movie_156',
 'movie_24',
 'movie_31',
 'movie_403',
 'movie_408',
 'movie_433',
 'movie_547',
 'movie_618',
 'movie_657',
 'movie_694'},
'user_8': {'movie_0',
 'movie_21',
 'movie_236',
 'movie_239',
 'movie_24',
 'movie_240',
 'movie_260',
 'movie_5',
 'movie_942',
 'movie_989'},
'user_9': {'movie_295',
 'movie_329',
 'movie_342',
 'movie_355',
 'movie_454',
 'movie_485',
 'movie_49',
 'movie_538',
 'movie_546',
 'movie_561'},
'user_10': {'movie_12',
 'movie_200',
 'movie_216',
 'movie_329',
 'movie_356',
 'movie_357',
 'movie_479',
 'movie_518',
 'movie_722',
```

```
 'movie_888'},
'user_11': {'movie_11',
 'movie_1389',
 'movie_26',
 'movie_354',
 'movie_377',
 'movie_491',
 'movie_556',
 'movie_60',
 'movie_73',
 'movie_8'},
'user_12': {'movie_20',
 'movie_24',
 'movie_31',
 'movie_50',
 'movie_56',
 'movie_57',
 'movie_60',
 'movie_695',
 'movie_76',
 'movie_877'},
'user_13': {'movie_1',
 'movie_10',
 'movie_118',
 'movie_315',
 'movie_342',
 'movie_345',
 'movie_359',
 'movie_431',
 'movie_496',
 'movie_657'},
'user_14': {'movie_12',
 'movie_156',
 'movie_200',
 'movie_247',
 'movie_264',
 'movie_293',
 'movie_318',
 'movie_408',
 'movie_465',
 'movie_719'},
'user_15': {'movie_100',
 'movie_101',
 'movie_102',
 'movie_108',
 'movie_311',
 'movie_320',
```

```
 'movie_34',
 'movie_36',
 'movie_665',
 'movie_672'},
'user_16': {'movie_1211',
 'movie_254',
 'movie_262',
 'movie_273',
 'movie_288',
 'movie_318',
 'movie_491',
 'movie_669',
 'movie_693',
 'movie_987'},
'user_17': {'movie_189',
 'movie_200',
 'movie_230',
 'movie_307',
 'movie_321',
 'movie_329',
 'movie_431',
 'movie_436',
 'movie_471',
 'movie_979'},
'user_18': {'movie_161',
 'movie_355',
 'movie_367',
 'movie_394',
 'movie_470',
 'movie_49',
 'movie_77',
 'movie_876',
 'movie_883',
 'movie_95'},
'user_19': {'movie_275',
 'movie_384',
 'movie_389',
 'movie_436',
 'movie_480',
 'movie_58',
 'movie_638',
 'movie_780',
 'movie_781',
 'movie_795'},
'user_20': {'movie_1076',
 'movie_12',
 'movie_1246',
```

```
 'movie_49',
 'movie_647',
 'movie_712',
 'movie_797',
 'movie_8',
 'movie_864',
 'movie_931'},
'user_21': {'movie_1115',
 'movie_1287',
 'movie_389',
 'movie_46',
 'movie_67',
 'movie_693',
 'movie_753',
 'movie_795',
 'movie_888',
 'movie_988'},
'user_22': {'movie_174',
 'movie_189',
 'movie_288',
 'movie_297',
 'movie_512',
 'movie_522',
 'movie_524',
 'movie_574',
 'movie_618',
 'movie_856'},
'user_23': {'movie_1097',
 'movie_44',
 'movie_47',
 'movie_49',
 'movie_496',
 'movie_517',
 'movie_57',
 'movie_694',
 'movie_729',
 'movie_750'},
'user_24': {'movie_112',
 'movie_509',
 'movie_528',
 'movie_529',
 'movie_531',
 'movie_57',
 'movie_58',
 'movie_590',
 'movie_693',
 'movie_719'},
```

```
'user_25': {'movie_20',
 'movie_22',
 'movie_27',
 'movie_274',
 'movie_293',
 'movie_320',
 'movie_540',
 'movie_55',
 'movie_7',
 'movie_8'},
'user_26': {'movie_253',
 'movie_288',
 'movie_300',
 'movie_307',
 'movie_31',
 'movie_357',
 'movie_433',
 'movie_513',
 'movie_568',
 'movie_618'},
'user_27': {'movie_1022',
 'movie_148',
 'movie_198',
 'movie_222',
 'movie_28',
 'movie_305',
 'movie_404',
 'movie_432',
 'movie_665',
 'movie_672'},
'user_28': {'movie_216',
 'movie_29',
 'movie_333',
 'movie_355',
 'movie_356',
 'movie_359',
 'movie_38',
 'movie_49',
 'movie_695',
 'movie_73'},
'user_29': {'movie_1123',
 'movie_139',
 'movie_242',
 'movie_355',
 'movie_483',
 'movie_497',
 'movie_502',
```

```
 'movie_568',
 'movie_648',
 'movie_855'},
'user_30': {'movie_112',
 'movie_239',
 'movie_240',
 'movie_444',
 'movie_49',
 'movie_52',
 'movie_55',
 'movie_615',
 'movie_629',
 'movie_646'},
'user_31': {'movie_31',
 'movie_34',
 'movie_36',
 'movie_408',
 'movie_544',
 'movie_66',
 'movie_694',
 'movie_77',
 'movie_78',
 'movie_884'},
'user_32': {'movie_161',
 'movie_174',
 'movie_309',
 'movie_347',
 'movie_408',
 'movie_438',
 'movie_471',
 'movie_52',
 'movie_56',
 'movie_57'},
'user_33': {'movie_240',
 'movie_38',
 'movie_431',
 'movie_436',
 'movie_550',
 'movie_57',
 'movie_617',
 'movie_63',
 'movie_694',
 'movie_721'},
'user_34': {'movie_1310',
 'movie_148',
 'movie_189',
 'movie_24',
```

```
 'movie_289',
 'movie_371',
 'movie_427',
 'movie_719',
 'movie_751',
 'movie_770'},
'user_35': {'movie_176',
 'movie_340',
 'movie_364',
 'movie_403',
 'movie_594',
 'movie_60',
 'movie_647',
 'movie_77',
 'movie_86',
 'movie_960'},
'user_36': {'movie_289',
 'movie_321',
 'movie_36',
 'movie_408',
 'movie_491',
 'movie_508',
 'movie_58',
 'movie_60',
 'movie_719',
 'movie_839'},
'user_37': {'movie_22',
 'movie_24',
 'movie_277',
 'movie_292',
 'movie_311',
 'movie_321',
 'movie_329',
 'movie_36',
 'movie_50',
 'movie_623'},
'user_38': {'movie_22',
 'movie_29',
 'movie_53',
 'movie_54',
 'movie_62',
 'movie_649',
 'movie_657',
 'movie_669',
 'movie_673',
 'movie_697'},
'user_39': {'movie_1047',
```

```
 'movie_1063',
 'movie_113',
 'movie_118',
 'movie_125',
 'movie_49',
 'movie_740',
 'movie_873',
 'movie_884',
 'movie_99'},
'user_40': {'movie_130',
 'movie_139',
 'movie_184',
 'movie_200',
 'movie_243',
 'movie_297',
 'movie_325',
 'movie_357',
 'movie_49',
 'movie_719'},
'user_41': {'movie_29',
 'movie_492',
 'movie_497',
 'movie_506',
 'movie_513',
 'movie_534',
 'movie_58',
 'movie_695',
 'movie_719',
 'movie_751'},
'user_42': {'movie_12',
 'movie_24',
 'movie_4',
 'movie_44',
 'movie_462',
 'movie_471',
 'movie_49',
 'movie_500',
 'movie_502',
 'movie_751'},
'user_43': {'movie_405',
 'movie_52',
 'movie_60',
 'movie_61',
 'movie_623',
 'movie_806',
 'movie_814',
 'movie_822',
```

```
 'movie_881',
 'movie_883'},
'user_44': {'movie_1104',
 'movie_1319',
 'movie_139',
 'movie_1456',
 'movie_346',
 'movie_380',
 'movie_407',
 'movie_45',
 'movie_608',
 'movie_810'},
'user_45': {'movie_111',
 'movie_31',
 'movie_36',
 'movie_46',
 'movie_49',
 'movie_63',
 'movie_693',
 'movie_734',
 'movie_873',
 'movie_992'},
'user_46': {'movie_101',
 'movie_157',
 'movie_468',
 'movie_49',
 'movie_52',
 'movie_614',
 'movie_629',
 'movie_719',
 'movie_77',
 'movie_87'},
'user_47': {'movie_101',
 'movie_26',
 'movie_52',
 'movie_538',
 'movie_586',
 'movie_600',
 'movie_66',
 'movie_68',
 'movie_719',
 'movie_89'},
'user_48': {'movie_100',
 'movie_103',
 'movie_11',
 'movie_300',
 'movie_357',
```

'movie_382',
'movie_408',
'movie_719',
'movie_77',
'movie_78'},
'user_49': {'movie_156',
'movie_200',
'movie_239',
'movie_307',
'movie_31',
'movie_355',
'movie_357',
'movie_647',
'movie_719',
'movie_86'},
'user_50': {'movie_1042',
'movie_1058',
'movie_144',
'movie_1519',
'movie_213',
'movie_719',
'movie_815',
'movie_860',
'movie_928',
'movie_960'},
'user_51': {'movie_101',
'movie_125',
'movie_157',
'movie_186',
'movie_24',
'movie_247',
'movie_277',
'movie_31',
'movie_357',
'movie_86'},
'user_52': {'movie_22',
'movie_31',
'movie_368',
'movie_377',
'movie_387',
'movie_399',
'movie_401',
'movie_419',
'movie_431',
'movie_49'},
'user_53': {'movie_102',
'movie_31',

```
 'movie_431',
 'movie_452',
 'movie_456',
 'movie_502',
 'movie_57',
 'movie_66',
 'movie_68',
 'movie_78'},
'user_54': {'movie_33',
 'movie_423',
 'movie_438',
 'movie_447',
 'movie_456',
 'movie_476',
 'movie_52',
 'movie_53',
 'movie_801',
 'movie_923'},
'user_55': {'movie_101',
 'movie_112',
 'movie_293',
 'movie_302',
 'movie_33',
 'movie_357',
 'movie_52',
 'movie_53',
 'movie_57',
 'movie_816'},
'user_56': {'movie_22',
 'movie_31',
 'movie_311',
 'movie_49',
 'movie_538',
 'movie_571',
 'movie_608',
 'movie_611',
 'movie_623',
 'movie_635'},
'user_57': {'movie_1028',
 'movie_144',
 'movie_17',
 'movie_231',
 'movie_274',
 'movie_364',
 'movie_408',
 'movie_52',
 'movie_547',
```

```
 'movie_99'},
'user_58': {'movie_1',
 'movie_1145',
 'movie_1150',
 'movie_26',
 'movie_66',
 'movie_77',
 'movie_94',
 'movie_943',
 'movie_950',
 'movie_96'},
'user_59': {'movie_148',
 'movie_200',
 'movie_216',
 'movie_297',
 'movie_428',
 'movie_49',
 'movie_5',
 'movie_561',
 'movie_622',
 'movie_64'},
'user_60': {'movie_1646',
 'movie_475',
 'movie_498',
 'movie_541',
 'movie_574',
 'movie_58',
 'movie_66',
 'movie_673',
 'movie_699',
 'movie_723'},
'user_61': {'movie_10',
 'movie_12',
 'movie_156',
 'movie_157',
 'movie_200',
 'movie_239',
 'movie_24',
 'movie_275',
 'movie_382',
 'movie_400'},
'user_62': {'movie_166',
 'movie_191',
 'movie_25',
 'movie_26',
 'movie_309',
 'movie_321',
```

```
 'movie_408',
 'movie_719',
 'movie_816',
 'movie_87'},
'user_63': {'movie_21',
 'movie_24',
 'movie_29',
 'movie_389',
 'movie_410',
 'movie_492',
 'movie_517',
 'movie_56',
 'movie_571',
 'movie_59'},
'user_64': {'movie_111',
 'movie_254',
 'movie_389',
 'movie_440',
 'movie_512',
 'movie_538',
 'movie_595',
 'movie_669',
 'movie_782',
 'movie_9'},
'user_65': {'movie_277',
 'movie_294',
 'movie_309',
 'movie_31',
 'movie_311',
 'movie_389',
 'movie_492',
 'movie_695',
 'movie_702',
 'movie_719'},
'user_66': {'movie_104',
 'movie_115',
 'movie_189',
 'movie_289',
 'movie_420',
 'movie_511',
 'movie_60',
 'movie_652',
 'movie_719',
 'movie_851'},
'user_67': {'movie_101',
 'movie_118',
 'movie_161',
```

 'movie_191',
 'movie_320',
 'movie_347',
 'movie_357',
 'movie_389',
 'movie_458',
 'movie_57'},
'user_68': {'movie_321',
 'movie_356',
 'movie_357',
 'movie_454',
 'movie_475',
 'movie_5',
 'movie_50',
 'movie_571',
 'movie_730',
 'movie_81'},
'user_69': {'movie_182',
 'movie_252',
 'movie_296',
 'movie_422',
 'movie_573',
 'movie_60',
 'movie_628',
 'movie_652',
 'movie_668',
 'movie_76'},
'user_70': {'movie_140',
 'movie_22',
 'movie_471',
 'movie_49',
 'movie_502',
 'movie_552',
 'movie_557',
 'movie_652',
 'movie_862',
 'movie_864'},
'user_71': {'movie_145',
 'movie_305',
 'movie_479',
 'movie_49',
 'movie_5',
 'movie_52',
 'movie_58',
 'movie_873',
 'movie_91',
 'movie_93'},

```
'user_72': {'movie_115',
 'movie_28',
 'movie_449',
 'movie_48',
 'movie_53',
 'movie_57',
 'movie_68',
 'movie_693',
 'movie_712',
 'movie_760'},
'user_73': {'movie_101',
 'movie_12',
 'movie_182',
 'movie_189',
 'movie_22',
 'movie_357',
 'movie_538',
 'movie_622',
 'movie_629',
 'movie_76'},
'user_74': {'movie_101',
 'movie_12',
 'movie_130',
 'movie_189',
 'movie_242',
 'movie_357',
 'movie_49',
 'movie_52',
 'movie_652',
 'movie_719'},
'user_75': {'movie_1032',
 'movie_1110',
 'movie_1130',
 'movie_1196',
 'movie_1251',
 'movie_1258',
 'movie_1569',
 'movie_44',
 'movie_690',
 'movie_693'},
'user_76': {'movie_0',
 'movie_36',
 'movie_469',
 'movie_474',
 'movie_476',
 'movie_492',
 'movie_495',
```

```
 'movie_58',
 'movie_61',
 'movie_973'},
'user_77': {'movie_102',
 'movie_157',
 'movie_209',
 'movie_216',
 'movie_31',
 'movie_49',
 'movie_60',
 'movie_61',
 'movie_635',
 'movie_78'},
'user_78': {'movie_31',
 'movie_331',
 'movie_355',
 'movie_36',
 'movie_361',
 'movie_399',
 'movie_49',
 'movie_939',
 'movie_94',
 'movie_95'},
'user_79': {'movie_139',
 'movie_209',
 'movie_242',
 'movie_355',
 'movie_357',
 'movie_382',
 'movie_52',
 'movie_60',
 'movie_695',
 'movie_862'},
'user_80': {'movie_449',
 'movie_473',
 'movie_475',
 'movie_482',
 'movie_49',
 'movie_56',
 'movie_57',
 'movie_58',
 'movie_62',
 'movie_914'},
'user_81': {'movie_1',
 'movie_31',
 'movie_325',
 'movie_329',
```

```
 'movie_351',
 'movie_355',
 'movie_386',
 'movie_408',
 'movie_431',
 'movie_510'},
'user_82': {'movie_118',
 'movie_251',
 'movie_254',
 'movie_309',
 'movie_31',
 'movie_45',
 'movie_5',
 'movie_719',
 'movie_78',
 'movie_935'},
'user_83': {'movie_101',
 'movie_32',
 'movie_34',
 'movie_47',
 'movie_49',
 'movie_52',
 'movie_569',
 'movie_571',
 'movie_58',
 'movie_91'},
'user_84': {'movie_140',
 'movie_141',
 'movie_175',
 'movie_204',
 'movie_24',
 'movie_364',
 'movie_49',
 'movie_647',
 'movie_652',
 'movie_77'},
'user_85': {'movie_101',
 'movie_118',
 'movie_240',
 'movie_307',
 'movie_333',
 'movie_357',
 'movie_496',
 'movie_5',
 'movie_661',
 'movie_796'},
'user_86': {'movie_140',
```

```
 'movie_24',
 'movie_259',
 'movie_364',
 'movie_400',
 'movie_445',
 'movie_644',
 'movie_647',
 'movie_652',
 'movie_719'},
'user_87': {'movie_102',
 'movie_104',
 'movie_1092',
 'movie_33',
 'movie_456',
 'movie_57',
 'movie_60',
 'movie_629',
 'movie_748',
 'movie_801'},
'user_88': {'movie_1107',
 'movie_1125',
 'movie_1354',
 'movie_1357',
 'movie_24',
 'movie_30',
 'movie_49',
 'movie_822',
 'movie_873',
 'movie_887'},
'user_89': {'movie_1142',
 'movie_125',
 'movie_239',
 'movie_256',
 'movie_260',
 'movie_277',
 'movie_36',
 'movie_5',
 'movie_541',
 'movie_754'},
'user_90': {'movie_29',
 'movie_32',
 'movie_329',
 'movie_337',
 'movie_342',
 'movie_389',
 'movie_49',
 'movie_610',
```

```
 'movie_81',
 'movie_931'},
'user_91': {'movie_23',
 'movie_471',
 'movie_485',
 'movie_498',
 'movie_53',
 'movie_568',
 'movie_571',
 'movie_600',
 'movie_623',
 'movie_629'},
'user_92': {'movie_101',
 'movie_112',
 'movie_118',
 'movie_305',
 'movie_357',
 'movie_367',
 'movie_465',
 'movie_49',
 'movie_50',
 'movie_93'},
'user_93': {'movie_31',
 'movie_44',
 'movie_479',
 'movie_488',
 'movie_49',
 'movie_534',
 'movie_586',
 'movie_592',
 'movie_599',
 'movie_600'},
'user_94': {'movie_258',
 'movie_288',
 'movie_297',
 'movie_329',
 'movie_367',
 'movie_492',
 'movie_5',
 'movie_571',
 'movie_592',
 'movie_94'},
'user_95': {'movie_101',
 'movie_174',
 'movie_239',
 'movie_297',
 'movie_31',
```

```
 'movie_36',
 'movie_465',
 'movie_49',
 'movie_561',
 'movie_920'},
'user_96': {'movie_11',
 'movie_200',
 'movie_21',
 'movie_239',
 'movie_276',
 'movie_31',
 'movie_318',
 'movie_340',
 'movie_358',
 'movie_431'},
'user_97': {'movie_264',
 'movie_289',
 'movie_348',
 'movie_374',
 'movie_49',
 'movie_587',
 'movie_616',
 'movie_68',
 'movie_958',
 'movie_961'},
'user_98': {'movie_17',
 'movie_24',
 'movie_31',
 'movie_33',
 'movie_406',
 'movie_630',
 'movie_64',
 'movie_92',
 'movie_95',
 'movie_96'},
'user_99': {'movie_200',
 'movie_21',
 'movie_289',
 'movie_297',
 'movie_360',
 'movie_423',
 'movie_524',
 'movie_547',
 'movie_6',
 'movie_754'},
'user_100': {'movie_100',
 'movie_1006',
```

```
 'movie_103',
 'movie_34',
 'movie_49',
 'movie_546',
 'movie_561',
 'movie_567',
 'movie_657',
 'movie_66'},
'user_101': {'movie_1008',
 'movie_111',
 'movie_189',
 'movie_239',
 'movie_240',
 'movie_297',
 'movie_367',
 'movie_571',
 'movie_719',
 'movie_88'},
'user_102': {'movie_100',
 'movie_118',
 'movie_238',
 'movie_275',
 'movie_357',
 'movie_5',
 'movie_506',
 'movie_550',
 'movie_58',
 'movie_96'},
'user_103': {'movie_240',
 'movie_247',
 'movie_297',
 'movie_359',
 'movie_36',
 'movie_361',
 'movie_399',
 'movie_403',
 'movie_406',
 'movie_509'},
'user_104': {'movie_118',
 'movie_148',
 'movie_161',
 'movie_240',
 'movie_297',
 'movie_364',
 'movie_498',
 'movie_52',
 'movie_77',
```

```
 'movie_86'},
'user_105': {'movie_103',
 'movie_288',
 'movie_361',
 'movie_367',
 'movie_529',
 'movie_660',
 'movie_699',
 'movie_739',
 'movie_88',
 'movie_93'},
'user_106': {'movie_223',
 'movie_251',
 'movie_289',
 'movie_380',
 'movie_420',
 'movie_568',
 'movie_60',
 'movie_608',
 'movie_815',
 'movie_95'},
'user_107': {'movie_157',
 'movie_161',
 'movie_185',
 'movie_289',
 'movie_355',
 'movie_357',
 'movie_52',
 'movie_695',
 'movie_86',
 'movie_89'},
'user_108': {'movie_145',
 'movie_176',
 'movie_217',
 'movie_256',
 'movie_401',
 'movie_509',
 'movie_632',
 'movie_649',
 'movie_669',
 'movie_873'},
'user_109': {'movie_179',
 'movie_221',
 'movie_260',
 'movie_342',
 'movie_389',
 'movie_47',
```

```
 'movie_492',
 'movie_622',
 'movie_67',
 'movie_950'},
'user_110': {'movie_101',
 'movie_102',
 'movie_260',
 'movie_277',
 'movie_289',
 'movie_496',
 'movie_506',
 'movie_669',
 'movie_725',
 'movie_78'},
'user_111': {'movie_101',
 'movie_1020',
 'movie_1165',
 'movie_49',
 'movie_649',
 'movie_678',
 'movie_734',
 'movie_796',
 'movie_92',
 'movie_93'},
'user_112': {'movie_101',
 'movie_138',
 'movie_174',
 'movie_200',
 'movie_239',
 'movie_240',
 'movie_244',
 'movie_49',
 'movie_725',
 'movie_9'},
'user_113': {'movie_120',
 'movie_161',
 'movie_357',
 'movie_364',
 'movie_392',
 'movie_52',
 'movie_652',
 'movie_719',
 'movie_77',
 'movie_95'},
'user_114': {'movie_0',
 'movie_31',
 'movie_315',
```

```
 'movie_316',
 'movie_318',
 'movie_319',
 'movie_387',
 'movie_389',
 'movie_465',
 'movie_473'},
'user_115': {'movie_1',
 'movie_115',
 'movie_189',
 'movie_200',
 'movie_49',
 'movie_552',
 'movie_58',
 'movie_60',
 'movie_695',
 'movie_719'},
'user_116': {'movie_1196',
 'movie_1277',
 'movie_22',
 'movie_24',
 'movie_26',
 'movie_31',
 'movie_56',
 'movie_60',
 'movie_613',
 'movie_66'},
'user_117': {'movie_1167',
 'movie_29',
 'movie_31',
 'movie_348',
 'movie_364',
 'movie_377',
 'movie_60',
 'movie_853',
 'movie_862',
 'movie_864'},
'user_118': {'movie_24',
 'movie_38',
 'movie_49',
 'movie_52',
 'movie_53',
 'movie_648',
 'movie_68',
 'movie_690',
 'movie_71',
 'movie_719'},
```

'user_119': {'movie_161',
 'movie_209',
 'movie_22',
 'movie_32',
 'movie_34',
 'movie_357',
 'movie_52',
 'movie_57',
 'movie_61',
 'movie_77'},
'user_120': {'movie_1076',
 'movie_1084',
 'movie_177',
 'movie_357',
 'movie_380',
 'movie_420',
 'movie_52',
 'movie_552',
 'movie_671',
 'movie_755'},
'user_121': {'movie_1077',
 'movie_166',
 'movie_189',
 'movie_221',
 'movie_239',
 'movie_240',
 'movie_342',
 'movie_359',
 'movie_58',
 'movie_996'},
'user_122': {'movie_1',
 'movie_100',
 'movie_102',
 'movie_395',
 'movie_405',
 'movie_49',
 'movie_500',
 'movie_529',
 'movie_53',
 'movie_882'},
'user_123': {'movie_1069',
 'movie_302',
 'movie_333',
 'movie_387',
 'movie_432',
 'movie_487',
 'movie_657',

```
 'movie_66',
 'movie_68',
 'movie_977'},
'user_124': {'movie_133',
 'movie_179',
 'movie_191',
 'movie_307',
 'movie_31',
 'movie_325',
 'movie_355',
 'movie_49',
 'movie_695',
 'movie_86'},
'user_125': {'movie_118',
 'movie_191',
 'movie_195',
 'movie_225',
 'movie_34',
 'movie_423',
 'movie_50',
 'movie_612',
 'movie_781',
 'movie_882'},
'user_126': {'movie_22',
 'movie_260',
 'movie_261',
 'movie_291',
 'movie_303',
 'movie_321',
 'movie_33',
 'movie_34',
 'movie_52',
 'movie_56'},
'user_127': {'movie_101',
 'movie_136',
 'movie_177',
 'movie_408',
 'movie_528',
 'movie_53',
 'movie_543',
 'movie_552',
 'movie_6',
 'movie_864'},
'user_128': {'movie_101',
 'movie_102',
 'movie_108',
 'movie_113',
```

```
 'movie_156',
 'movie_240',
 'movie_359',
 'movie_496',
 'movie_677',
 'movie_735'},
'user_129': {'movie_1108',
 'movie_1397',
 'movie_21',
 'movie_24',
 'movie_47',
 'movie_5',
 'movie_6',
 'movie_8',
 'movie_856',
 'movie_877'},
'user_130': {'movie_1077',
 'movie_139',
 'movie_175',
 'movie_357',
 'movie_392',
 'movie_43',
 'movie_49',
 'movie_52',
 'movie_643',
 'movie_656'},
'user_131': {'movie_1',
 'movie_1049',
 'movie_12',
 'movie_31',
 'movie_389',
 'movie_408',
 'movie_431',
 'movie_456',
 'movie_480',
 'movie_66'},
'user_132': {'movie_112',
 'movie_200',
 'movie_29',
 'movie_31',
 'movie_320',
 'movie_36',
 'movie_52',
 'movie_57',
 'movie_58',
 'movie_608'},
'user_133': {'movie_230',
```

```
 'movie_24',
 'movie_269',
 'movie_289',
 'movie_355',
 'movie_403',
 'movie_52',
 'movie_58',
 'movie_60',
 'movie_73'},
'user_134': {'movie_1',
 'movie_1120',
 'movie_1166',
 'movie_214',
 'movie_289',
 'movie_60',
 'movie_666',
 'movie_807',
 'movie_824',
 'movie_9'},
'user_135': {'movie_1023',
 'movie_1036',
 'movie_1261',
 'movie_1466',
 'movie_206',
 'movie_288',
 'movie_357',
 'movie_824',
 'movie_838',
 'movie_995'},
'user_136': {'movie_101',
 'movie_153',
 'movie_174',
 'movie_182',
 'movie_200',
 'movie_297',
 'movie_433',
 'movie_538',
 'movie_57',
 'movie_78'},
'user_137': {'movie_38',
 'movie_49',
 'movie_550',
 'movie_581',
 'movie_733',
 'movie_753',
 'movie_781',
 'movie_862',
```

```
 'movie_873',
 'movie_982'},
'user_138': {'movie_1016',
 'movie_1024',
 'movie_1090',
 'movie_1134',
 'movie_1162',
 'movie_31',
 'movie_46',
 'movie_580',
 'movie_852',
 'movie_931'},
'user_139': {'movie_299',
 'movie_309',
 'movie_337',
 'movie_40',
 'movie_423',
 'movie_433',
 'movie_454',
 'movie_475',
 'movie_491',
 'movie_502'},
'user_140': {'movie_216',
 'movie_239',
 'movie_288',
 'movie_297',
 'movie_317',
 'movie_492',
 'movie_497',
 'movie_512',
 'movie_592',
 'movie_595'},
'user_141': {'movie_157',
 'movie_189',
 'movie_357',
 'movie_393',
 'movie_404',
 'movie_406',
 'movie_487',
 'movie_49',
 'movie_502',
 'movie_58'},
'user_142': {'movie_1444',
 'movie_261',
 'movie_395',
 'movie_609',
 'movie_652',
```

```
 'movie_719',
 'movie_72',
 'movie_839',
 'movie_95',
 'movie_98'},
'user_143': {'movie_195',
 'movie_198',
 'movie_253',
 'movie_277',
 'movie_367',
 'movie_474',
 'movie_479',
 'movie_52',
 'movie_538',
 'movie_57'},
'user_144': {'movie_31',
 'movie_355',
 'movie_36',
 'movie_423',
 'movie_49',
 'movie_58',
 'movie_612',
 'movie_623',
 'movie_657',
 'movie_72'},
'user_145': {'movie_130',
 'movie_189',
 'movie_191',
 'movie_275',
 'movie_347',
 'movie_408',
 'movie_485',
 'movie_49',
 'movie_647',
 'movie_93'},
'user_146': {'movie_1035',
 'movie_1559',
 'movie_157',
 'movie_228',
 'movie_243',
 'movie_357',
 'movie_387',
 'movie_585',
 'movie_809',
 'movie_860'},
'user_147': {'movie_150',
 'movie_24',
```

```
 'movie_275',
 'movie_311',
 'movie_357',
 'movie_471',
 'movie_50',
 'movie_52',
 'movie_57',
 'movie_95'},
'user_148': {'movie_100',
 'movie_1108',
 'movie_157',
 'movie_244',
 'movie_315',
 'movie_479',
 'movie_491',
 'movie_492',
 'movie_583',
 'movie_92'},
'user_149': {'movie_256',
 'movie_297',
 'movie_31',
 'movie_311',
 'movie_361',
 'movie_367',
 'movie_592',
 'movie_719',
 'movie_808',
 'movie_93'},
'user_150': {'movie_108',
 'movie_110',
 'movie_118',
 'movie_133',
 'movie_137',
 'movie_273',
 'movie_289',
 'movie_50',
 'movie_56',
 'movie_730'},
'user_151': {'movie_105',
 'movie_112',
 'movie_23',
 'movie_338',
 'movie_354',
 'movie_359',
 'movie_406',
 'movie_408',
 'movie_629',
```

```
 'movie_89'},
'user_152': {'movie_1023',
 'movie_104',
 'movie_130',
 'movie_157',
 'movie_269',
 'movie_280',
 'movie_321',
 'movie_355',
 'movie_404',
 'movie_60'},
'user_153': {'movie_1022',
 'movie_24',
 'movie_31',
 'movie_411',
 'movie_479',
 'movie_49',
 'movie_491',
 'movie_538',
 'movie_573',
 'movie_665'},
'user_154': {'movie_1',
 'movie_24',
 'movie_28',
 'movie_31',
 'movie_34',
 'movie_57',
 'movie_58',
 'movie_59',
 'movie_862',
 'movie_942'},
'user_155': {'movie_144',
 'movie_157',
 'movie_252',
 'movie_280',
 'movie_60',
 'movie_652',
 'movie_719',
 'movie_891',
 'movie_959',
 'movie_98'},
'user_156': {'movie_101',
 'movie_217',
 'movie_288',
 'movie_294',
 'movie_329',
 'movie_356',
```

```
 'movie_357',
 'movie_5',
 'movie_561',
 'movie_78'},
'user_157': {'movie_217',
 'movie_221',
 'movie_241',
 'movie_256',
 'movie_275',
 'movie_321',
 'movie_329',
 'movie_354',
 'movie_473',
 'movie_492'},
'user_158': {'movie_1223',
 'movie_29',
 'movie_520',
 'movie_534',
 'movie_558',
 'movie_590',
 'movie_60',
 'movie_600',
 'movie_61',
 'movie_64'},
'user_159': {'movie_166',
 'movie_203',
 'movie_254',
 'movie_299',
 'movie_389',
 'movie_449',
 'movie_465',
 'movie_629',
 'movie_66',
 'movie_693'},
'user_160': {'movie_117',
 'movie_191',
 'movie_245',
 'movie_265',
 'movie_420',
 'movie_476',
 'movie_48',
 'movie_652',
 'movie_719',
 'movie_860'},
'user_161': {'movie_1034',
 'movie_140',
 'movie_25',
```

```
 'movie_364',
 'movie_403',
 'movie_445',
 'movie_52',
 'movie_552',
 'movie_613',
 'movie_77'},
'user_162': {'movie_1063',
 'movie_108',
 'movie_440',
 'movie_502',
 'movie_567',
 'movie_58',
 'movie_600',
 'movie_89',
 'movie_91',
 'movie_92'},
'user_163': {'movie_115',
 'movie_130',
 'movie_31',
 'movie_325',
 'movie_380',
 'movie_4',
 'movie_491',
 'movie_630',
 'movie_681',
 'movie_950'},
'user_164': {'movie_115',
 'movie_130',
 'movie_157',
 'movie_245',
 'movie_325',
 'movie_357',
 'movie_380',
 'movie_491',
 'movie_630',
 'movie_671'},
'user_165': {'movie_1020',
 'movie_103',
 'movie_112',
 'movie_311',
 'movie_357',
 'movie_418',
 'movie_50',
 'movie_57',
 'movie_78',
 'movie_99'},
```

```
'user_166': {'movie_103',
 'movie_1576',
 'movie_263',
 'movie_273',
 'movie_287',
 'movie_297',
 'movie_30',
 'movie_357',
 'movie_371',
 'movie_60'},
'user_167': {'movie_1',
 'movie_115',
 'movie_355',
 'movie_357',
 'movie_49',
 'movie_491',
 'movie_644',
 'movie_68',
 'movie_695',
 'movie_719'},
'user_168': {'movie_0',
 'movie_1',
 'movie_130',
 'movie_289',
 'movie_404',
 'movie_491',
 'movie_60',
 'movie_666',
 'movie_719',
 'movie_881'},
'user_169': {'movie_102',
 'movie_118',
 'movie_170',
 'movie_200',
 'movie_431',
 'movie_49',
 'movie_502',
 'movie_547',
 'movie_561',
 'movie_57'},
'user_170': {'movie_1',
 'movie_115',
 'movie_130',
 'movie_189',
 'movie_289',
 'movie_325',
 'movie_608',
```

```
 'movie_630',
 'movie_751',
 'movie_842'},
'user_171': {'movie_1',
 'movie_157',
 'movie_200',
 'movie_239',
 'movie_40',
 'movie_408',
 'movie_511',
 'movie_652',
 'movie_719',
 'movie_98'},
'user_172': {'movie_108',
 'movie_139',
 'movie_216',
 'movie_260',
 'movie_280',
 'movie_311',
 'movie_357',
 'movie_4',
 'movie_49',
 'movie_529'},
'user_173': {'movie_118',
 'movie_22',
 'movie_288',
 'movie_297',
 'movie_31',
 'movie_356',
 'movie_357',
 'movie_60',
 'movie_82',
 'movie_94'},
'user_174': {'movie_161',
 'movie_357',
 'movie_371',
 'movie_52',
 'movie_534',
 'movie_60',
 'movie_600',
 'movie_73',
 'movie_860',
 'movie_862'},
'user_175': {'movie_1047',
 'movie_113',
 'movie_1195',
 'movie_24',
```

```
 'movie_29',
 'movie_44',
 'movie_49',
 'movie_692',
 'movie_806',
 'movie_860'},
'user_176': {'movie_102',
 'movie_112',
 'movie_311',
 'movie_347',
 'movie_357',
 'movie_458',
 'movie_52',
 'movie_58',
 'movie_6',
 'movie_62'},
'user_177': {'movie_17',
 'movie_175',
 'movie_24',
 'movie_309',
 'movie_403',
 'movie_408',
 'movie_68',
 'movie_719',
 'movie_860',
 'movie_99'},
'user_178': {'movie_31',
 'movie_411',
 'movie_431',
 'movie_472',
 'movie_518',
 'movie_527',
 'movie_53',
 'movie_567',
 'movie_693',
 'movie_83'},
'user_179': {'movie_200',
 'movie_216',
 'movie_289',
 'movie_31',
 'movie_379',
 'movie_472',
 'movie_475',
 'movie_50',
 'movie_57',
 'movie_623'},
'user_180': {'movie_26',
```

```
 'movie_34',
 'movie_49',
 'movie_524',
 'movie_538',
 'movie_540',
 'movie_56',
 'movie_630',
 'movie_657',
 'movie_725'},
'user_181': {'movie_1113',
 'movie_230',
 'movie_253',
 'movie_256',
 'movie_437',
 'movie_474',
 'movie_60',
 'movie_758',
 'movie_87',
 'movie_900'},
'user_182': {'movie_101',
 'movie_112',
 'movie_188',
 'movie_189',
 'movie_297',
 'movie_34',
 'movie_491',
 'movie_492',
 'movie_93',
 'movie_99'},
'user_183': {'movie_1111',
 'movie_118',
 'movie_161',
 'movie_217',
 'movie_311',
 'movie_357',
 'movie_49',
 'movie_492',
 'movie_5',
 'movie_52'},
'user_184': {'movie_1155',
 'movie_241',
 'movie_242',
 'movie_333',
 'movie_34',
 'movie_359',
 'movie_36',
 'movie_389',
```

```
 'movie_390',
 'movie_5'},
'user_185': {'movie_12',
 'movie_150',
 'movie_357',
 'movie_367',
 'movie_518',
 'movie_52',
 'movie_614',
 'movie_68',
 'movie_719',
 'movie_87'},
'user_186': {'movie_172',
 'movie_20',
 'movie_244',
 'movie_502',
 'movie_53',
 'movie_557',
 'movie_57',
 'movie_643',
 'movie_66',
 'movie_783'},
'user_187': {'movie_101',
 'movie_102',
 'movie_112',
 'movie_114',
 'movie_136',
 'movie_391',
 'movie_403',
 'movie_52',
 'movie_656',
 'movie_97'},
'user_188': {'movie_333',
 'movie_357',
 'movie_429',
 'movie_45',
 'movie_49',
 'movie_497',
 'movie_52',
 'movie_695',
 'movie_719',
 'movie_733'},
'user_189': {'movie_0',
 'movie_157',
 'movie_289',
 'movie_325',
 'movie_40',
```

```
 'movie_404',
 'movie_491',
 'movie_719',
 'movie_751',
 'movie_98'},
'user_190': {'movie_1',
 'movie_103',
 'movie_1167',
 'movie_235',
 'movie_240',
 'movie_357',
 'movie_371',
 'movie_408',
 'movie_52',
 'movie_719'},
'user_191': {'movie_1107',
 'movie_112',
 'movie_113',
 'movie_1401',
 'movie_216',
 'movie_217',
 'movie_235',
 'movie_31',
 'movie_361',
 'movie_389'},
'user_192': {'movie_139',
 'movie_189',
 'movie_214',
 'movie_216',
 'movie_229',
 'movie_30',
 'movie_571',
 'movie_58',
 'movie_719',
 'movie_982'},
'user_193': {'movie_217',
 'movie_239',
 'movie_240',
 'movie_273',
 'movie_359',
 'movie_405',
 'movie_492',
 'movie_574',
 'movie_660',
 'movie_81'},
'user_194': {'movie_161',
 'movie_170',
```

```
 'movie_200',
 'movie_347',
 'movie_351',
 'movie_371',
 'movie_383',
 'movie_49',
 'movie_613',
 'movie_734'},
'user_195': {'movie_1343',
 'movie_174',
 'movie_200',
 'movie_295',
 'movie_34',
 'movie_357',
 'movie_5',
 'movie_538',
 'movie_67',
 'movie_89'},
'user_196': {'movie_117',
 'movie_120',
 'movie_23',
 'movie_280',
 'movie_298',
 'movie_381',
 'movie_403',
 'movie_52',
 'movie_652',
 'movie_890'},
'user_197': {'movie_1167',
 'movie_269',
 'movie_40',
 'movie_48',
 'movie_484',
 'movie_51',
 'movie_511',
 'movie_60',
 'movie_652',
 'movie_72'},
'user_198': {'movie_101',
 'movie_125',
 'movie_161',
 'movie_174',
 'movie_185',
 'movie_340',
 'movie_357',
 'movie_387',
 'movie_52',
```

 'movie_57'},
'user_199': {'movie_0',
 'movie_1',
 'movie_130',
 'movie_174',
 'movie_243',
 'movie_289',
 'movie_491',
 'movie_58',
 'movie_66',
 'movie_744'},
'user_200': {'movie_10',
 'movie_101',
 'movie_189',
 'movie_198',
 'movie_24',
 'movie_357',
 'movie_432',
 'movie_49',
 'movie_695',
 'movie_719'},
'user_201': {'movie_0',
 'movie_1',
 'movie_115',
 'movie_133',
 'movie_289',
 'movie_49',
 'movie_491',
 'movie_502',
 'movie_512',
 'movie_873'},
'user_202': {'movie_140',
 'movie_544',
 'movie_56',
 'movie_569',
 'movie_57',
 'movie_58',
 'movie_6',
 'movie_683',
 'movie_77',
 'movie_95'},
'user_203': {'movie_1103',
 'movie_1181',
 'movie_1229',
 'movie_1231',
 'movie_668',
 'movie_713',

```
 'movie_719',
 'movie_930',
 'movie_98',
 'movie_999'},
'user_204': {'movie_120',
 'movie_136',
 'movie_175',
 'movie_350',
 'movie_387',
 'movie_403',
 'movie_52',
 'movie_652',
 'movie_68',
 'movie_762'},
'user_205': {'movie_113',
 'movie_20',
 'movie_21',
 'movie_26',
 'movie_31',
 'movie_57',
 'movie_60',
 'movie_70',
 'movie_703',
 'movie_971'},
'user_206': {'movie_1532',
 'movie_239',
 'movie_240',
 'movie_273',
 'movie_31',
 'movie_4',
 'movie_652',
 'movie_66',
 'movie_842',
 'movie_855'},
'user_207': {'movie_143',
 'movie_33',
 'movie_389',
 'movie_403',
 'movie_406',
 'movie_408',
 'movie_468',
 'movie_485',
 'movie_56',
 'movie_57'},
'user_208': {'movie_12',
 'movie_146',
 'movie_156',
```

```
 'movie_216',
 'movie_231',
 'movie_54',
 'movie_698',
 'movie_801',
 'movie_82',
 'movie_97'},
'user_209': {'movie_139',
 'movie_157',
 'movie_188',
 'movie_209',
 'movie_289',
 'movie_331',
 'movie_380',
 'movie_49',
 'movie_863',
 'movie_873'},
'user_210': {'movie_103',
 'movie_12',
 'movie_1206',
 'movie_141',
 'movie_267',
 'movie_269',
 'movie_322',
 'movie_355',
 'movie_482',
 'movie_99'},
'user_211': {'movie_1190',
 'movie_12',
 'movie_157',
 'movie_196',
 'movie_278',
 'movie_340',
 'movie_357',
 'movie_52',
 'movie_572',
 'movie_652'},
'user_212': {'movie_1',
 'movie_1257',
 'movie_139',
 'movie_157',
 'movie_252',
 'movie_357',
 'movie_43',
 'movie_491',
 'movie_60',
 'movie_95'},
```

```
'user_213': {'movie_10',
 'movie_177',
 'movie_360',
 'movie_404',
 'movie_514',
 'movie_534',
 'movie_642',
 'movie_656',
 'movie_77',
 'movie_851'},
'user_214': {'movie_1',
 'movie_1026',
 'movie_29',
 'movie_38',
 'movie_44',
 'movie_49',
 'movie_63',
 'movie_66',
 'movie_806',
 'movie_819'},
'user_215': {'movie_26',
 'movie_321',
 'movie_347',
 'movie_357',
 'movie_367',
 'movie_419',
 'movie_50',
 'movie_622',
 'movie_734',
 'movie_96'},
'user_216': {'movie_130',
 'movie_139',
 'movie_157',
 'movie_325',
 'movie_4',
 'movie_43',
 'movie_511',
 'movie_863',
 'movie_950',
 'movie_960'},
'user_217': {'movie_1',
 'movie_1098',
 'movie_130',
 'movie_139',
 'movie_355',
 'movie_357',
 'movie_49',
```

```
 'movie_491',
 'movie_66',
 'movie_751'},
'user_218': {'movie_0',
 'movie_1114',
 'movie_133',
 'movie_157',
 'movie_180',
 'movie_217',
 'movie_277',
 'movie_297',
 'movie_31',
 'movie_506'},
'user_219': {'movie_0',
 'movie_101',
 'movie_1069',
 'movie_1559',
 'movie_174',
 'movie_36',
 'movie_380',
 'movie_538',
 'movie_66',
 'movie_665'},
'user_220': {'movie_0',
 'movie_1',
 'movie_144',
 'movie_243',
 'movie_289',
 'movie_346',
 'movie_497',
 'movie_72',
 'movie_860',
 'movie_98'},
'user_221': {'movie_108',
 'movie_189',
 'movie_251',
 'movie_260',
 'movie_273',
 'movie_361',
 'movie_471',
 'movie_491',
 'movie_512',
 'movie_9'},
'user_222': {'movie_139',
 'movie_161',
 'movie_189',
 'movie_216',
```

```
 'movie_30',
 'movie_325',
 'movie_355',
 'movie_357',
 'movie_52',
 'movie_695'},
'user_223': {'movie_29',
 'movie_302',
 'movie_31',
 'movie_342',
 'movie_357',
 'movie_365',
 'movie_52',
 'movie_568',
 'movie_647',
 'movie_68'},
'user_224': {'movie_200',
 'movie_229',
 'movie_320',
 'movie_364',
 'movie_408',
 'movie_438',
 'movie_440',
 'movie_476',
 'movie_56',
 'movie_571'},
'user_225': {'movie_101',
 'movie_161',
 'movie_240',
 'movie_254',
 'movie_256',
 'movie_31',
 'movie_359',
 'movie_496',
 'movie_52',
 'movie_754'},
'user_226': {'movie_125',
 'movie_145',
 'movie_161',
 'movie_561',
 'movie_57',
 'movie_647',
 'movie_68',
 'movie_735',
 'movie_76',
 'movie_99'},
'user_227': {'movie_12',
```

```
 'movie_157',
 'movie_161',
 'movie_200',
 'movie_357',
 'movie_50',
 'movie_52',
 'movie_619',
 'movie_629',
 'movie_860'},
'user_228': {'movie_17',
 'movie_273',
 'movie_309',
 'movie_33',
 'movie_34',
 'movie_374',
 'movie_53',
 'movie_550',
 'movie_622',
 'movie_99'},
'user_229': {'movie_0',
 'movie_130',
 'movie_357',
 'movie_49',
 'movie_512',
 'movie_522',
 'movie_600',
 'movie_630',
 'movie_719',
 'movie_807'},
'user_230': {'movie_0',
 'movie_115',
 'movie_130',
 'movie_175',
 'movie_311',
 'movie_325',
 'movie_360',
 'movie_630',
 'movie_68',
 'movie_860'},
'user_231': {'movie_1195',
 'movie_140',
 'movie_217',
 'movie_235',
 'movie_307',
 'movie_31',
 'movie_311',
 'movie_351',
```

```
 'movie_58',
 'movie_612'},
'user_232': {'movie_102',
 'movie_12',
 'movie_124',
 'movie_133',
 'movie_173',
 'movie_181',
 'movie_24',
 'movie_302',
 'movie_530',
 'movie_89'},
'user_233': {'movie_140',
 'movie_141',
 'movie_196',
 'movie_209',
 'movie_278',
 'movie_280',
 'movie_387',
 'movie_445',
 'movie_51',
 'movie_8'},
'user_234': {'movie_130',
 'movie_1436',
 'movie_189',
 'movie_273',
 'movie_31',
 'movie_325',
 'movie_466',
 'movie_608',
 'movie_66',
 'movie_955'},
'user_235': {'movie_139',
 'movie_174',
 'movie_189',
 'movie_200',
 'movie_216',
 'movie_239',
 'movie_289',
 'movie_436',
 'movie_49',
 'movie_58'},
'user_236': {'movie_118',
 'movie_185',
 'movie_189',
 'movie_273',
 'movie_359',
```

```
 'movie_361',
 'movie_50',
 'movie_673',
 'movie_78',
 'movie_94'},
'user_237': {'movie_101',
 'movie_208',
 'movie_216',
 'movie_239',
 'movie_278',
 'movie_296',
 'movie_357',
 'movie_456',
 'movie_758',
 'movie_82'},
'user_238': {'movie_17',
 'movie_23',
 'movie_387',
 'movie_434',
 'movie_445',
 'movie_446',
 'movie_531',
 'movie_552',
 'movie_719',
 'movie_86'},
'user_239': {'movie_100',
 'movie_118',
 'movie_260',
 'movie_29',
 'movie_337',
 'movie_49',
 'movie_52',
 'movie_56',
 'movie_629',
 'movie_8'},
'user_240': {'movie_133',
 'movie_148',
 'movie_189',
 'movie_209',
 'movie_289',
 'movie_502',
 'movie_568',
 'movie_58',
 'movie_873',
 'movie_89'},
'user_241': {'movie_1',
 'movie_130',
```

```
 'movie_200',
 'movie_346',
 'movie_357',
 'movie_49',
 'movie_502',
 'movie_538',
 'movie_751',
 'movie_863'},
'user_242': {'movie_136',
 'movie_216',
 'movie_269',
 'movie_29',
 'movie_31',
 'movie_382',
 'movie_523',
 'movie_594',
 'movie_61',
 'movie_614'},
'user_243': {'movie_103',
 'movie_449',
 'movie_491',
 'movie_599',
 'movie_68',
 'movie_719',
 'movie_851',
 'movie_86',
 'movie_89',
 'movie_9'},
'user_244': {'movie_1',
 'movie_133',
 'movie_139',
 'movie_174',
 'movie_251',
 'movie_289',
 'movie_49',
 'movie_512',
 'movie_568',
 'movie_851'},
'user_245': {'movie_1',
 'movie_130',
 'movie_204',
 'movie_209',
 'movie_297',
 'movie_357',
 'movie_554',
 'movie_60',
 'movie_719',
```

```
 'movie_77'},
'user_246': {'movie_113',
 'movie_185',
 'movie_31',
 'movie_311',
 'movie_329',
 'movie_389',
 'movie_399',
 'movie_431',
 'movie_623',
 'movie_953'},
'user_247': {'movie_103',
 'movie_148',
 'movie_209',
 'movie_355',
 'movie_371',
 'movie_387',
 'movie_49',
 'movie_491',
 'movie_695',
 'movie_95'},
'user_248': {'movie_1151',
 'movie_258',
 'movie_273',
 'movie_326',
 'movie_342',
 'movie_357',
 'movie_49',
 'movie_50',
 'movie_611',
 'movie_750'},
'user_249': {'movie_1',
 'movie_101',
 'movie_189',
 'movie_200',
 'movie_239',
 'movie_275',
 'movie_49',
 'movie_68',
 'movie_719',
 'movie_862'},
'user_250': {'movie_102',
 'movie_112',
 'movie_209',
 'movie_22',
 'movie_31',
 'movie_367',
```

  'movie_368',
  'movie_413',
  'movie_583',
  'movie_638'},
 'user_251': {'movie_1',
  'movie_1120',
  'movie_200',
  'movie_289',
  'movie_357',
  'movie_4',
  'movie_491',
  'movie_719',
  'movie_751',
  'movie_842'},
 'user_252': {'movie_12',
  'movie_130',
  'movie_139',
  'movie_30',
  'movie_346',
  'movie_427',
  'movie_497',
  'movie_512',
  'movie_534',
  'movie_842'},
 'user_253': {'movie_100',
  'movie_200',
  'movie_216',
  'movie_239',
  'movie_240',
  'movie_254',
  'movie_320',
  'movie_34',
  'movie_540',
  'movie_751'},
 'user_254': {'movie_256',
  'movie_258',
  'movie_260',
  'movie_29',
  'movie_36',
  'movie_423',
  'movie_433',
  'movie_436',
  'movie_729',
  'movie_733'},
 'user_255': {'movie_135',
  'movie_161',
  'movie_174',

```
 'movie_189',
 'movie_239',
 'movie_244',
 'movie_31',
 'movie_357',
 'movie_52',
 'movie_99'},
'user_256': {'movie_108',
 'movie_156',
 'movie_157',
 'movie_201',
 'movie_216',
 'movie_239',
 'movie_240',
 'movie_31',
 'movie_356',
 'movie_6'},
'user_257': {'movie_1',
 'movie_191',
 'movie_200',
 'movie_289',
 'movie_357',
 'movie_491',
 'movie_60',
 'movie_600',
 'movie_652',
 'movie_719'},
'user_258': {'movie_184',
 'movie_25',
 'movie_280',
 'movie_305',
 'movie_331',
 'movie_403',
 'movie_60',
 'movie_608',
 'movie_734',
 'movie_860'},
'user_259': {'movie_0',
 'movie_1',
 'movie_130',
 'movie_133',
 'movie_180',
 'movie_289',
 'movie_355',
 'movie_491',
 'movie_512',
 'movie_665'},
```

```
'user_260': {'movie_102',
 'movie_112',
 'movie_12',
 'movie_157',
 'movie_23',
 'movie_319',
 'movie_403',
 'movie_491',
 'movie_592',
 'movie_881'},
'user_261': {'movie_1',
 'movie_141',
 'movie_17',
 'movie_364',
 'movie_427',
 'movie_452',
 'movie_501',
 'movie_502',
 'movie_512',
 'movie_970'},
'user_262': {'movie_1',
 'movie_1120',
 'movie_130',
 'movie_157',
 'movie_280',
 'movie_325',
 'movie_4',
 'movie_511',
 'movie_60',
 'movie_719'},
'user_263': {'movie_102',
 'movie_1059',
 'movie_1401',
 'movie_216',
 'movie_289',
 'movie_371',
 'movie_49',
 'movie_541',
 'movie_599',
 'movie_94'},
'user_264': {'movie_1034',
 'movie_26',
 'movie_27',
 'movie_56',
 'movie_593',
 'movie_60',
 'movie_647',
```

```
 'movie_652',
 'movie_77',
 'movie_935'},
'user_265': {'movie_133',
 'movie_189',
 'movie_200',
 'movie_289',
 'movie_400',
 'movie_49',
 'movie_665',
 'movie_695',
 'movie_697',
 'movie_750'},
'user_266': {'movie_130',
 'movie_157',
 'movie_280',
 'movie_397',
 'movie_511',
 'movie_60',
 'movie_630',
 'movie_652',
 'movie_719',
 'movie_881'},
'user_267': {'movie_1042',
 'movie_157',
 'movie_171',
 'movie_243',
 'movie_43',
 'movie_497',
 'movie_652',
 'movie_891',
 'movie_95',
 'movie_997'},
'user_268': {'movie_153',
 'movie_17',
 'movie_269',
 'movie_348',
 'movie_51',
 'movie_614',
 'movie_652',
 'movie_77',
 'movie_89',
 'movie_95'},
'user_269': {'movie_1116',
 'movie_184',
 'movie_200',
 'movie_214',
```

```
 'movie_43',
 'movie_52',
 'movie_819',
 'movie_891',
 'movie_959',
 'movie_98'},
'user_270': {'movie_1',
 'movie_148',
 'movie_174',
 'movie_209',
 'movie_446',
 'movie_49',
 'movie_501',
 'movie_534',
 'movie_58',
 'movie_89'},
'user_271': {'movie_139',
 'movie_24',
 'movie_267',
 'movie_355',
 'movie_49',
 'movie_502',
 'movie_695',
 'movie_751',
 'movie_870',
 'movie_873'},
'user_272': {'movie_139',
 'movie_200',
 'movie_273',
 'movie_30',
 'movie_315',
 'movie_360',
 'movie_547',
 'movie_647',
 'movie_754',
 'movie_92'},
'user_273': {'movie_1',
 'movie_10',
 'movie_133',
 'movie_302',
 'movie_40',
 'movie_522',
 'movie_66',
 'movie_661',
 'movie_665',
 'movie_89'},
'user_274': {'movie_175',
```

```
 'movie_231',
 'movie_280',
 'movie_339',
 'movie_408',
 'movie_471',
 'movie_48',
 'movie_519',
 'movie_652',
 'movie_719'},
'user_275': {'movie_115',
 'movie_117',
 'movie_133',
 'movie_289',
 'movie_404',
 'movie_42',
 'movie_420',
 'movie_666',
 'movie_680',
 'movie_681'},
'user_276': {'movie_1',
 'movie_130',
 'movie_157',
 'movie_239',
 'movie_357',
 'movie_43',
 'movie_491',
 'movie_497',
 'movie_719',
 'movie_960'},
'user_277': {'movie_0',
 'movie_1184',
 'movie_1473',
 'movie_420',
 'movie_48',
 'movie_49',
 'movie_497',
 'movie_585',
 'movie_815',
 'movie_891'},
'user_278': {'movie_10',
 'movie_1110',
 'movie_133',
 'movie_217',
 'movie_240',
 'movie_299',
 'movie_355',
 'movie_367',
```

```
 'movie_546',
 'movie_668'},
'user_279': {'movie_100',
 'movie_112',
 'movie_161',
 'movie_22',
 'movie_357',
 'movie_506',
 'movie_52',
 'movie_6',
 'movie_693',
 'movie_719'},
'user_280': {'movie_0',
 'movie_1',
 'movie_1243',
 'movie_200',
 'movie_243',
 'movie_420',
 'movie_466',
 'movie_66',
 'movie_751',
 'movie_969'},
'user_281': {'movie_1',
 'movie_351',
 'movie_392',
 'movie_399',
 'movie_401',
 'movie_491',
 'movie_496',
 'movie_512',
 'movie_534',
 'movie_541'},
'user_282': {'movie_137',
 'movie_238',
 'movie_241',
 'movie_288',
 'movie_466',
 'movie_496',
 'movie_5',
 'movie_640',
 'movie_863',
 'movie_936'},
'user_283': {'movie_1069',
 'movie_31',
 'movie_316',
 'movie_333',
 'movie_418',
```

```
 'movie_482',
 'movie_508',
 'movie_567',
 'movie_600',
 'movie_629'},
'user_284': {'movie_216',
 'movie_305',
 'movie_355',
 'movie_357',
 'movie_387',
 'movie_404',
 'movie_485',
 'movie_52',
 'movie_613',
 'movie_657'},
'user_285': {'movie_10',
 'movie_307',
 'movie_325',
 'movie_342',
 'movie_366',
 'movie_49',
 'movie_77',
 'movie_797',
 'movie_86',
 'movie_945'},
'user_286': {'movie_101',
 'movie_118',
 'movie_189',
 'movie_191',
 'movie_289',
 'movie_333',
 'movie_49',
 'movie_622',
 'movie_672',
 'movie_719'},
'user_287': {'movie_1034',
 'movie_136',
 'movie_157',
 'movie_200',
 'movie_239',
 'movie_24',
 'movie_289',
 'movie_552',
 'movie_58',
 'movie_647'},
'user_288': {'movie_1078',
 'movie_1184',
```

```
 'movie_1398',
 'movie_175',
 'movie_280',
 'movie_289',
 'movie_346',
 'movie_719',
 'movie_72',
 'movie_863'},
'user_289': {'movie_140',
 'movie_158',
 'movie_209',
 'movie_364',
 'movie_369',
 'movie_49',
 'movie_613',
 'movie_719',
 'movie_87',
 'movie_97'},
'user_290': {'movie_0',
 'movie_189',
 'movie_24',
 'movie_251',
 'movie_39',
 'movie_427',
 'movie_45',
 'movie_512',
 'movie_522',
 'movie_786'},
'user_291': {'movie_130',
 'movie_240',
 'movie_267',
 'movie_289',
 'movie_335',
 'movie_355',
 'movie_552',
 'movie_652',
 'movie_731',
 'movie_751'},
'user_292': {'movie_1229',
 'movie_144',
 'movie_1493',
 'movie_289',
 'movie_554',
 'movie_56',
 'movie_680',
 'movie_700',
 'movie_807',
```

```
 'movie_860'},
'user_293': {'movie_1000',
 'movie_1059',
 'movie_114',
 'movie_153',
 'movie_189',
 'movie_289',
 'movie_357',
 'movie_44',
 'movie_485',
 'movie_57'},
'user_294': {'movie_101',
 'movie_157',
 'movie_189',
 'movie_239',
 'movie_357',
 'movie_471',
 'movie_498',
 'movie_629',
 'movie_719',
 'movie_83'},
'user_295': {'movie_100',
 'movie_169',
 'movie_36',
 'movie_50',
 'movie_58',
 'movie_618',
 'movie_622',
 'movie_725',
 'movie_843',
 'movie_91'},
'user_296': {'movie_130',
 'movie_133',
 'movie_139',
 'movie_200',
 'movie_242',
 'movie_289',
 'movie_357',
 'movie_49',
 'movie_502',
 'movie_89'},
'user_297': {'movie_130',
 'movie_242',
 'movie_31',
 'movie_325',
 'movie_432',
 'movie_491',
```

```
 'movie_630',
 'movie_719',
 'movie_751',
 'movie_95'},
'user_298': {'movie_1',
 'movie_1076',
 'movie_1167',
 'movie_120',
 'movie_136',
 'movie_153',
 'movie_289',
 'movie_403',
 'movie_719',
 'movie_83'},
'user_299': {'movie_1',
 'movie_104',
 'movie_139',
 'movie_268',
 'movie_380',
 'movie_404',
 'movie_48',
 'movie_719',
 'movie_804',
 'movie_851'},
'user_300': {'movie_189',
 'movie_216',
 'movie_229',
 'movie_351',
 'movie_357',
 'movie_400',
 'movie_52',
 'movie_571',
 'movie_629',
 'movie_979'},
'user_301': {'movie_1',
 'movie_136',
 'movie_140',
 'movie_157',
 'movie_209',
 'movie_216',
 'movie_239',
 'movie_297',
 'movie_357',
 'movie_906'},
'user_302': {'movie_10',
 'movie_140',
 'movie_189',
```

```
 'movie_244',
 'movie_289',
 'movie_299',
 'movie_57',
 'movie_62',
 'movie_725',
 'movie_793'},
'user_303': {'movie_108',
 'movie_34',
 'movie_488',
 'movie_561',
 'movie_599',
 'movie_622',
 'movie_649',
 'movie_665',
 'movie_852',
 'movie_86'},
'user_304': {'movie_0',
 'movie_120',
 'movie_166',
 'movie_191',
 'movie_247',
 'movie_254',
 'movie_31',
 'movie_47',
 'movie_647',
 'movie_801'},
'user_305': {'movie_0',
 'movie_1120',
 'movie_139',
 'movie_243',
 'movie_280',
 'movie_346',
 'movie_713',
 'movie_755',
 'movie_842',
 'movie_95'},
'user_306': {'movie_130',
 'movie_175',
 'movie_191',
 'movie_261',
 'movie_289',
 'movie_376',
 'movie_652',
 'movie_68',
 'movie_719',
 'movie_860'},
```

```
'user_307': {'movie_130',
 'movie_204',
 'movie_289',
 'movie_412',
 'movie_431',
 'movie_491',
 'movie_60',
 'movie_66',
 'movie_719',
 'movie_72'},
'user_308': {'movie_180',
 'movie_241',
 'movie_357',
 'movie_361',
 'movie_480',
 'movie_492',
 'movie_612',
 'movie_625',
 'movie_697',
 'movie_750'},
'user_309': {'movie_1049',
 'movie_145',
 'movie_324',
 'movie_347',
 'movie_351',
 'movie_354',
 'movie_368',
 'movie_373',
 'movie_499',
 'movie_746'},
'user_310': {'movie_230',
 'movie_241',
 'movie_256',
 'movie_275',
 'movie_321',
 'movie_431',
 'movie_58',
 'movie_583',
 'movie_607',
 'movie_618'},
'user_311': {'movie_100',
 'movie_200',
 'movie_285',
 'movie_29',
 'movie_317',
 'movie_341',
 'movie_346',
```

```
 'movie_44',
 'movie_562',
 'movie_730'},
'user_312': {'movie_110',
 'movie_136',
 'movie_145',
 'movie_17',
 'movie_174',
 'movie_179',
 'movie_253',
 'movie_262',
 'movie_271',
 'movie_46'},
'user_313': {'movie_100',
 'movie_256',
 'movie_297',
 'movie_317',
 'movie_408',
 'movie_436',
 'movie_471',
 'movie_476',
 'movie_58',
 'movie_982'},
'user_314': {'movie_1047',
 'movie_174',
 'movie_273',
 'movie_34',
 'movie_36',
 'movie_38',
 'movie_492',
 'movie_496',
 'movie_719',
 'movie_94'},
'user_315': {'movie_36',
 'movie_406',
 'movie_408',
 'movie_433',
 'movie_435',
 'movie_465',
 'movie_469',
 'movie_52',
 'movie_549',
 'movie_569'},
'user_316': {'movie_101',
 'movie_153',
 'movie_189',
 'movie_239',
```

```
 'movie_287',
 'movie_357',
 'movie_381',
 'movie_403',
 'movie_49',
 'movie_58'},
'user_317': {'movie_1',
 'movie_1243',
 'movie_139',
 'movie_491',
 'movie_538',
 'movie_571',
 'movie_574',
 'movie_594',
 'movie_910',
 'movie_915'},
'user_318': {'movie_1',
 'movie_1018',
 'movie_23',
 'movie_354',
 'movie_356',
 'movie_357',
 'movie_368',
 'movie_408',
 'movie_43',
 'movie_65'},
'user_319': {'movie_1',
 'movie_130',
 'movie_148',
 'movie_275',
 'movie_280',
 'movie_289',
 'movie_357',
 'movie_629',
 'movie_66',
 'movie_873'},
'user_320': {'movie_0',
 'movie_1',
 'movie_1200',
 'movie_1231',
 'movie_1343',
 'movie_157',
 'movie_357',
 'movie_43',
 'movie_491',
 'movie_960'},
'user_321': {'movie_112',
```

```
 'movie_118',
 'movie_1560',
 'movie_189',
 'movie_243',
 'movie_347',
 'movie_359',
 'movie_400',
 'movie_49',
 'movie_969'},
'user_322': {'movie_1018',
 'movie_1229',
 'movie_223',
 'movie_51',
 'movie_60',
 'movie_652',
 'movie_668',
 'movie_696',
 'movie_893',
 'movie_944'},
'user_323': {'movie_101',
 'movie_130',
 'movie_157',
 'movie_184',
 'movie_200',
 'movie_239',
 'movie_240',
 'movie_31',
 'movie_320',
 'movie_99'},
'user_324': {'movie_101',
 'movie_161',
 'movie_291',
 'movie_325',
 'movie_33',
 'movie_357',
 'movie_435',
 'movie_52',
 'movie_53',
 'movie_630'},
'user_325': {'movie_1',
 'movie_36',
 'movie_49',
 'movie_721',
 'movie_744',
 'movie_750',
 'movie_767',
 'movie_920',
```

```
 'movie_950',
 'movie_969'},
'user_326': {'movie_1',
 'movie_1017',
 'movie_175',
 'movie_181',
 'movie_274',
 'movie_519',
 'movie_594',
 'movie_60',
 'movie_814',
 'movie_95'},
'user_327': {'movie_120',
 'movie_141',
 'movie_339',
 'movie_355',
 'movie_360',
 'movie_364',
 'movie_404',
 'movie_695',
 'movie_696',
 'movie_960'},
'user_328': {'movie_1108',
 'movie_148',
 'movie_216',
 'movie_297',
 'movie_49',
 'movie_512',
 'movie_522',
 'movie_873',
 'movie_88',
 'movie_888'},
'user_329': {'movie_1054',
 'movie_204',
 'movie_221',
 'movie_247',
 'movie_365',
 'movie_367',
 'movie_510',
 'movie_644',
 'movie_77',
 'movie_778'},
'user_330': {'movie_1098',
 'movie_117',
 'movie_144',
 'movie_184',
 'movie_269',
```

```
 'movie_346',
 'movie_491',
 'movie_60',
 'movie_652',
 'movie_751'},
'user_331': {'movie_157',
 'movie_238',
 'movie_260',
 'movie_268',
 'movie_342',
 'movie_357',
 'movie_52',
 'movie_567',
 'movie_608',
 'movie_884'},
'user_332': {'movie_108',
 'movie_118',
 'movie_154',
 'movie_260',
 'movie_273',
 'movie_277',
 'movie_36',
 'movie_587',
 'movie_592',
 'movie_856'},
'user_333': {'movie_1056',
 'movie_38',
 'movie_491',
 'movie_492',
 'movie_496',
 'movie_538',
 'movie_56',
 'movie_571',
 'movie_58',
 'movie_66'},
'user_334': {'movie_0',
 'movie_144',
 'movie_157',
 'movie_213',
 'movie_268',
 'movie_289',
 'movie_343',
 'movie_568',
 'movie_860',
 'movie_928'},
'user_335': {'movie_1',
 'movie_254',
```

```
 'movie_262',
 'movie_288',
 'movie_32',
 'movie_611',
 'movie_649',
 'movie_687',
 'movie_93',
 'movie_94'},
'user_336': {'movie_244',
 'movie_309',
 'movie_321',
 'movie_33',
 'movie_428',
 'movie_433',
 'movie_56',
 'movie_6',
 'movie_78',
 'movie_816'},
'user_337': {'movie_116',
 'movie_175',
 'movie_353',
 'movie_554',
 'movie_585',
 'movie_608',
 'movie_609',
 'movie_719',
 'movie_814',
 'movie_98'},
'user_338': {'movie_189',
 'movie_239',
 'movie_31',
 'movie_357',
 'movie_420',
 'movie_585',
 'movie_719',
 'movie_925',
 'movie_928',
 'movie_950'},
'user_339': {'movie_24',
 'movie_31',
 'movie_471',
 'movie_473',
 'movie_497',
 'movie_5',
 'movie_538',
 'movie_66',
 'movie_89',
```

```
 'movie_91'},
'user_340': {'movie_1063',
 'movie_277',
 'movie_482',
 'movie_485',
 'movie_49',
 'movie_502',
 'movie_512',
 'movie_522',
 'movie_550',
 'movie_592'},
'user_341': {'movie_297',
 'movie_299',
 'movie_3',
 'movie_31',
 'movie_325',
 'movie_329',
 'movie_365',
 'movie_465',
 'movie_522',
 'movie_53'},
'user_342': {'movie_101',
 'movie_102',
 'movie_112',
 'movie_161',
 'movie_354',
 'movie_356',
 'movie_357',
 'movie_408',
 'movie_571',
 'movie_99'},
'user_343': {'movie_531',
 'movie_591',
 'movie_60',
 'movie_646',
 'movie_647',
 'movie_652',
 'movie_656',
 'movie_740',
 'movie_746',
 'movie_783'},
'user_344': {'movie_1120',
 'movie_130',
 'movie_139',
 'movie_184',
 'movie_243',
 'movie_251',
```

```
 'movie_325',
 'movie_346',
 'movie_568',
 'movie_807'},
'user_345': {'movie_101',
 'movie_156',
 'movie_216',
 'movie_217',
 'movie_22',
 'movie_23',
 'movie_357',
 'movie_408',
 'movie_52',
 'movie_719'},
'user_346': {'movie_1076',
 'movie_1294',
 'movie_188',
 'movie_49',
 'movie_512',
 'movie_552',
 'movie_565',
 'movie_59',
 'movie_656',
 'movie_855'},
'user_347': {'movie_102',
 'movie_144',
 'movie_239',
 'movie_247',
 'movie_280',
 'movie_403',
 'movie_452',
 'movie_60',
 'movie_861',
 'movie_971'},
'user_348': {'movie_36',
 'movie_377',
 'movie_379',
 'movie_473',
 'movie_498',
 'movie_512',
 'movie_522',
 'movie_541',
 'movie_574',
 'movie_592'},
'user_349': {'movie_115',
 'movie_144',
 'movie_157',
```

```
 'movie_265',
 'movie_43',
 'movie_681',
 'movie_719',
 'movie_755',
 'movie_815',
 'movie_824'},
'user_350': {'movie_1271',
 'movie_1453',
 'movie_200',
 'movie_389',
 'movie_595',
 'movie_612',
 'movie_669',
 'movie_699',
 'movie_772',
 'movie_780'},
'user_351': {'movie_101',
 'movie_161',
 'movie_22',
 'movie_31',
 'movie_311',
 'movie_356',
 'movie_357',
 'movie_462',
 'movie_501',
 'movie_719'},
'user_352': {'movie_239',
 'movie_240',
 'movie_260',
 'movie_273',
 'movie_289',
 'movie_436',
 'movie_496',
 'movie_512',
 'movie_522',
 'movie_538'},
'user_353': {'movie_1133',
 'movie_12',
 'movie_199',
 'movie_49',
 'movie_52',
 'movie_590',
 'movie_612',
 'movie_623',
 'movie_63',
 'movie_649'},
```

'user_354': {'movie_1231',
 'movie_130',
 'movie_133',
 'movie_289',
 'movie_511',
 'movie_568',
 'movie_60',
 'movie_8',
 'movie_83',
 'movie_86'},
'user_355': {'movie_1',
 'movie_130',
 'movie_268',
 'movie_280',
 'movie_48',
 'movie_60',
 'movie_652',
 'movie_681',
 'movie_719',
 'movie_890'},
'user_356': {'movie_1102',
 'movie_1114',
 'movie_120',
 'movie_242',
 'movie_357',
 'movie_40',
 'movie_600',
 'movie_83',
 'movie_860',
 'movie_89'},
'user_357': {'movie_1220',
 'movie_130',
 'movie_157',
 'movie_175',
 'movie_184',
 'movie_280',
 'movie_297',
 'movie_31',
 'movie_630',
 'movie_890'},
'user_358': {'movie_258',
 'movie_262',
 'movie_276',
 'movie_289',
 'movie_321',
 'movie_357',
 'movie_361',

```
 'movie_482',
 'movie_488',
 'movie_71'},
'user_359': {'movie_139',
 'movie_189',
 'movie_222',
 'movie_251',
 'movie_357',
 'movie_371',
 'movie_404',
 'movie_45',
 'movie_52',
 'movie_60'},
'user_360': {'movie_100',
 'movie_101',
 'movie_112',
 'movie_125',
 'movie_289',
 'movie_496',
 'movie_58',
 'movie_660',
 'movie_78',
 'movie_936'},
'user_361': {'movie_209',
 'movie_216',
 'movie_311',
 'movie_357',
 'movie_361',
 'movie_373',
 'movie_437',
 'movie_492',
 'movie_497',
 'movie_60'},
'user_362': {'movie_24',
 'movie_256',
 'movie_324',
 'movie_338',
 'movie_357',
 'movie_373',
 'movie_44',
 'movie_52',
 'movie_719',
 'movie_819'},
'user_363': {'movie_156',
 'movie_240',
 'movie_307',
 'movie_320',
```

```
 'movie_354',
 'movie_408',
 'movie_556',
 'movie_57',
 'movie_6',
 'movie_77'},
'user_364': {'movie_26',
 'movie_482',
 'movie_50',
 'movie_52',
 'movie_53',
 'movie_544',
 'movie_654',
 'movie_670',
 'movie_71',
 'movie_89'},
'user_365': {'movie_1',
 'movie_10',
 'movie_154',
 'movie_36',
 'movie_6',
 'movie_625',
 'movie_66',
 'movie_723',
 'movie_818',
 'movie_965'},
'user_366': {'movie_18',
 'movie_22',
 'movie_23',
 'movie_25',
 'movie_46',
 'movie_49',
 'movie_6',
 'movie_719',
 'movie_758',
 'movie_801'},
'user_367': {'movie_1004',
 'movie_29',
 'movie_31',
 'movie_513',
 'movie_577',
 'movie_655',
 'movie_710',
 'movie_739',
 'movie_886',
 'movie_895'},
'user_368': {'movie_118',
```

```
 'movie_182',
 'movie_20',
 'movie_230',
 'movie_253',
 'movie_357',
 'movie_364',
 'movie_403',
 'movie_474',
 'movie_58'},
'user_369': {'movie_200',
 'movie_254',
 'movie_31',
 'movie_408',
 'movie_433',
 'movie_440',
 'movie_502',
 'movie_567',
 'movie_754',
 'movie_89'},
'user_370': {'movie_1005',
 'movie_1164',
 'movie_31',
 'movie_338',
 'movie_361',
 'movie_421',
 'movie_513',
 'movie_602',
 'movie_61',
 'movie_806'},
'user_371': {'movie_113',
 'movie_130',
 'movie_29',
 'movie_31',
 'movie_321',
 'movie_367',
 'movie_60',
 'movie_719',
 'movie_88',
 'movie_95'},
'user_372': {'movie_154',
 'movie_256',
 'movie_29',
 'movie_36',
 'movie_476',
 'movie_60',
 'movie_602',
 'movie_652',
```

```
 'movie_783',
 'movie_801'},
'user_373': {'movie_1',
 'movie_153',
 'movie_161',
 'movie_180',
 'movie_209',
 'movie_289',
 'movie_297',
 'movie_357',
 'movie_52',
 'movie_719'},
'user_374': {'movie_307',
 'movie_31',
 'movie_311',
 'movie_34',
 'movie_36',
 'movie_377',
 'movie_424',
 'movie_431',
 'movie_449',
 'movie_496'},
'user_375': {'movie_1',
 'movie_217',
 'movie_289',
 'movie_297',
 'movie_361',
 'movie_366',
 'movie_367',
 'movie_482',
 'movie_5',
 'movie_753'},
'user_376': {'movie_118',
 'movie_125',
 'movie_24',
 'movie_50',
 'movie_53',
 'movie_600',
 'movie_750',
 'movie_767',
 'movie_781',
 'movie_822'},
'user_377': {'movie_1233',
 'movie_1239',
 'movie_125',
 'movie_581',
 'movie_595',
```

```
 'movie_611',
 'movie_612',
 'movie_67',
 'movie_702',
 'movie_78'},
'user_378': {'movie_181',
 'movie_213',
 'movie_216',
 'movie_273',
 'movie_5',
 'movie_571',
 'movie_694',
 'movie_711',
 'movie_920',
 'movie_94'},
'user_379': {'movie_1219',
 'movie_323',
 'movie_351',
 'movie_357',
 'movie_367',
 'movie_38',
 'movie_612',
 'movie_646',
 'movie_672',
 'movie_673'},
'user_380': {'movie_1',
 'movie_274',
 'movie_289',
 'movie_331',
 'movie_491',
 'movie_682',
 'movie_719',
 'movie_72',
 'movie_815',
 'movie_95'},
'user_381': {'movie_139',
 'movie_161',
 'movie_269',
 'movie_357',
 'movie_408',
 'movie_52',
 'movie_53',
 'movie_63',
 'movie_630',
 'movie_719'},
'user_382': {'movie_1',
 'movie_140',
```

```
 'movie_24',
 'movie_364',
 'movie_391',
 'movie_60',
 'movie_604',
 'movie_61',
 'movie_77',
 'movie_930'},
'user_383': {'movie_12',
 'movie_136',
 'movie_140',
 'movie_189',
 'movie_216',
 'movie_31',
 'movie_357',
 'movie_58',
 'movie_613',
 'movie_97'},
'user_384': {'movie_1046',
 'movie_1047',
 'movie_36',
 'movie_49',
 'movie_547',
 'movie_57',
 'movie_574',
 'movie_581',
 'movie_611',
 'movie_625'},
'user_385': {'movie_325',
 'movie_329',
 'movie_356',
 'movie_36',
 'movie_361',
 'movie_5',
 'movie_571',
 'movie_630',
 'movie_78',
 'movie_9'},
'user_386': {'movie_1',
 'movie_1225',
 'movie_140',
 'movie_239',
 'movie_387',
 'movie_40',
 'movie_49',
 'movie_568',
 'movie_652',
```

 'movie_89'},
'user_387': {'movie_1217',
 'movie_132',
 'movie_204',
 'movie_297',
 'movie_357',
 'movie_452',
 'movie_56',
 'movie_972',
 'movie_98',
 'movie_99'},
'user_388': {'movie_10',
 'movie_30',
 'movie_440',
 'movie_491',
 'movie_492',
 'movie_537',
 'movie_538',
 'movie_568',
 'movie_571',
 'movie_599'},
'user_389': {'movie_1',
 'movie_240',
 'movie_254',
 'movie_31',
 'movie_357',
 'movie_49',
 'movie_491',
 'movie_52',
 'movie_719',
 'movie_89'},
'user_390': {'movie_135',
 'movie_140',
 'movie_156',
 'movie_247',
 'movie_338',
 'movie_357',
 'movie_364',
 'movie_367',
 'movie_408',
 'movie_431'},
'user_391': {'movie_329',
 'movie_330',
 'movie_502',
 'movie_538',
 'movie_569',
 'movie_574',

```
 'movie_618',
 'movie_622',
 'movie_629',
 'movie_66'},
'user_392': {'movie_161',
 'movie_273',
 'movie_34',
 'movie_357',
 'movie_361',
 'movie_57',
 'movie_719',
 'movie_725',
 'movie_91',
 'movie_96'},
'user_393': {'movie_101',
 'movie_307',
 'movie_351',
 'movie_356',
 'movie_357',
 'movie_384',
 'movie_5',
 'movie_673',
 'movie_730',
 'movie_94'},
'user_394': {'movie_200',
 'movie_207',
 'movie_355',
 'movie_363',
 'movie_433',
 'movie_452',
 'movie_53',
 'movie_56',
 'movie_590',
 'movie_68'},
'user_395': {'movie_12',
 'movie_120',
 'movie_140',
 'movie_269',
 'movie_278',
 'movie_280',
 'movie_364',
 'movie_403',
 'movie_51',
 'movie_564'},
'user_396': {'movie_22',
 'movie_476',
 'movie_53',
```

```
 'movie_540',
 'movie_622',
 'movie_630',
 'movie_718',
 'movie_723',
 'movie_754',
 'movie_89'},
'user_397': {'movie_125',
 'movie_247',
 'movie_356',
 'movie_367',
 'movie_49',
 'movie_58',
 'movie_618',
 'movie_68',
 'movie_78',
 'movie_89'},
'user_398': {'movie_130',
 'movie_189',
 'movie_200',
 'movie_239',
 'movie_360',
 'movie_408',
 'movie_502',
 'movie_538',
 'movie_630',
 'movie_68'},
'user_399': {'movie_10',
 'movie_101',
 'movie_161',
 'movie_24',
 'movie_283',
 'movie_303',
 'movie_320',
 'movie_325',
 'movie_347',
 'movie_57'},
'user_400': {'movie_1',
 'movie_130',
 'movie_157',
 'movie_189',
 'movie_239',
 'movie_289',
 'movie_297',
 'movie_491',
 'movie_630',
 'movie_719'},
```

```
'user_401': {'movie_12',
 'movie_120',
 'movie_136',
 'movie_175',
 'movie_181',
 'movie_23',
 'movie_357',
 'movie_577',
 'movie_647',
 'movie_77'},
'user_402': {'movie_1016',
 'movie_29',
 'movie_498',
 'movie_52',
 'movie_556',
 'movie_56',
 'movie_57',
 'movie_60',
 'movie_61',
 'movie_82'},
'user_403': {'movie_144',
 'movie_157',
 'movie_245',
 'movie_251',
 'movie_289',
 'movie_404',
 'movie_491',
 'movie_652',
 'movie_719',
 'movie_881'},
'user_404': {'movie_112',
 'movie_361',
 'movie_5',
 'movie_50',
 'movie_52',
 'movie_571',
 'movie_692',
 'movie_694',
 'movie_699',
 'movie_835'},
'user_405': {'movie_101',
 'movie_260',
 'movie_321',
 'movie_329',
 'movie_361',
 'movie_367',
 'movie_389',
```

```
 'movie_488',
 'movie_93',
 'movie_99'},
'user_406': {'movie_1255',
 'movie_307',
 'movie_31',
 'movie_326',
 'movie_357',
 'movie_367',
 'movie_49',
 'movie_767',
 'movie_795',
 'movie_99'},
'user_407': {'movie_175',
 'movie_339',
 'movie_408',
 'movie_491',
 'movie_578',
 'movie_652',
 'movie_719',
 'movie_72',
 'movie_960',
 'movie_98'},
'user_408': {'movie_133',
 'movie_243',
 'movie_289',
 'movie_357',
 'movie_52',
 'movie_58',
 'movie_600',
 'movie_873',
 'movie_89',
 'movie_969'},
'user_409': {'movie_1672',
 'movie_29',
 'movie_3',
 'movie_31',
 'movie_36',
 'movie_566',
 'movie_568',
 'movie_6',
 'movie_613',
 'movie_623'},
'user_410': {'movie_1',
 'movie_325',
 'movie_355',
 'movie_4',
```

```
 'movie_43',
 'movie_491',
 'movie_568',
 'movie_58',
 'movie_95',
 'movie_960'},
'user_411': {'movie_1',
 'movie_239',
 'movie_289',
 'movie_31',
 'movie_376',
 'movie_397',
 'movie_506',
 'movie_58',
 'movie_630',
 'movie_719'},
'user_412': {'movie_161',
 'movie_174',
 'movie_276',
 'movie_307',
 'movie_321',
 'movie_357',
 'movie_36',
 'movie_367',
 'movie_52',
 'movie_6'},
'user_413': {'movie_195',
 'movie_216',
 'movie_307',
 'movie_401',
 'movie_405',
 'movie_482',
 'movie_541',
 'movie_571',
 'movie_667',
 'movie_81'},
'user_414': {'movie_101',
 'movie_102',
 'movie_200',
 'movie_209',
 'movie_216',
 'movie_307',
 'movie_321',
 'movie_357',
 'movie_562',
 'movie_719'},
'user_415': {'movie_1155',
```

```
 'movie_133',
 'movie_189',
 'movie_241',
 'movie_482',
 'movie_49',
 'movie_702',
 'movie_72',
 'movie_873',
 'movie_89'},
'user_416': {'movie_141',
 'movie_148',
 'movie_267',
 'movie_289',
 'movie_305',
 'movie_31',
 'movie_437',
 'movie_497',
 'movie_63',
 'movie_695'},
'user_417': {'movie_12',
 'movie_161',
 'movie_240',
 'movie_269',
 'movie_297',
 'movie_351',
 'movie_355',
 'movie_357',
 'movie_4',
 'movie_60'},
'user_418': {'movie_101',
 'movie_133',
 'movie_238',
 'movie_319',
 'movie_321',
 'movie_380',
 'movie_49',
 'movie_611',
 'movie_795',
 'movie_89'},
'user_419': {'movie_101',
 'movie_112',
 'movie_1271',
 'movie_174',
 'movie_175',
 'movie_216',
 'movie_357',
 'movie_571',
```

```
 'movie_78',
 'movie_98'},
'user_420': {'movie_1',
 'movie_115',
 'movie_130',
 'movie_157',
 'movie_239',
 'movie_49',
 'movie_491',
 'movie_502',
 'movie_695',
 'movie_950'},
'user_421': {'movie_371',
 'movie_408',
 'movie_418',
 'movie_482',
 'movie_49',
 'movie_52',
 'movie_66',
 'movie_71',
 'movie_91',
 'movie_913'},
'user_422': {'movie_1',
 'movie_115',
 'movie_117',
 'movie_130',
 'movie_261',
 'movie_325',
 'movie_484',
 'movie_491',
 'movie_719',
 'movie_72'},
'user_423': {'movie_1015',
 'movie_1103',
 'movie_243',
 'movie_343',
 'movie_519',
 'movie_627',
 'movie_736',
 'movie_815',
 'movie_976',
 'movie_997'},
'user_424': {'movie_1473',
 'movie_157',
 'movie_173',
 'movie_23',
 'movie_280',
```

'movie_305',
'movie_40',
'movie_60',
'movie_86',
'movie_881'},
'user_425': {'movie_1167',
'movie_144',
'movie_360',
'movie_407',
'movie_49',
'movie_502',
'movie_66',
'movie_815',
'movie_89',
'movie_95'},
'user_426': {'movie_10',
'movie_242',
'movie_325',
'movie_336',
'movie_427',
'movie_52',
'movie_60',
'movie_614',
'movie_644',
'movie_647'},
'user_427': {'movie_161',
'movie_22',
'movie_256',
'movie_29',
'movie_297',
'movie_31',
'movie_311',
'movie_329',
'movie_351',
'movie_592'},
'user_428': {'movie_101',
'movie_102',
'movie_24',
'movie_275',
'movie_277',
'movie_292',
'movie_297',
'movie_30',
'movie_52',
'movie_938'},
'user_429': {'movie_1',
'movie_101',

```
 'movie_297',
 'movie_330',
 'movie_357',
 'movie_491',
 'movie_574',
 'movie_660',
 'movie_695',
 'movie_779'},
'user_430': {'movie_200',
 'movie_220',
 'movie_238',
 'movie_36',
 'movie_496',
 'movie_611',
 'movie_635',
 'movie_66',
 'movie_660',
 'movie_694'},
'user_431': {'movie_201',
 'movie_228',
 'movie_247',
 'movie_291',
 'movie_302',
 'movie_356',
 'movie_364',
 'movie_479',
 'movie_693',
 'movie_719'},
'user_432': {'movie_1053',
 'movie_1069',
 'movie_200',
 'movie_31',
 'movie_49',
 'movie_501',
 'movie_595',
 'movie_77',
 'movie_873',
 'movie_99'},
'user_433': {'movie_487',
 'movie_530',
 'movie_552',
 'movie_564',
 'movie_60',
 'movie_614',
 'movie_77',
 'movie_774',
 'movie_809',
```

```
 'movie_83'},
'user_434': {'movie_1378',
 'movie_428',
 'movie_496',
 'movie_502',
 'movie_509',
 'movie_546',
 'movie_56',
 'movie_568',
 'movie_57',
 'movie_96'},
'user_435': {'movie_101',
 'movie_117',
 'movie_124',
 'movie_178',
 'movie_191',
 'movie_403',
 'movie_445',
 'movie_577',
 'movie_647',
 'movie_77'},
'user_436': {'movie_1167',
 'movie_117',
 'movie_136',
 'movie_175',
 'movie_243',
 'movie_244',
 'movie_51',
 'movie_652',
 'movie_719',
 'movie_95'},
'user_437': {'movie_161',
 'movie_216',
 'movie_234',
 'movie_239',
 'movie_240',
 'movie_311',
 'movie_357',
 'movie_499',
 'movie_794',
 'movie_900'},
'user_438': {'movie_0',
 'movie_1015',
 'movie_1058',
 'movie_1493',
 'movie_199',
 'movie_357',
```

```
 'movie_40',
 'movie_498',
 'movie_519',
 'movie_600'},
'user_439': {'movie_102',
 'movie_1219',
 'movie_1395',
 'movie_216',
 'movie_285',
 'movie_355',
 'movie_4',
 'movie_471',
 'movie_751',
 'movie_93'},
'user_440': {'movie_157',
 'movie_174',
 'movie_217',
 'movie_256',
 'movie_403',
 'movie_49',
 'movie_52',
 'movie_569',
 'movie_623',
 'movie_73'},
'user_441': {'movie_1084',
 'movie_1127',
 'movie_262',
 'movie_49',
 'movie_512',
 'movie_574',
 'movie_695',
 'movie_779',
 'movie_808',
 'movie_913'},
'user_442': {'movie_1',
 'movie_22',
 'movie_36',
 'movie_395',
 'movie_454',
 'movie_475',
 'movie_498',
 'movie_5',
 'movie_503',
 'movie_70'},
'user_443': {'movie_1',
 'movie_117',
 'movie_189',
```

```
 'movie_268',
 'movie_289',
 'movie_357',
 'movie_404',
 'movie_420',
 'movie_491',
 'movie_960'},
'user_444': {'movie_1012',
 'movie_133',
 'movie_148',
 'movie_1578',
 'movie_289',
 'movie_30',
 'movie_534',
 'movie_568',
 'movie_695',
 'movie_77'},
'user_445': {'movie_1015',
 'movie_274',
 'movie_280',
 'movie_376',
 'movie_43',
 'movie_484',
 'movie_60',
 'movie_608',
 'movie_814',
 'movie_98'},
'user_446': {'movie_1247',
 'movie_217',
 'movie_31',
 'movie_49',
 'movie_491',
 'movie_531',
 'movie_57',
 'movie_586',
 'movie_589',
 'movie_7'},
'user_447': {'movie_130',
 'movie_389',
 'movie_438',
 'movie_511',
 'movie_652',
 'movie_68',
 'movie_719',
 'movie_754',
 'movie_860',
 'movie_960'},
```

'user_448': {'movie_130',
 'movie_209',
 'movie_216',
 'movie_297',
 'movie_30',
 'movie_351',
 'movie_408',
 'movie_49',
 'movie_52',
 'movie_646'},
'user_449': {'movie_1',
 'movie_130',
 'movie_157',
 'movie_229',
 'movie_297',
 'movie_357',
 'movie_4',
 'movie_479',
 'movie_49',
 'movie_60'},
'user_450': {'movie_113',
 'movie_157',
 'movie_189',
 'movie_216',
 'movie_231',
 'movie_254',
 'movie_273',
 'movie_289',
 'movie_320',
 'movie_357'},
'user_451': {'movie_1',
 'movie_101',
 'movie_130',
 'movie_161',
 'movie_357',
 'movie_380',
 'movie_502',
 'movie_52',
 'movie_522',
 'movie_98'},
'user_452': {'movie_1078',
 'movie_1120',
 'movie_115',
 'movie_174',
 'movie_346',
 'movie_491',
 'movie_502',

```
 'movie_568',
 'movie_842',
 'movie_960'},
'user_453': {'movie_297',
 'movie_320',
 'movie_321',
 'movie_34',
 'movie_357',
 'movie_36',
 'movie_361',
 'movie_367',
 'movie_389',
 'movie_59'},
'user_454': {'movie_125',
 'movie_133',
 'movie_139',
 'movie_161',
 'movie_189',
 'movie_216',
 'movie_36',
 'movie_4',
 'movie_465',
 'movie_483'},
'user_455': {'movie_108',
 'movie_111',
 'movie_189',
 'movie_496',
 'movie_503',
 'movie_506',
 'movie_597',
 'movie_600',
 'movie_62',
 'movie_89'},
'user_456': {'movie_229',
 'movie_420',
 'movie_51',
 'movie_511',
 'movie_594',
 'movie_60',
 'movie_652',
 'movie_719',
 'movie_77',
 'movie_98'},
'user_457': {'movie_10',
 'movie_104',
 'movie_140',
 'movie_161',
```

```
  'movie_305',
  'movie_31',
  'movie_445',
  'movie_531',
  'movie_577',
  'movie_95'},
 'user_458': {'movie_1069',
  'movie_133',
  'movie_148',
  'movie_224',
  'movie_49',
  'movie_58',
  'movie_695',
  'movie_719',
  'movie_807',
  'movie_873'},
 'user_459': {'movie_157',
  'movie_200',
  'movie_239',
  'movie_357',
  'movie_40',
  'movie_43',
  'movie_502',
  'movie_512',
  'movie_719',
  'movie_772'},
 'user_460': {'movie_100',
  'movie_139',
  'movie_189',
  'movie_198',
  'movie_200',
  'movie_209',
  'movie_216',
  'movie_240',
  'movie_49',
  'movie_661'},
 'user_461': {'movie_31',
  'movie_356',
```

[ ]:
```
top_k = 5
user_ids_to_check = [121, 31, 77]

# Invert index map if needed
index_to_user_id = {idx: uid for uid, idx in user_id_to_index.items()}
user_id_to_row = {uid: idx for idx, uid in enumerate(all_user_ids)}  # for␣
↪direct row lookup

for uid in user_ids_to_check:
```

```
  'movie_101',
  'movie_1098',
  'movie_1473',
  'movie_22',
  'movie_325',
  'movie_357',
```

```
    user_row = user_id_to_row[uid]
    predicted_scores = rating_matrix_scaled_1_5[user_row]

    # Get top k movie indices
    top_k_indices = np.argsort(predicted_scores)[::-1][:top_k]
    top_movie_ids = [all_movie_ids[i] for i in top_k_indices]
    top_scores = predicted_scores[top_k_indices]

    # Get movie titles
    titles = movies.set_index('item_id').loc[top_movie_ids]['movie_title'].
↪values

    print(f"\nTop {top_k} Recommendations for User {uid}:")
    for mid, title, score in zip(top_movie_ids, titles, top_scores):
        print(f"  Movie ID: {mid}, Title: {title}, Predicted Rating: {score:.
↪3f}")
```

```
Top 5 Recommendations for User 121:
  Movie ID: 1204, Title: To Be or Not to Be (1942), Predicted Rating: 4.289
  Movie ID: 175, Title: Brazil (1985), Predicted Rating: 4.242
  Movie ID: 493, Title: Thin Man, The (1934), Predicted Rating: 4.210
  Movie ID: 7, Title: Twelve Monkeys (1995), Predicted Rating: 4.175
  Movie ID: 134, Title: Citizen Kane (1941), Predicted Rating: 4.158

Top 5 Recommendations for User 31:
  Movie ID: 1086, Title: It's My Party (1995), Predicted Rating: 4.267
  Movie ID: 218, Title: Cape Fear (1991), Predicted Rating: 4.116
  Movie ID: 888, Title: One Night Stand (1997), Predicted Rating: 4.113
  Movie ID: 1227, Title: Awfully Big Adventure, An (1995), Predicted Rating:
4.041
  Movie ID: 1053, Title: Now and Then (1995), Predicted Rating: 4.001

Top 5 Recommendations for User 77:
  Movie ID: 1334, Title: Somebody to Love (1994), Predicted Rating: 4.317
  Movie ID: 750, Title: Amistad (1997), Predicted Rating: 4.316
  Movie ID: 1634, Title: Etz Hadomim Tafus (Under the Domin Tree) (1994),
Predicted Rating: 4.309
  Movie ID: 1053, Title: Now and Then (1995), Predicted Rating: 4.281
  Movie ID: 1397, Title: Of Human Bondage (1934), Predicted Rating: 4.273
```

[ ]:
```
print("user121 soft:", top_soft['user_121'])
print("user121 hard:", top_hard['user_121'])
```

```
user121 soft: {'movie_58', 'movie_166', 'movie_239', 'movie_1077', 'movie_996',
'movie_189', 'movie_342', 'movie_359', 'movie_240', 'movie_221'}
user121 hard: {'movie_166', 'movie_58', 'movie_239', 'movie_1077', 'movie_996',
'movie_189', 'movie_342', 'movie_359', 'movie_240', 'movie_221'}
```

[ ]:
```python
print("user31 soft:", top_soft['user_31'])
print("user31 hard:", top_hard['user_31'])
```

```
user31 soft: {'movie_36', 'movie_408', 'movie_34', 'movie_544', 'movie_66',
'movie_884', 'movie_77', 'movie_31', 'movie_78', 'movie_694'}
user31 hard: {'movie_544', 'movie_66', 'movie_77', 'movie_78', 'movie_512',
'movie_522', 'movie_694', 'movie_319', 'movie_819', 'movie_1255'}
```

[ ]:
```python
print("user77 soft:", top_soft['user_77'])
print("user77 hard:", top_hard['user_77'])
```

```
user77 soft: {'movie_216', 'movie_635', 'movie_60', 'movie_49', 'movie_61',
'movie_31', 'movie_209', 'movie_78', 'movie_157', 'movie_102'}
user77 hard: {'movie_311', 'movie_60', 'movie_314', 'movie_49', 'movie_61',
'movie_31', 'movie_101', 'movie_209', 'movie_157', 'movie_102'}
```

[ ]:
```python

```