



Univerza v Ljubljani
Fakulteta
za računalništvo
in informatiko

Recommender System

Fourth homework assignment for Datamining

Ardit Nela (63190375)

User Interface

I created a GUI using the PyQt library (the template file is stored as a .ui file and I load it in my main python script).

User Profile

User:

Movie:

Rating:

Your ratings:

Select an action using the buttons (once the action is complete this field will notify you of its completion)

Recommendations:

Default User Profile

The image shows a window titled "User Profile" with a standard Windows-style title bar (minimize, maximize, close buttons). Inside the window, there are three input fields at the top: "User:" with a text box containing the number "3", "Movie:" with a dropdown menu showing "Toy Story (1995)", and "Rating:" with a spinner box showing "0.00". Below these is a section labeled "Your ratings:" containing a large empty text area with the word "Done!" at the top. At the bottom, there is a "Recommendations:" section with a dropdown menu. The dropdown menu is open, showing a list of movie titles: "Father of the Bride Part II (1995)", "Father of the Bride Part II (1995)", "Grumpier Old Men (1995)", "Casino (1995)", "Sudden Death (1995)", and "Back to the Future (1985)". The first two entries are highlighted in blue. To the right of the dropdown menu are two buttons: "Suggest" and "Save Rating". At the very bottom of the window, there is a small black area with green text that reads "In [2]: runf".

After a user ID is entered and the “Suggest” button is pressed, your movie recommendations will appear in a combo box.

User Profile

User:

777

Movie:

Ghostbusters (2016)

Rating:

2.00

Your ratings:

Toy Story (1995):3.0 + Heat (1995):5.0 + Ghostbusters (2016):2.0 + Shin Godzilla (2016):4.0 +

Recommendations:

Suggest

Save Rating

You may also add a new rating for the given user.

NOTE: Suggestions only work for already existing users.

Recommender

I decided to use matrix factorization (more specifically, singular value decomposition) to build my recommender system (I was inspired by the early days of Netflix). I used the built-in `numpy.linalg.svd` function to get `u`, `sigma` and `vh`. Afterwards, I calculated the recommendation matrix by using their dot product:

```
# to ease multiplication
sigma = np.diag(sigma)
pred_matrix = np.dot(np.dot(u, sigma), vh)
```

Finally, I wrote a function that would give 5 suggestions for the given user based on the prediction matrix:

```
def getRecommendations(self, userID, movies, ratings, predictions):
    user_row_number = userID - 1
    sorted_preds = predictions.iloc[user_row_number].sort_values(ascending=False)

    # get user data and merge with movies
    user_data = ratings[ratings["userId"] == userID]
    user_full = (user_data.merge(movies, how = 'left', left_on = 'movieId', right_on = 'movieId').
                 sort_values(['rating'], ascending=False))

    recommendations = (movies[~movies['movieId'].isin(user_full['movieId'])].
                       merge(pd.DataFrame(sorted_preds).reset_index(), how = 'left',
                              left_on = 'movieId',
                              right_on = 'movieId').
                       rename(columns = {user_row_number: 'predictions'}).
                       sort_values('predictions', ascending = False).
                       iloc[:5, :-1])

    return user_full, recommendations
```