Here are the steps you might follow:

1. Data Loading and Preprocessing:

* First, import necessary libraries such as pandas, numpy, and PyTorch.
* Load your datasets (links.csv, movie.csv, rating.csv, tags.csv) using pandas’s read\_csv function
* Preprocess your data. This might involve handling missing values, converting categories variables into numerical ones, or normalizing numerical variables.

1. Data Analysis:

* Perform some exploratory data analysis to understand your data better. This might involve looking at the distribution of movie ratings, the number of ratings per movie, etc.

1. Feature Extraction:

* Decide on which features you want to use to predict movie ratings. These might be drived from movies.csv and tags.csv file. Features could include genres, movies, titles, or tags.

1. Data Splitting:

* Split your data into a training set and a test set. A common split might be 80% of your data for training and 20% for testing.

1. Model Building:

* Define your model architecture. With PyTorch, you’ll define a class for your model that extends torch.nn.Module. In the class, you’ll define your layers in the \_\_init\_\_ method and how to those layers are connected in the forward method

1. Training:

* Train your model using your training data. This will involve choosing an optimizer and a loss function. You’ll use your optimizer to update your model’s weights based on the gradient computed from your loss function.

1. Evaluation:

* Evaluate your model’s performance on your test data. You might look at metrics such as mean squared error to do this.

1. Hyperparameter Tuning:

* Depending on your model’s performance, you can use it to predict the rating of movies.

1. Prediction:

* Once you’re satisfied with your model’s performance, you can use it to predict the ratings of movies.