Take-Home Assessment - BE Engineer

The following is the schema for the ACME data warehouse:

Table users:
- id - primary key int
- first_name - varchar
- last_name - varchar
- year_joined - int (could potentially be datetime as well)
- country - varchar

This is me post writing all the scripts, but country could be its own normalized table so that we only include country_id in the users table and have a separate table for country/location.

users				
id	**first_name**	**last_name**	**year_joined**	**country**

Table movies:
- id - primary key int
- name - varchar
- year - int (could potentially be datetime as well)

movies		
id	**name**	**year**

Table engagement:
- id - primary key int
- user_id - int
- movie_id - int
- watchtime - int (number of mins)

The id primary key is not used in this assessment, but I think for future expansion of this database where we might add more variables to engagement, the primary key may be useful.

engagement			
id	**movie_id**	**user_id**	**watchtime**

The queries for the questions are in the queries.py file attached in the email.

The first prediction model is going to be based on maximum watchtime. The movie with the maximum watchtime in each country is going to be the recommended movie for users in that country. If multiple movies are to be recommended, they can be recommended as the top n movies based on watchtime in the country.

Testing:

In order to test how well this prediction model performs, we could perform A/B testing on the model.

This will involve a control group and a treatment group. The testing could be done on both existing as well as new users as two separate entities but for simplicity we can keep the test groups high level. The control group will have no recommendations suggested to them and we will monitor what movies they end up watching and for how long (watchtime). The treatment group will be suggested the recommended movies based on the prediction model and we will monitor if they clicked on the suggestion and if so how long they watched it for (to determine if they liked it or not).

We will finally compare the results of the two groups to see whether both sets of users watched and liked watching the same movies (in the same region/country) to determine how well the prediction model performed. If both groups watched and enjoyed watching the same movies, then we know that the model could be performing well. Otherwise, maybe the model is not as successful, or there may be external factors to consider.