Hollywood +Bollywood

Brandon Hubacher & Natania Christopher

Area of interest

We are curious if there are patterns to be found between Hollywood and its (culturally distinct) cousin Bollywood. Is there a trend in the types of genres released that is the same or different between the two? Surprising similarities or marked disparities could provide insight to the culture surrounding the movie industry.

Difficulties: The bollywood dataset provided minimal data in quantity and variety compared to the IMDb dataset. It was difficult to make insightful queries that aligned with our interests.

Dataset Selections and Overview of Raw Data

IMDb dataset: https://www.imdb.com/interfaces/

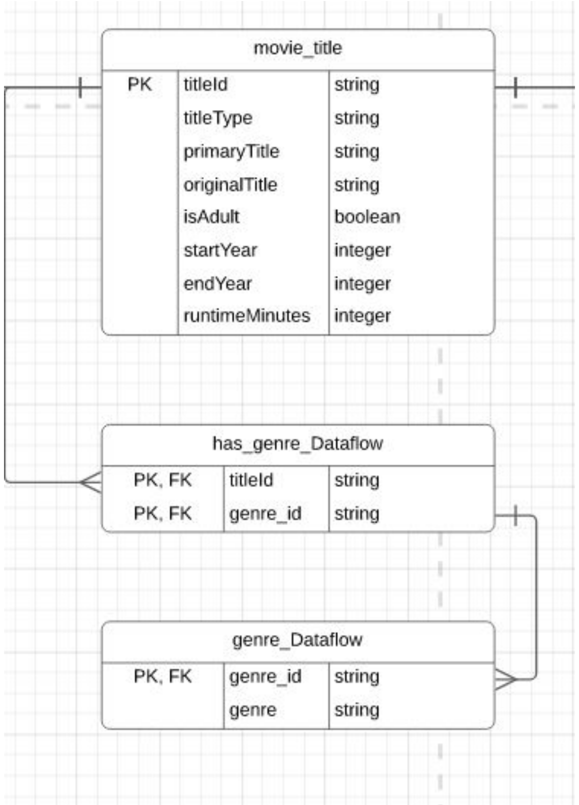
Kaggle (Bollywood) dataset:
https://www.kaggle.com/alokdas527/bollywood-movies-analysis

Staging and Modeled Tables

For some tables we had to manually create the schema, e.g. title_basics, because the csv file contained columns of the ARRAY data type. These would later have to be cleaned in apache beam to conform to a proper SQL schema.

Model tables: we had to save splitting tables for apache beam in milestone 3. The only thing done here was to cast easily converted data types (like from string to integer). We also added columns to be able to UNION the title_basics and corresponding bollywood tables. We had to generate universal unique ids to be the titleID for the bollywood dataset. The length of this generated ID would later prove very helpful in distinguishing between the industry for queries.

Beam pipelines & cross-dataset queries



movie_title

PK	titleId	string
	titleType	string
	primaryTitle	string
	originalTitle	string
	isAdult	boolean
	startYear	integer
	endYear	integer
	runtimeMinutes	integer

has_genre_Dataflow

PK, FK	titleId	string
PK, FK	genre_id	string

genre_Dataflow

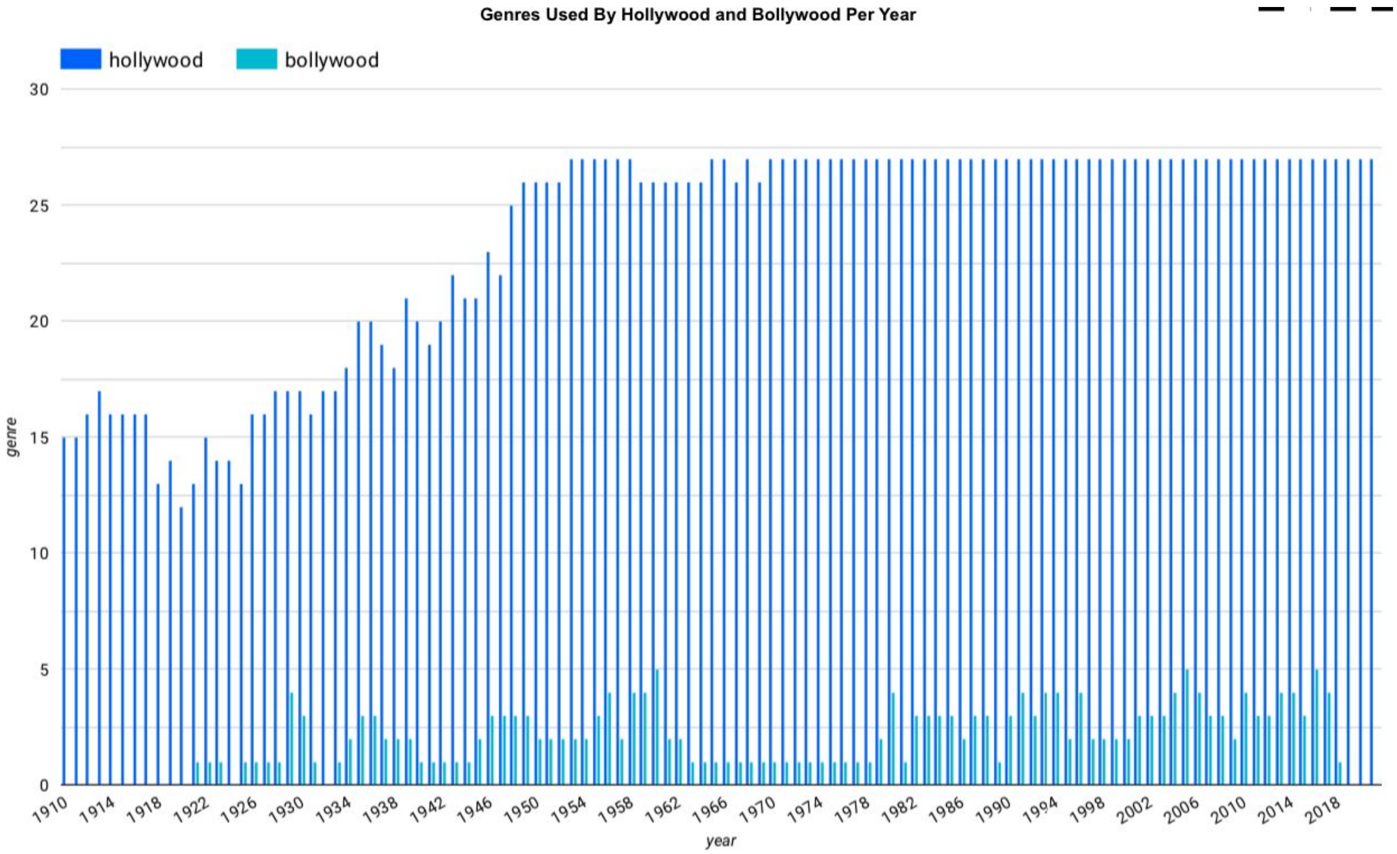
PK, FK	genre_id	string
	genre	string

Our goal with creating the beam pipelines in apache beam was to take the genres (an array represented as a string) and convert it to a table similar to a junction table (has_genre) and a genre table (genre). This is because there's a m:m relationship between a genre and movie title.

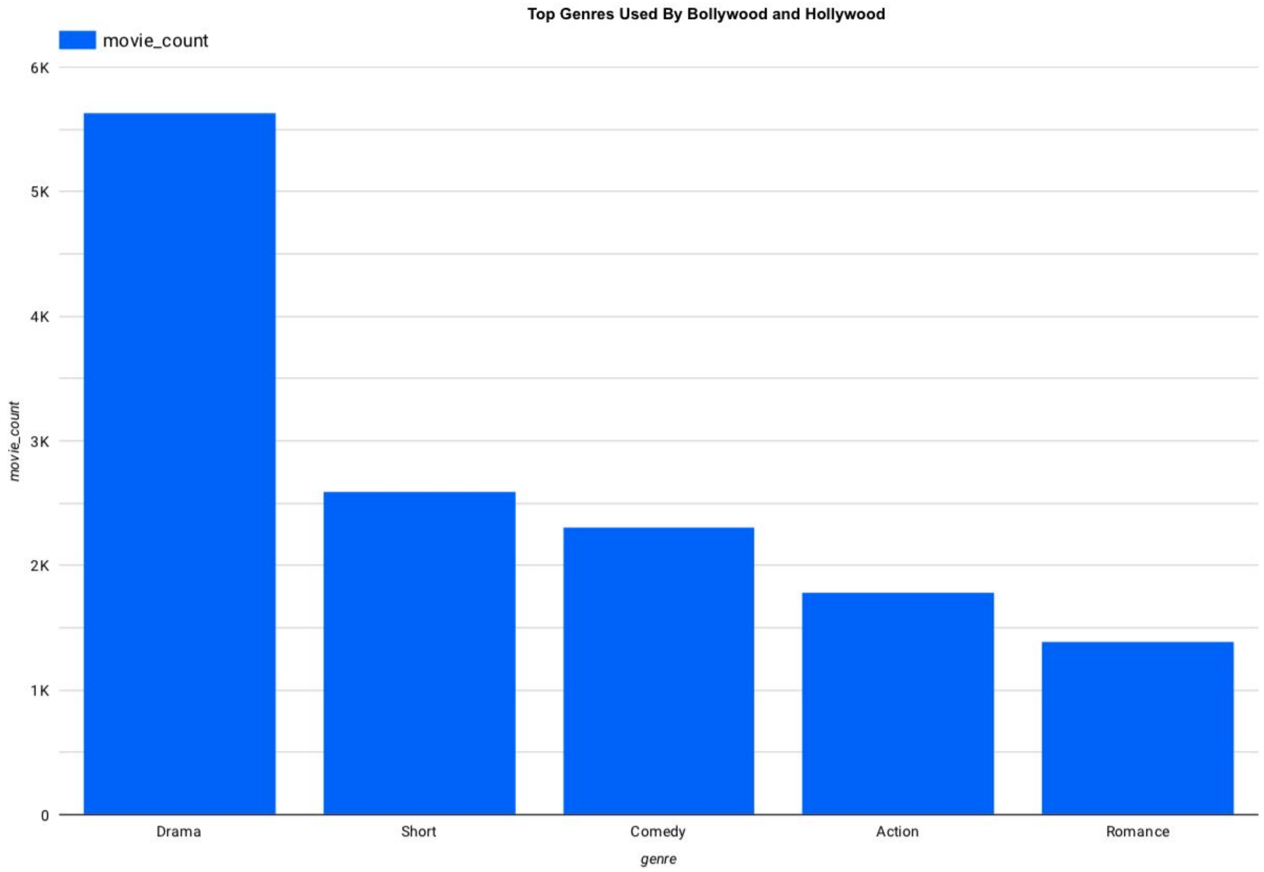
We could not limit our query to 500 results because it would ruin referential integrity.

We had to generate unique genre ids for movie titles using MD5, a hash function. The genre was passed in and its hash value was then converted to a string using the TO_BASE64() function.

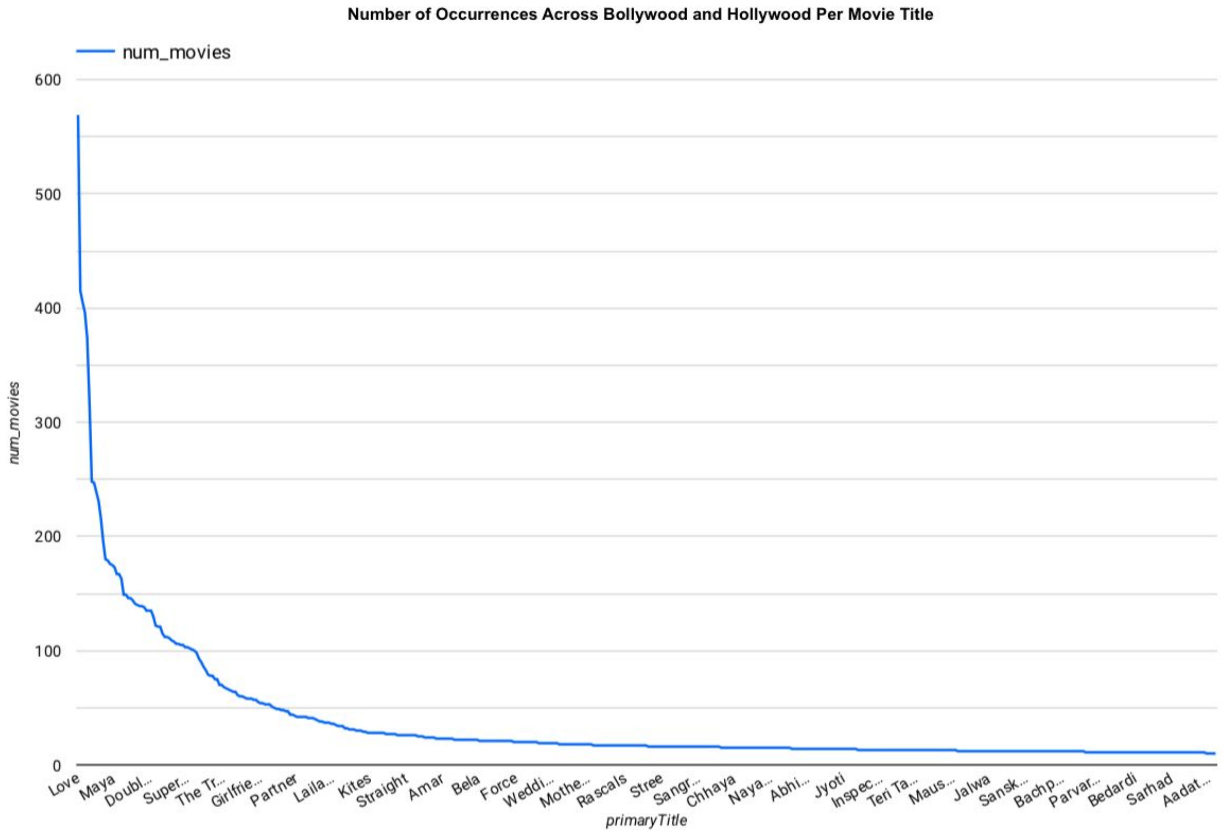
Data visualization #1



Data visualization #2



Data visualization #3



Future improvements

- Find insight on other information in datasets, like Directors, Actors, and actresses
- Use a dataset with more information on Bollywood movies - this would give us a more comprehensive and potentially holistic view of the similarities/differences between Hollywood and Bollywood
- For tables that we combined in to one (e.g. title_basics and bollywood), clean the tables separately to be able to perform joins for ease of querying. Formulating subqueries on one table is much more difficult.