Milestone 2 Submission

April 12, 2017

0.1 CS109b Final Project - Group 24

Tyler Finkelstein
James Fallon
Aaron Myran
Ihsan Patel

1 Description of Data

Data was initially obtained using the TMDb popular movies API, which returned the ~20,000 most popular movies in TMDb's database along with some metadata for the movies. This data did not, however, have an IMDb ID for the movies, so we looped through each TMDb record and used its TMDb id to obtain the IMDb id and 15 other features (described in the feature selection section) through TMDb's search function. Movie posters were also pulled using the link provided by TMDb.

Before pulling the IMDb data, we filtered the dataset for english language movies only to ensure the metadata (particularly things like plot summaries) remained consistent and based on the assumption that cultural differences could lead the same genre to have very different characteristics depending on the culture.

We then pulled IMDb data using the `IMDBPY` library which yielded an additional 91 features for each movie not all of which were used (the ones that were used are described in the feature selection section).

2 What We're Predicting

One of the issues we've encountered with the datasets is the fact that the IMDb and TMDb genres do not always match up. We discussed a variety of options for dealing with this including. First, though, we agreed that there are two types of matching genres across the two data sources: 1. Exact matches - IMDb and TMDb have the same exact name for a type of movie. 2. Synonymous genres - IMDb list "Sci-Fi", TMDb lists "Science Fiction", and clearly these mean the same. Therefore, the first step is to convert all genres names to one common language.

Then, we came up with the following options: 1. Unioning lists of genres together for each movie. For example, if IMDb lists only Comedy, and TMDb lists only Family, our result would be Comedy and Family. 2. Intersectioning lists of genres together for each movie. In other words, for a given movie, only include genres that are included on both websites. 3. Considering genre counts across each data source, with the idea being if both sources list a given genre, that should be of stronger predictive value than only one source listing a genre, but we can still keep genres

not listed from both sources. 4. Using genres from just one data source, and completely ignoring the genre data from the other data source.

After thinking this through, we believe that #2 is the best option here. The two data sources appear to have fairly consistent genre classifications in the first place, and by taking the intersection of genres across them, we can be more confident in our predictions. In the small number of cases where there is no common genre across the two data sources, we will include the genres from both databases (alternative: get rid of the movie in our dataset).

To be clear, what we will be predicting is whether a given genre is listed for a movie on both IMDb and TMDb. We are therefore predicting a boolean for each genre for each movie.

3 Response variable: Choosing Y

Our approach to choosing our response variable is to selectively pare down the genre for each film to one categorical value. The steps are to first reconcile TMDb and IMDb data (building off the consideration above), determine joint categories that are common enough to deserve their own genre label (for example "romantic comedies"), and then assign one genre category to movies that have been given multiple labels by both TMDb and IMDb.

Algorithmic approach: 1. For a given movie, include only genres that are included on both IMDb and TMDb. 2. Examine which genres appear together consistently (for example identify genre pairs or triplets which make up more than 5% or 10% of all genre labels) and make those their own labels (example: "romantic comedy"). 3. If a movie is assigned only one genre, keep that genre label.
4. If a movie is assigned more than one genre label, keep the label that is most frequent in the data (including the multi-categories).
5. Build a "Genre" category with a clean, single consistent categorical variable for each film.

4 Unbalanced Data: Some Genres Are Overrepresented

Another issue to address is imbalanced genre frequencies. For example, from our exploratory data analysis, we saw that Drama is about 5 times as popular as Sci-Fi. We would not want our classifier to be heavily biased towards Drama, so we considered the following adjustments: 1. Randomly undersampling the majority classes to have the same sample size as the minority class. 2. Randomly oversampling the minority classes to have the same sample size as the majority class. This uses duplication of minority class examples, which seems less ideal than having distinct examples throughout the dataset. We believe that method 1 is more favorable than this method. We already will be dealing with a relatively large dataset, so oversampling minority classes (by duplication) in order to produce an even larger dataset seems unnecessary. Instead, we will have a training dataset where all samples are unique. 3. Building cost-sensitive classifiers that take into account the imbalanced nature of the data when making predictions. We can adjust the prediction thresholds upwards or downwards from 0.5. While this method makes sense statistically, it will require adjustment of every model we build. Given the scope, timeline, and collaborative nature of the project, it may slow things down significantly. 4. Guided undersampling of the majority classes: remove majority samples that are very similar to one another. A problem with this is that we don't have an appropriate method of measuring distance between movies. 5. Synthetic oversampling of the minority classes: create artificial examples of the minority class. Why do this when we have a large dataset available?

For some of the reasons explained, we choose method 1: under sampling the majority classes.

In cases where we are not able to run the model on the entire dataset, we plan to undersample larger categories first to build a balanced dataset, and then sample randomly from the resulting dataset.

5 Feature Choice

5.0.1 Features from TMDb:

- **Poster** - There may be visual information within each poster that is associated with each genre
- **Part of Collection**- TMDb provides a `belongs_to_collection` feature which not all movies have. Based on some EDA it looks like the proportion of genres significantly depending on wheter or not movies are part of a collection. For example, dramas are 15 percentage points less of the total proportion of movies in a collection as compared to non-collection movies, while action and adventure movies are 5% more. This aligns with intuition since we would expect action and adventure movies to more likely to be parts of collections compared to dramas due to superhero movies and other action trilogies (Star Wars, Lord of the Rings, etc.).
- **Budget** - As our EDA from the previous milestone illustrated, budget varies significantly depending on genre.
- **Overview** - We plan to use both the length of the overview as well as some text analysis (tf - idf) since genres likely have different words describing the plot of the movie (i.e. love for romance, guns for action).
- **Popularity** - Some genres could be more popular than other genres (ie. action vs. documentaries).
- **# of Production Companies** - The number of production companies could vary by genre in a way similar to budget (more production companies needed for higher cost movies).
- **# of Production Countries** - Some genres (action, sci-fi) may require more landscapes and environments than others (comedy, romance) do.
- **Release Date** - We will look at both year and month. Genres may wax and wane in poularity over time (in which case year is useful), while some periods of the year are known for specific types of movies (action movies are released in the summer).
- **Revenue** - Similar to popularity (the more consumers the more revenue)
- **Runtime** - Some genres of movies may be longer than others (long action movies vs. short romantic comedies)
- **# of Spoken Languages** - Similar to Production Countries
- **Tagline** - Similar to Overview
- **Length of Title** - Some genres may have longer titles than other genres
- **Vote Average** - Assuming this is a proxy for quality, some genres may have on-average higher quality movies than other genres
- **Vote Count** - Similar to Popularity / Revenue

5.0.2 Features from IMDb:

- **Various Department Sizes (animation, art, camera and electrical, casting, costume, editorial, makeup, music, special effects, visual)** - All of these not only serve a similar purpose to Budget in IMDb but also seem like they could vary in other ways un-related to buget (sci-fi movies may have higher special effects budgets)

- **Cast Size** - Similar to Department Sizes
- **# of Distributors** - Similar to Revnue in TMDb data (more distributors could signify more demand)
- **MPAA** - The rating and rating description will be used. Ratings likely differ between genres ("PG" for Family vs. "R" for Horror), and the actual text of the rating (violence, language, etc.) could give a clue to the type of movie as well.
- **Plot /Plot Outline** - Similar to Overview from TMDb
- **Rating** - Similar to Vote Average in TMDb
- **Votes** - Similar to Vote Count in TMDb