



# Film Industry Analysis



Amy Li & Akshay Sharma



# Questions

---

- What is the optimal time to release a movie?
- A look into customer vs critic
- What kind of movies do viewers want to watch?
- Which directors are trending in the 21st century?

# How we got the data - Scrape

```
reviews = []  
titles = []  
num_reviews = []  
all_data = []
```

#function to scrape RT for movie titles, review score, and number of reviews

```
def movie_info(year):
```

```
    url = 'https://www.rottentomatoes.com/top/bestofrt/?year=' + str(year)
```

```
    r = requests.get(url)
```

```
    c = r.content
```

```
    soup = BeautifulSoup(c, 'html.parser')
```

```
    all_movies = soup.find('div', class_ = 'panel-body content_body allow-overflow')
```

```
    for movies in all_movies.find_all('a', class_='unstyled articleLink'):
```

```
        titles.append(movies.text.split('\n')[1].strip())
```

```
    for movies in all_movies.find_all('span', class_='tMeterScore'):
```

```
        reviews.append(movies.text.split('\xa0')[1])
```

```
    for movies in all_movies.find_all('td', class_='right hidden-xs'):
```

```
        num_reviews.append(movies.text)
```

```
def movie_info(movie):
    url = 'https://www.rottentomatoes.com/m/' + movie.replace(' ', '_')
    r = requests.get(url)
    c = r.content
    soup = BeautifulSoup(c, 'html.parser')
    director = []
    writer = []
    movies_info = {}
    #To get directors, writers, box office, movie duration, and when it was released in theaters
    info = soup.find_all('div', class_='meta-value')

    try:
        for directors in info[2].find_all('a'):
            director.append(directors.text)

        movies_info['Movie Title'] = movie
        movies_info['Director'] = director

        for writers in info[3].find_all('a'):
            writer.append(writers.text)

        movies_info['Writers'] = writer

        movies_info['Release Date'] = info[4].time.string

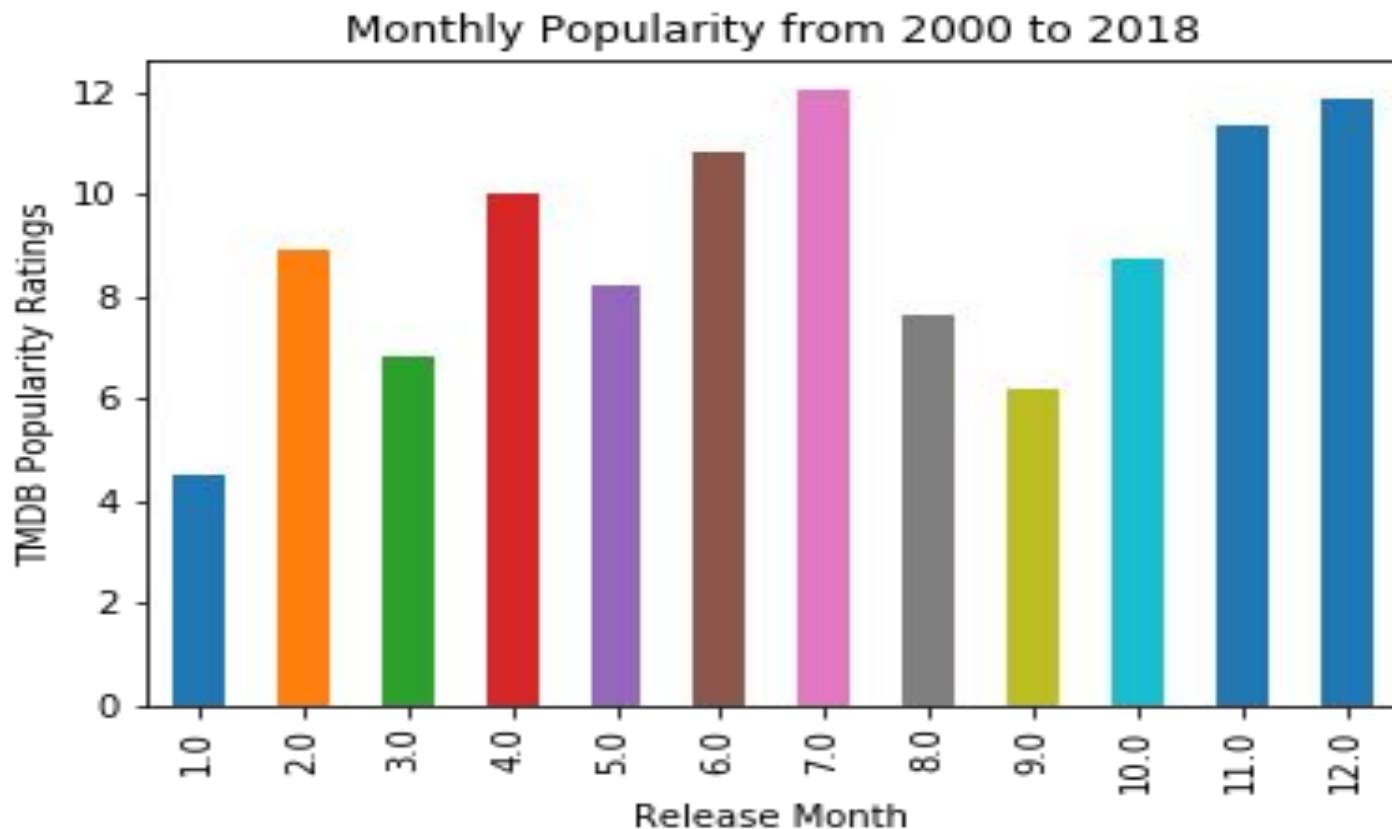
        movies_info['Box Office Info'] = info[6].text
```

# How we got the data - API

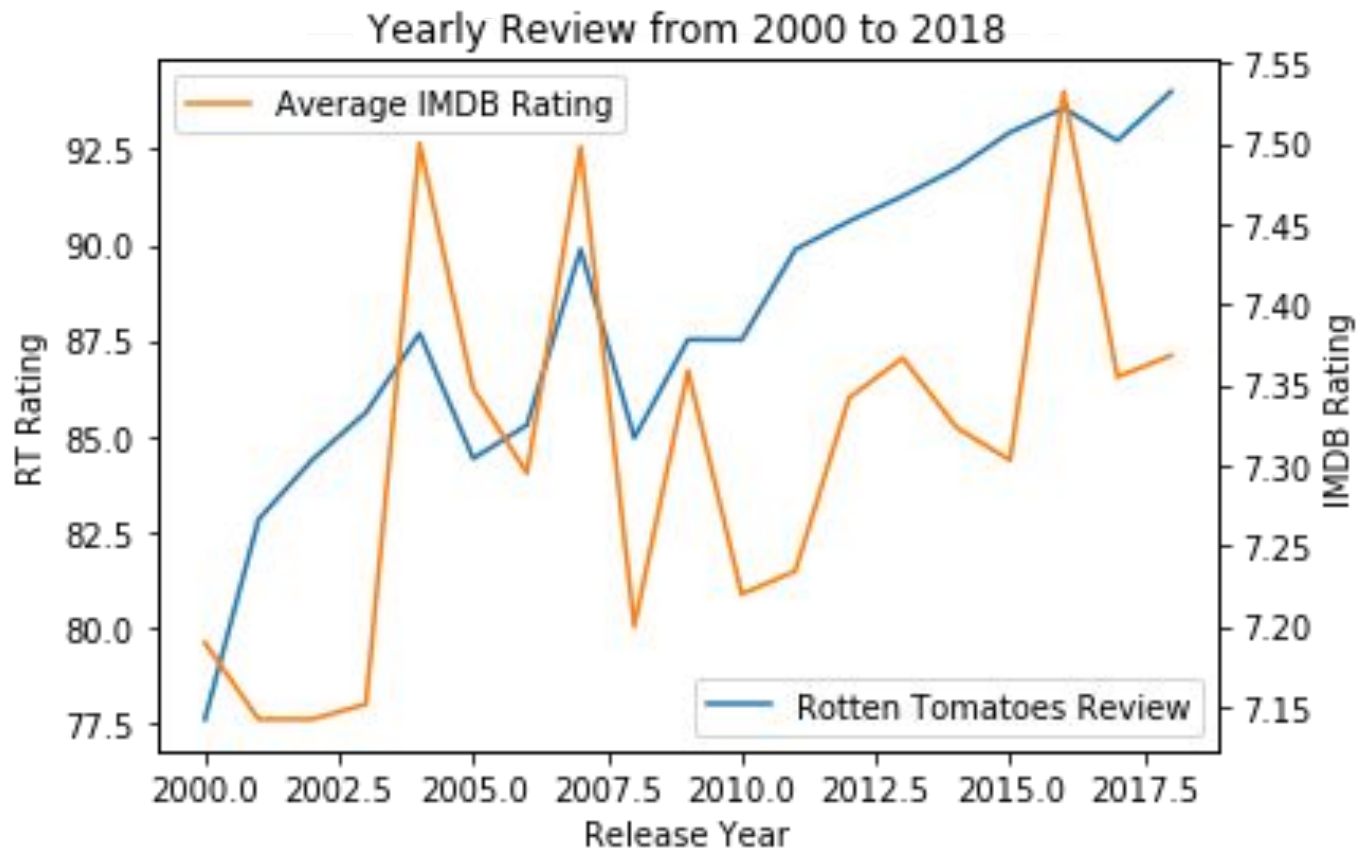
```
def tmdb_calls():
    movie_info = []
    counter = 0

    for movie in movies_search:
        try:
            if counter <= 40:
                URL =
                'https://api.themoviedb.org/3/search/movie?api_key={}&query={}'.format(access_token_tmdb, movie)
                response = requests.get(URL)
                movie_info.append(response.json())
                counter += 1
            else:
                print("Rate Limit Reached")
                time.sleep(10)
                counter = 0
        except:
            pass
    return movie_info
```

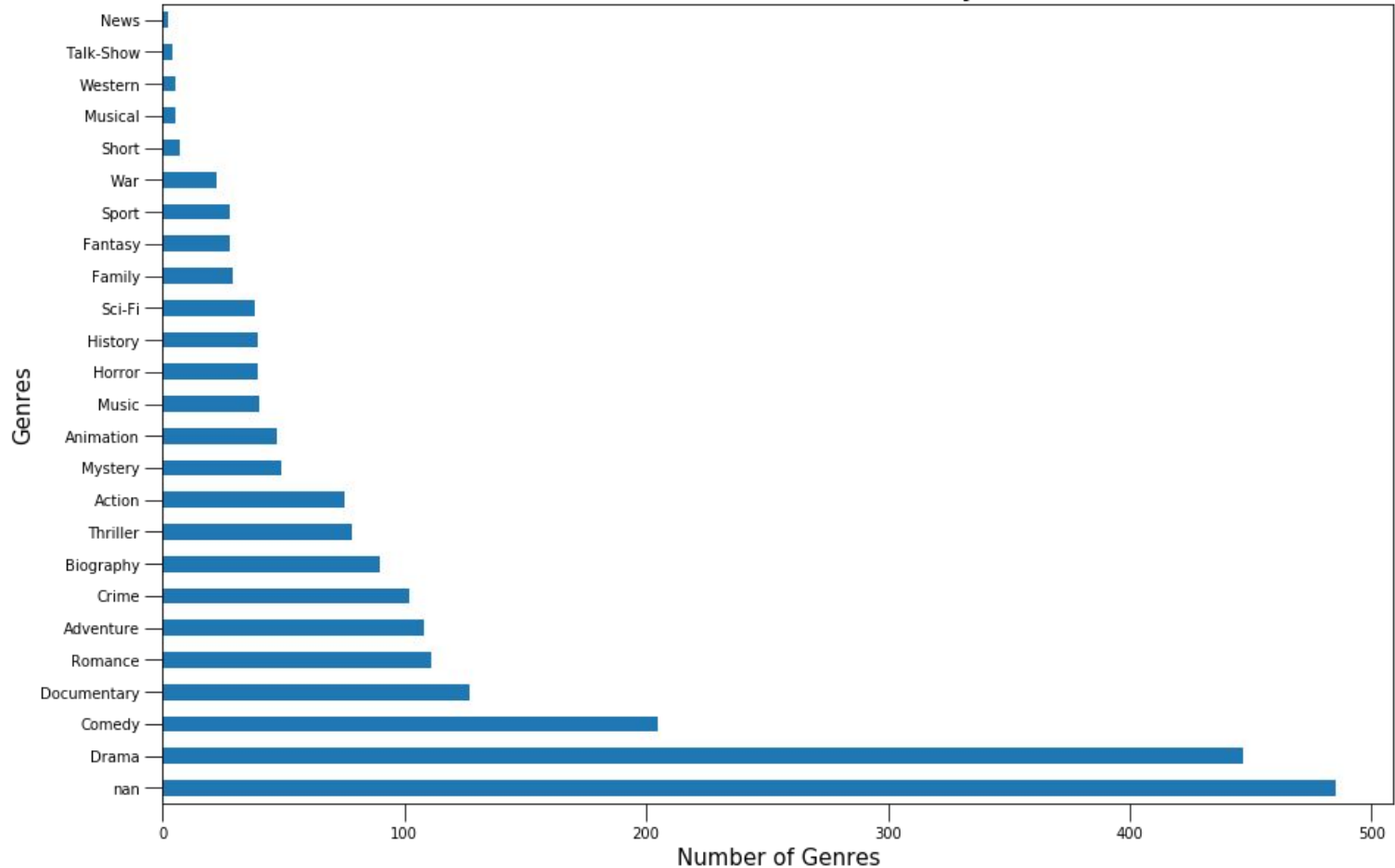
# A Closer Look Into Release Timings



# Everybody's A Critic

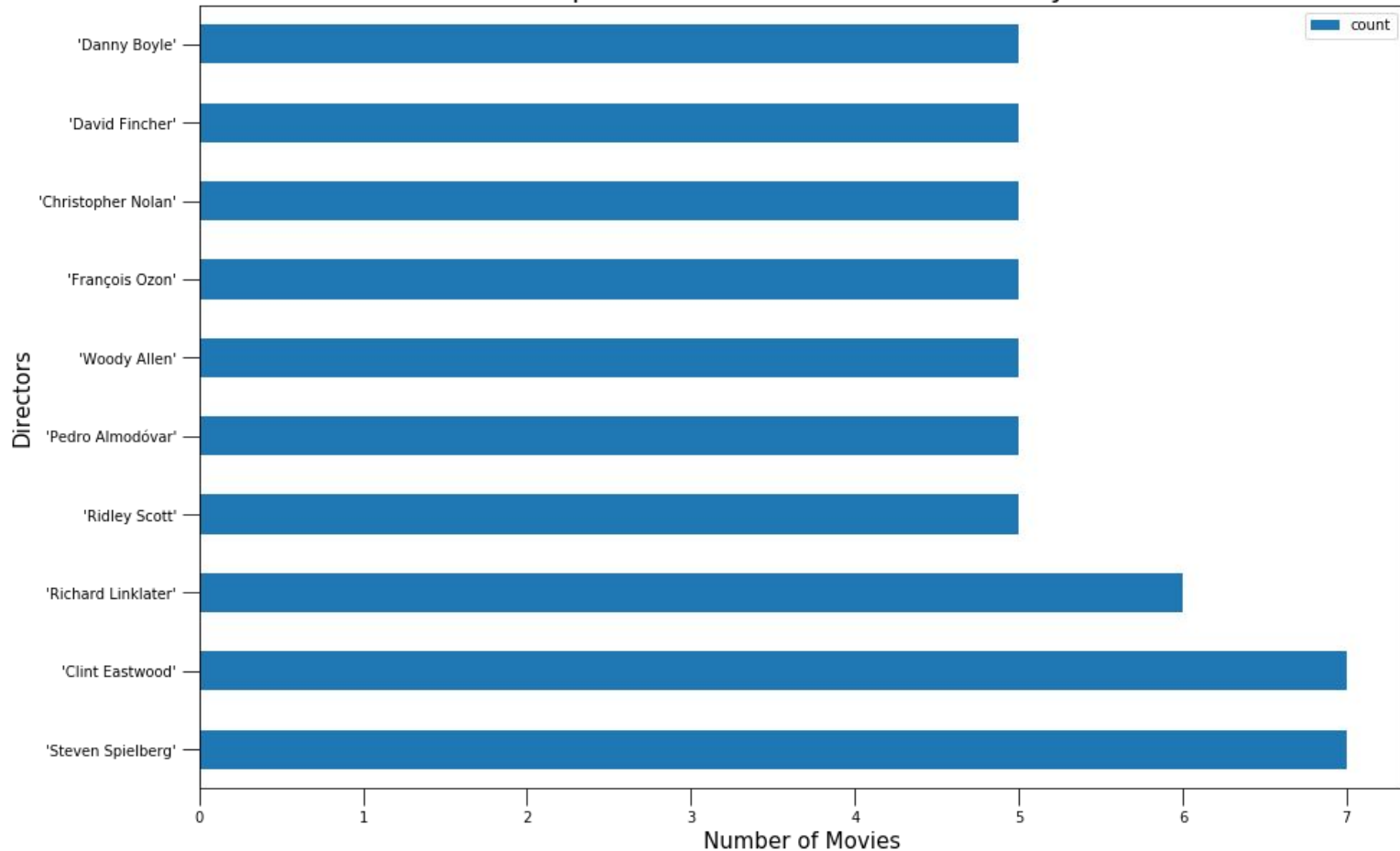


## Genres of the 21st Century





## Top 10 Directors of the 21st Century



# Next Steps

---

- Due to the budget of the engagement, we were unable to conduct additional analysis for more insights.
- If given additional resources, our next steps would include:
  - Assessing the correlation between the writers of the film with the revenue generated
  - Determining which director specializes in which type of genre
  - Determine the revenue per minute of film to determine how big of a scale you should be considering when making your next movie

# Sources Of Data

- Rotten Tomatoes: <https://www.rottentomatoes.com/top/bestofrt/?year=2018>
  - Used for Web Scraping
- The Movie Database: <https://developers.themoviedb.org/3/getting-started/introduction>
  - Used for API calls
- The Open Movie Database: <http://www.omdbapi.com/>
  - Attempted for API calls
- IMDB Public Datasets: <https://www.imdb.com/interfaces/>