Automated Insights Intern Evaluation, Data Science

Brian Clee, NC State University bpclee@ncsu.edu

Question 1

To determine what the five most popular genres in the movie dataset were, I first had to assume a definition of popularity. My initial idea was to use the box office earnings from the films to help determine popularity, since the highest grossing genres could be interpreted as the most popular. However, after inspecting the dataset I found that only some of the entries had associated box office scores. Because of this, I assumed a simple definition of popularity based on the number of occurrences of a genre in the dataset, thus the most popular genre is the one with the most associated films.

After parsing the dataset and counting every occurrence of each individual genre, I found that the five most popular genres (with their associated counts) were:

1. Drama: 19134
2. Comedy: 10467
3. Romance Film: 6666
4. Thriller: 6530
5. Action: 5868

Question 2

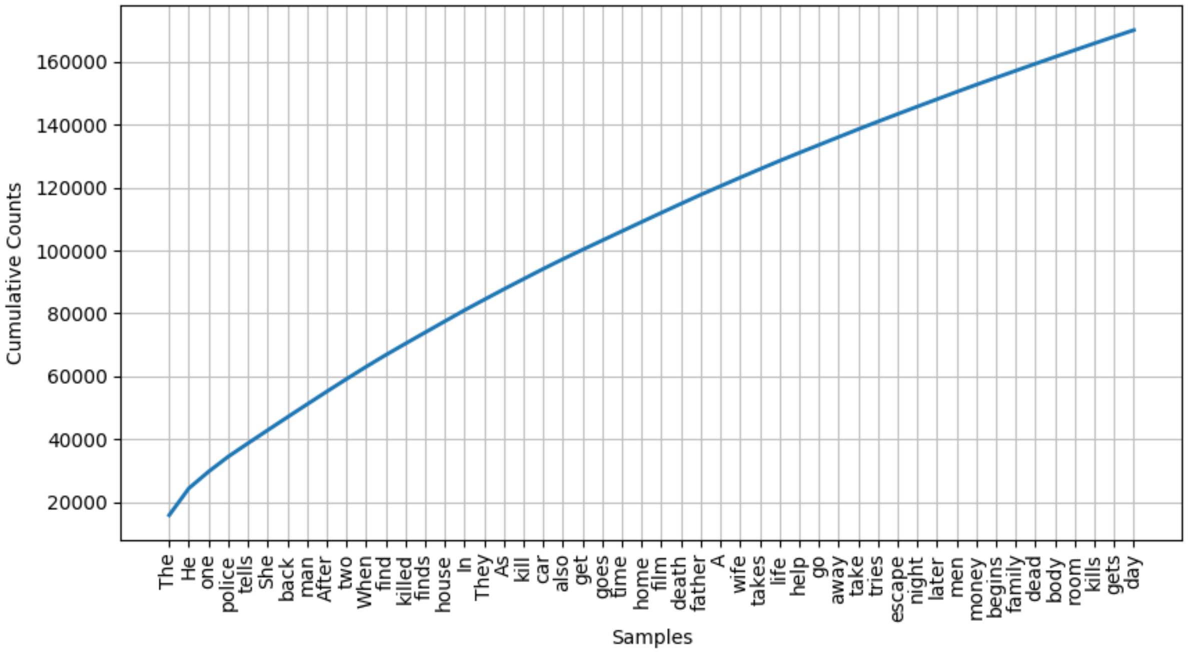
To find out which words are characteristic of the movie summaries of the top genres, I wanted to explore what words were unique to each of the genre's summaries. Investigating these unique word sets could give a good insight into the defining characteristics of each genre's summaries.

To approach finding the unique word sets for each genre, I first had to perform some natural language processing. I chose to use the Natural Language Toolkit (NLTK)[1], as I had prior experience using it for similar problems in the past. Using NLTK, I parsed through the movies dataset, and for each film that matched any of the top five genres I tokenized its summary text. After parsing the entire dataset, I had generated lists of tokens for each of the top genre's summaries, and could then perform further analysis to find the unique sets.

Rather than consider all tokens for each genre, I decided to focus on the top 50 most frequent tokens from each genre. Figure 1 below shows a frequency distribution for the Thriller genre's top 50 tokens. As shown in Figure 1, many of the most frequent words for the Thriller genre are common words like “the, he, she, go, etc.”, further motivating my search for unique word sets for each genre.

[1] http://www.nltk.org

Figure 1: Frequency distribution of the Thriller genre's 50 most common summary tokens.



With these frequent sets, I then determined the common set of words shared between the top 50 words of the top five genres. This common set contained 59 words, and from this set I was able to determine the unique set of words for each top genre. These unique sets are shown below in Table 1, and reveal interesting characteristics for each genre. For instance, the Romance Film genre has words one might expect like "relationship, meets, asks, leaves, together, married", while the Thriller genre has more ominous words like "begins, dead, body, room". Finally, another insight I gained from these unique sets was that Drama was the only genre with "story" in it's summary text corpus.

Table 1: Unique word sets for the top five genres.

Genre	**Unique Words**
Drama	daughter, story
Comedy	make, school, next
Romance Film	relationship, meets, asks, leaves, together, married
Thriller	begins, dead, body, room
Action	fight, group, gang

For future investigations into this problem, I would next investigate the collection of bigrams and collocations for each genre. Based on the unique sets in Table 1, I suspect "dead body" would be one such collocation for the Thriller genre.

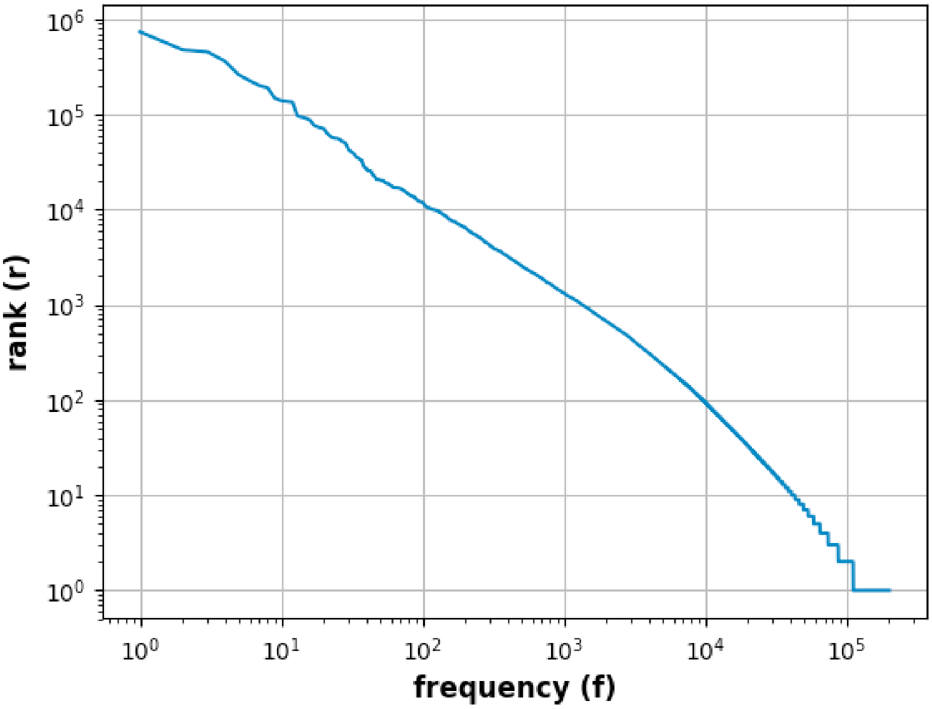
Question 3

Zipf's law is a mathematical property that has been found to be associated with many different types of data. In particular, Zipf's law is well known to occur in natural English. The law states that in any given text corpus, the word which occurs most frequently in that corpus will occur approximately twice as often as the second most frequent word. This continues as the most frequent word will in turn occur three times as often as the third most frequent word, and so on and so forth.

To find out if the corpus of movie summaries exhibits this property, I first had to parse the movie dataset and tokenize each movie's summary using NLTK. After tokenizing each summary, the tokens were added to a list which eventually contained tokens of every movie's summary text. From this list I was then able to create a frequency distribution to find out how often each word occurs in the text.

Finally, from this frequency distribution I generated Figure 2 shown below, which is a log-log plot of the summary corpus's rank versus frequency of all words. As Figure 2 shows, the movie summaries corpus does exhibit properties of Zipf's law, as the trend follows a near linear negative slope.

Figure 2: Zipf's distribution of all summary tokens' rank versus frequency.


rank (r)
10^6
10^5
10^4
10^3
10^2
10^1
10^0
10^0
10^1
10^2
10^3
10^4
10^5
frequency (f)