**Recommendation System Collaborative Filtering with Jaccard Similarity**

Aviel Leonardo Wijaya - 2201766173

Chandra Tan - 2201762931

Kevin Orlando Sutanto - 2201760945

Muhamad Daffa Mennawi - 2201810943

Devinca Limto - 2201758000

**Introduction**

While working on natural language models for recommendation system, there are questions that have been frequently asked “How similar are these two words?”, “How similar are these two sentences?”, “How similar are these two documents?”, “How similar are these two movies”. In this project, we will look more into techniques for sentence or document similarity.

One of the techniques in finding similarity is the Jaccard similarity index (sometimes called the Jaccard similarity coefficient) compares members for two sets to see which members are shared and which are distinct. It is a measure of similarity for the two sets of data, with a range from 0% to 100%. The higher the percentage, the more similar the two populations. Although it is easy to interpret, it is extremely sensitive to small samples sizes and may give erroneous results, especially with very small samples or data sets with missing observations. Other than Jaccard similarity there is Cosine similarity, Cosine similarity is a metric used to measure how similar the documents are irrespective of their size. Mathematically, it measures the cosine of the angle between two vectors projected in a multi-dimensional space. The cosine similarity is advantageous because even if the two similar documents are far apart by the Euclidean distance (due to the size of the document), chances are they may still be oriented closer together. The smaller the angle, higher the cosine similarity. Basically, this is the two most used techniques in finding similarity.

The main difference between these two is in cosine similarity, the number of common attributes is divided by the product of A and B's distance from zero. Whereas in Jaccard Similarity, the number of common attributes is divided by the number of attributes that exists in at least one of the two objects.

In this project, we will use Jaccard similarity. Basically, Jaccard similarity is good for cases where duplication does not matter, cosine similarity is good for cases where duplication matters while analyzing text similarity. For two product descriptions, it will be better to use Jaccard similarity as repetition of a word does not reduce their similarity. Based on our dataset where we will find the similarity between movies, it is better to use the Jaccard similarity due to the result will not be affected by duplication matter.

**Method**

There are two data that we used, which are the movie list, and the rating from users for some movies.

In this project, first we need to preprocess the data, then we could make the recommended system. For preprocessing the data, we used the *pandas* library from python to make it easier for us to preprocess the data. First step that is to remove duplicated movies. After we remove the duplicated movies, we split the movie genres to several columns with numerical values 0 or 1. And then drop useless columns that will not be used in the future when we create the recommendation System.

After we preprocessed the data, we could use the data for creating the recommendation system. main method that’s used in this project is the Jaccard Similarity. The module that we used for the recommendation is from the *scikit-learn* and *pandas* to process the objective of the project. The Jaccard Similarity is used to for item-based recommendation calculates the similarity between two movies by taking the rating by the name of the two movies and then compares it. After we got the similarity of each movies with the movies that have been picked. Then we show the results of the most similar based by rating, and with the name or movie have been chosen.

Other recommendation we have tested later, we find the recommendation of the movie based on name, rating and the genre of the movie. Basically, the first step is we choose the movie based on the rating, after that we choose the most based by the result of the genre.

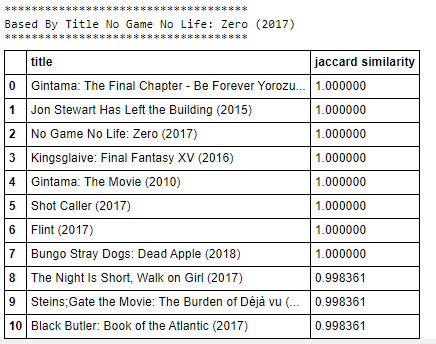
Below is the formula of Jaccard Similarity

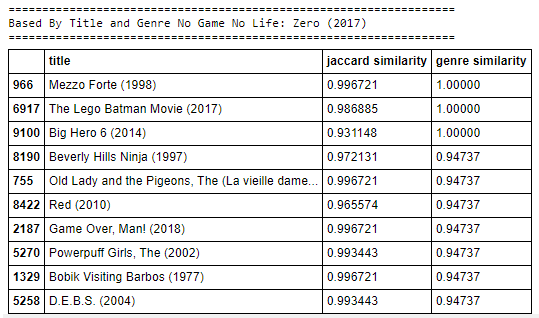


Where 𝑛𝐴 is the set of users who have rated movie A and 𝑛𝐵 is the set of users who have rated movie B.

**Results**

Having completed all steps to create the movie recommendation system, we obtained the recommended movies from the systems. We manually checked the movies if it were related to the pivot movie, most of the recommended movies were close.





These were the 10 top best recommended movies; first list was by title and second list were by title and genre. In case we get more column later, we could add more recommending system by those categories.

**Conclusion**

Jaccard Similarity is pretty accurate to be used in collaborative filtering and recommendation system. After we implemented the system, it can be seen that the achieved result is very accurate. We take a movie’s title and genre into consideration and the system manages to produce good quality recommendation. This system can be used very well to implement a search feature inside of an application where the search result will be changed according to the system’s recommendation.

We have also previously implemented the same concept using Pearson Correlation, but the result seems to be inferior to that of Jaccard Similarity by substantial margin. Since Jaccard Similarity focuses solely on the qualitative value of the data, it is concluded that for movie recommendation using title and genre, it is more important to look at the collaborative aspect rather than the quantitative aspect of the data such as when recommending a movie based on its genre.

For consideration in future improvement, the system proposed here takes a movie’s title and genre into account without counting a user’s rating. Due to this, the system might recommend some movies which has the highest Jaccard Similarity value but does not really fit a user’s taste.

Other than that, the system also suffers from lack of data. This became a huge problem solely because Jaccard Similarity is very sensitive to small sample size. Other consideration is if the user’s rating were also to be taken into account for the recommendation system, the lack of data will also heavily affect the outcome.

**References**

<https://en.wikipedia.org/wiki/Collaborative_filtering>

<https://iopscience.iop.org/article/10.1088/1742-6596/1362/1/012130/pdf>

<https://github.com/berkurka/MovieRecommender/tree/master/data>

<https://stackoverflow.com/questions/37003272/how-to-compute-jaccard-similarity-from-a-pandas-dataframe>