



# ANIME RECOMMENDER SYSTEM

javier alcantara  
comm 599

# CONTEXT & PROBLEM

Anime popularity grows!

- Global demand increased by 118% (2020-2022)
- US - highest overseas demand for anime (56M fans)
- Half of all Netflix users engage with anime

How can we keep  
anime fans engaged  
with anime?

# METHODOLOGY - Data

MyAnimeList Dataset 2023, Kaggle

user-score-2023.csv  
(sampled)

- user ids
- anime ids
- ratings

anime-data-set-2023.csv

- user ids
- anime ids
- titles

dataset -> <https://www.kaggle.com/datasets/dbdmobile/myanimelist-dataset>

# METHODOLOGY

Content-based Model  
using K-Nearest Neighbors  
(cosine similarity)

data cleaning,  
sampling

→ data  
pre-processing

→ data  
modeling

```
import pandas as pd  
import numpy as np
```

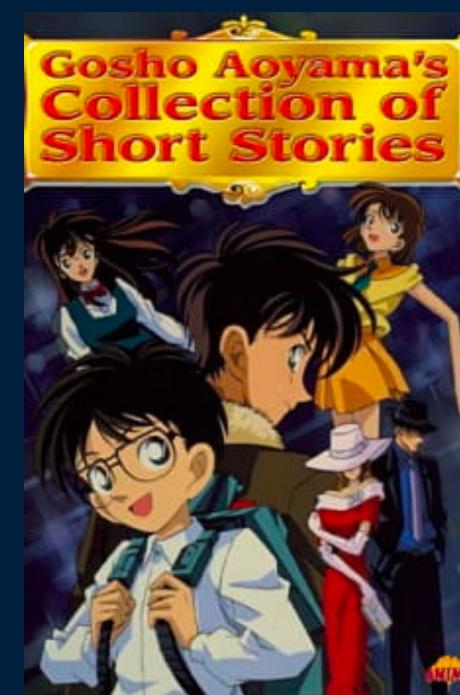
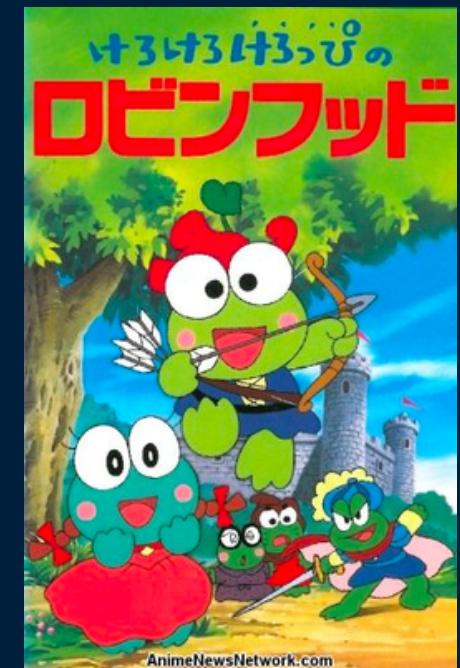
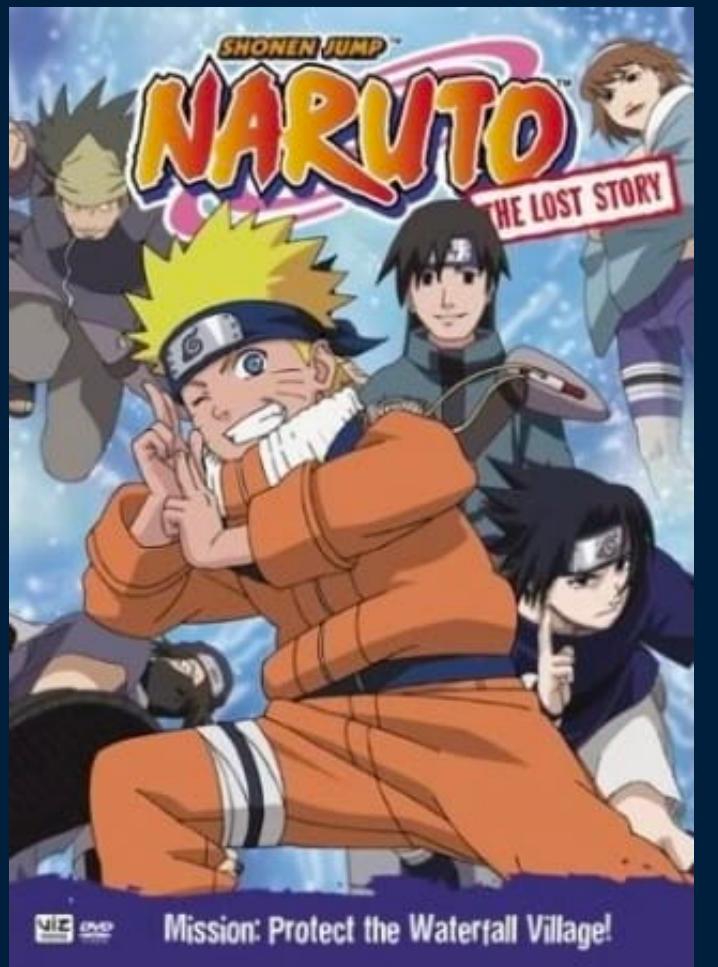
```
import scipy.stats  
from scipy import sparse  
from scipy.sparse import csr_matrix
```

```
from sklearn.neighbors import NearestNeighbors
```

```
title_lkp = dict(zip(anime.anime_id.values, anime.Name.values))
index_lkp = dict(zip(range(len(ua_matrix)), anime.anime_id.values))

def get_similar_anime(anime, data, index_lkp, model, title_lkp = None, n = 10):
    ## input to this function is the anime and number of top similar anime you want.
    if type(anime) == str:
        anime = [key for key, value in title_lkp.items() if value == anime][0]
    print(anime)
    input = data[data.index == anime].values.reshape(1,-1)
    print(input)
    distances, indices = model.kneighbors(input, n_neighbors=n+1)
    print(indices)
    results = [index_lkp[indices[0][i]] for i in range(0,len(distances[0]))]
    if title_lkp is not None:
        results = [title_lkp[x] for x in results]
    return(results)

get_similar_anime('Naruto', ua_matrix, index_lkp, knn_model, title_lkp)
```



# Next Steps

A

Rethink Data  
Cleaning,  
Processing

C

Consider  
Hyperparameter  
Fine-tuning

B

Add Content-based Methods  
+  
Matrix Factorization



thank you