Final Report: Movie Recommendation

by Grant Doan, Prayash Joshi, Reagan Orth, Ved Patel

Challenges and limitations

Cold Start:

- Initially, websites like IMDB have no prior data on a user

Data Sparsity:

- Difficult to find users that have rated the same movies

Scalability:

- With the large amount of data being used on popular websites, inaccurate results are inevitable

Filter Bubble:

- There are strongly correlated groups of movies, and recommendations don't often bridge between groups

Sentiment Consistency and Ambiguity:

- Discussed more in the NLP section

Our Approach: Overview

Web Scraped Datasets:

- Review Dataset
- Movie Dataset

Natural Language Processing:

- Use review dataset
- Find specific film elements that are preferred by users

Graph Neural Network:

- Use movie dataset
- Use a Graph Neural Network to learn similarity mappings for movies

Recommendation Algorithm

- Use different metrics to recommend movies

Dataset

Review Dataset
- Reviews
- Ratings
- Usefulness (proportion of users that found the review helpful)

	the	movie	satire	cat	superhero	imperceptible	
The Menu	3252.0	640.0	68.0	1.0	NaN	NaN	
Antman	3465.0	774.0	NaN	NaN	20.0	NaN	
Puss In Boots	3157.0	872.0	NaN	54.0	NaN	1.0	
Totals	9874.0	2286.0	68.0	55.0	20.0	1.0	
	I	Great	Disney	Horror	Fiennes	Marvel	Dreamworks
The Menu	571.0	2.0	3.0	2.0	76.0	1.0	NaN
Antman	648.0	4.0	36.0	NaN	NaN	144.0	NaN
Puss In Boots	642.0	3.0	29.0	NaN	NaN	NaN	49.0
Totals	1861.0	9.0	68.0	2.0	76.0	145.0	49.0

Movie Dataset
- Scraped IMDb's top 1000 movies
- Title
- Description
- Genre(s)
- Director(s)
- Up to 4 Actors

title	Action	Adventure	Crime	Drama	John Huston	Martin Scorsese	Al Pacino	Bette Davis	Brad Pitt
Life Is Beautiful	False	False	False	True	False	False	False	False	False
It's a Wonderful Life	False	False	False	True	False	False	False	False	False
Seven Samurai	True	False	False	True	False	False	False	False	False
Harakiri	True	False	False	True	False	False	False	False	False
Parasite	False	False	False	True	False	False	False	False	False
The Departed	False	False	True	True	False	True	False	False	False
Whiplash	False	False	False	True	False	False	False	False	False
Gladiator	True	True	False	True	False	False	False	False	False
Back to the Future	False	True	False	False	False	False	False	False	False
The Prestige	False	False	False	True	False	False	False	False	False
Alien	False	False	False	False	False	False	False	False	False
Léon: The Professional	True	False	True	True	False	False	False	False	False
The Lion King	False	True	False	True	False	False	False	False	False

Natural Language Processing and Sentiment Analysis

Data Cleaning

- Removal of HTML tagging from webscraper
- Cleaning up punctuation and creating a sparse matrix on the reviews
- DF/TDF to remove insignificant words
    - Give special attention to names mentioned in reviews

Sentiment Analysis

- Pretrained BERT for Aspect Based Sentiment Analysis finetuned to movie reviews
- Returns the probability estimate of a positive or negative review

Feature Extraction

- Use of Spacy to find specifically mentioned people
- Similarity scores to find movies, genres, and other film elements.

TF Bert For Sequence Classification training and metrics

Batchsize : 32
Training/Validation Split : 80/20 on ~60000 movie reviews

```
Epoch 1/3
600/600 [==============================] - 522s 760ms/step - loss: 0.3662 - accuracy: 0.8410 - val_loss: 0.3212 - val_accuracy: 0.8639
Epoch 2/3
600/600 [==============================] - 418s 698ms/step - loss: 0.3004 - accuracy: 0.8732 - val_loss: 0.2754 - val_accuracy: 0.8847
Epoch 3/3
600/600 [==============================] - 408s 681ms/step - loss: 0.2657 - accuracy: 0.8922 - val_loss: 0.2879 - val_accuracy: 0.8898
<keras.callbacks.History at 0x7fed86b47cd0>
```

NLP Obstacles

Review Consistency and Data Clarity

- Does the review reflect the rating?
- This inconsistency corrupts the training set
- Yields deceptive results

Reviews with no ratings

Ratings with no reviews

Similarity Learning

Create binary feature matrix
- 0s and 1s indicating present/absent features
- All film genres, directors, actors, and decades
- 2800 features in total

Train Graph Neural Network
- 80/10/10 train/validation/test
- Look at 3 neighbors at a time
- Two SAGEConv layers
- Update with custom similarity loss function

Obtain similarity matrix
- Run all movies through GNN

Export similarities to CSV for later analysis

Choosing Recommendations

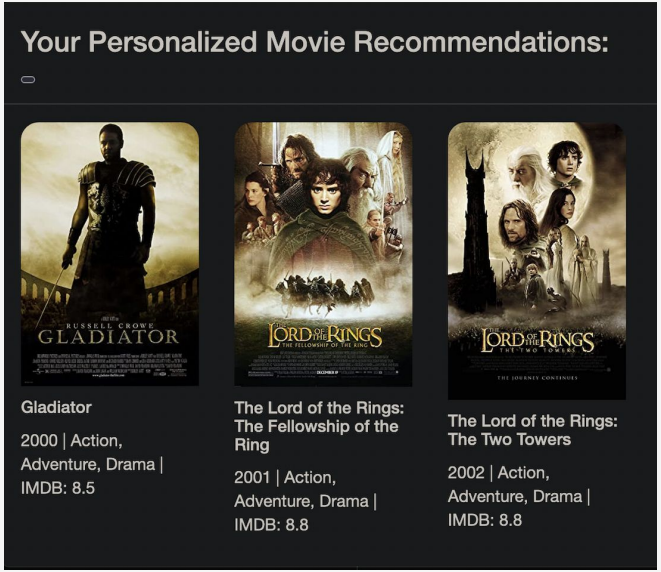
Recommendations often suffer from being too similar. Therefore, we use different selection algorithms.

Good rating/sentiment

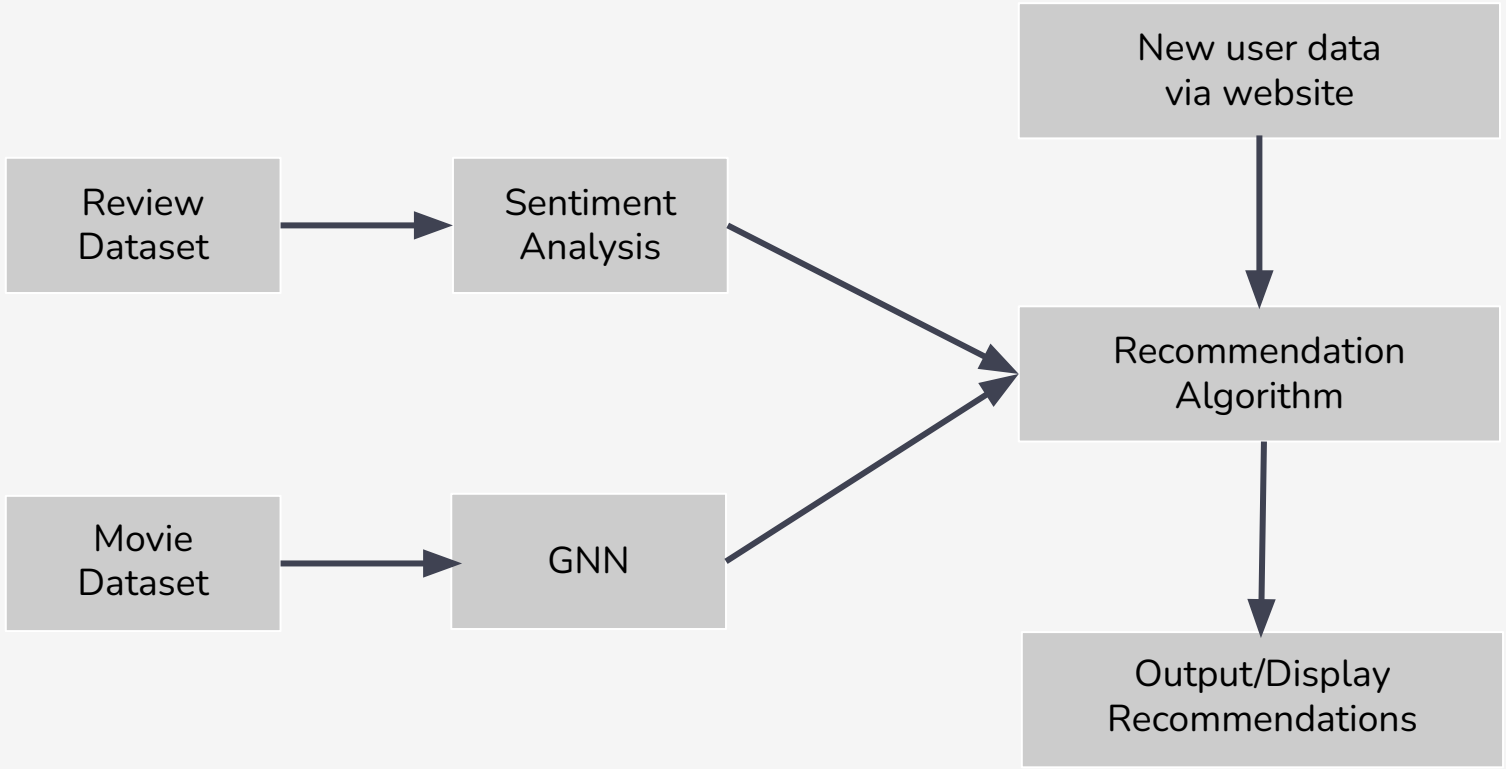
- Random of top n most similar to one movie
- Random of top n most similar to all movies
- Random of similar movies given genre
- Random of similar movies given director
- Random of similar movies given actor

Bad rating/sentiment

- No direct recommendation
- Inspect other recommendations for "bad" features



Pipeline diagram



Evaluation

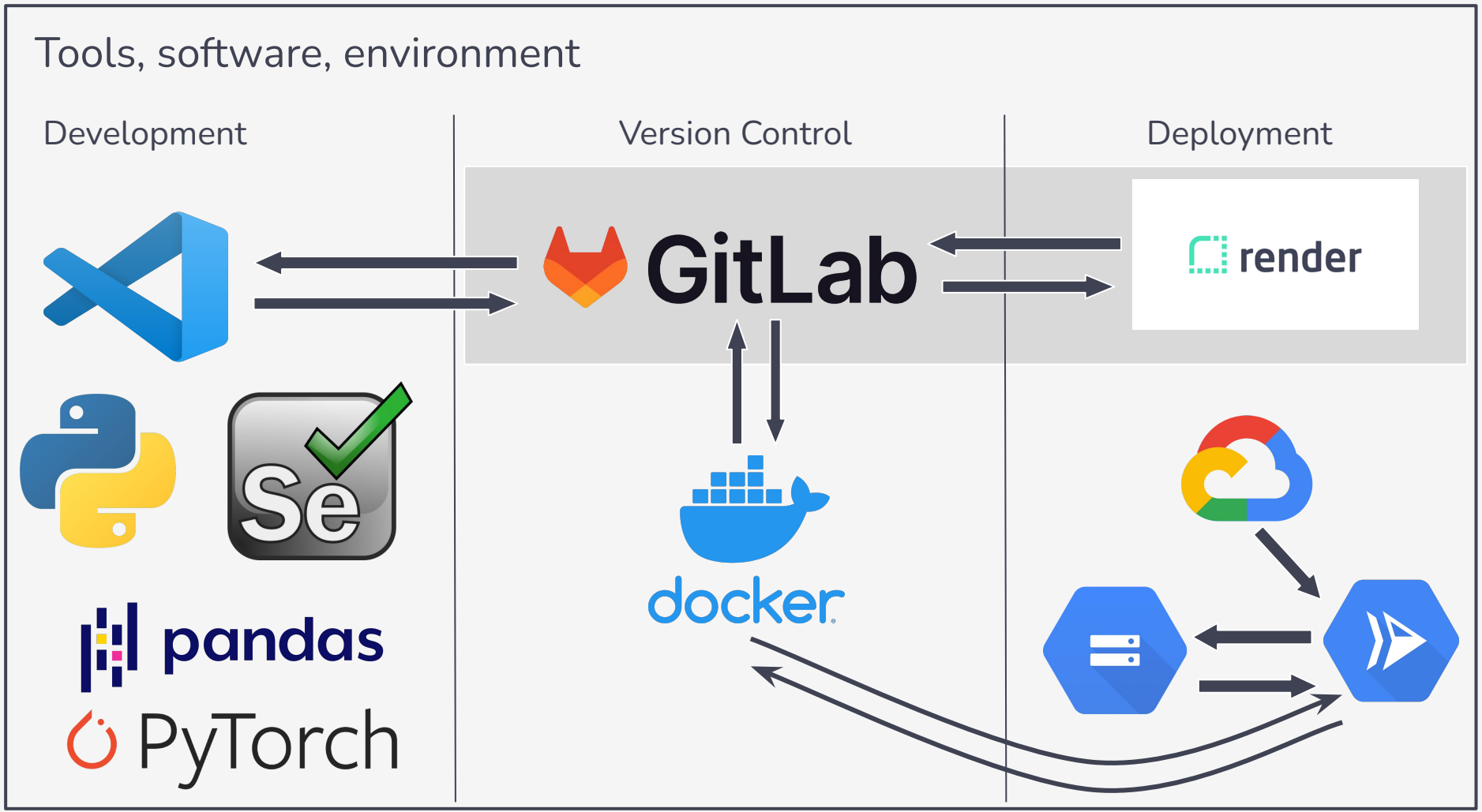
Obviously, there is no objective standard of evaluation, which is both freeing and limiting

However, certain similarities are obvious

- Sequels
- Remakes
- Repeated cast/crew
- Genres
- Decades

In addition, we performed some user testing

- IMDb users from review dataset may have rated several included films
- We ran several tests with us and our friends
    - If users had seen the movie, they gave feedback
    - If users had not seen the movie, asked them to see it where possible


Tools, software, environment
Development
Version Control
Deployment
GitLab
render
Se
pandas
PyTorch
docker

UI (User Interface)

DEMO

Tasks breakdown and team members contributions

Prayash: Full-Stack, Cloud Hosting & Database, dynamic web app,

Reagan: Data cleaning, GNN, recommendation algorithms

Grant: Scraping user reviews, sentiment analysis/feature extraction

Ved: Scraping movie dataset, website graphics, project poster

Discussion

Any Questions?

References

https://www.researchgate.net/figure/An-undirected-graph-with-7-nodes-and-7-edges_fig3_265428782