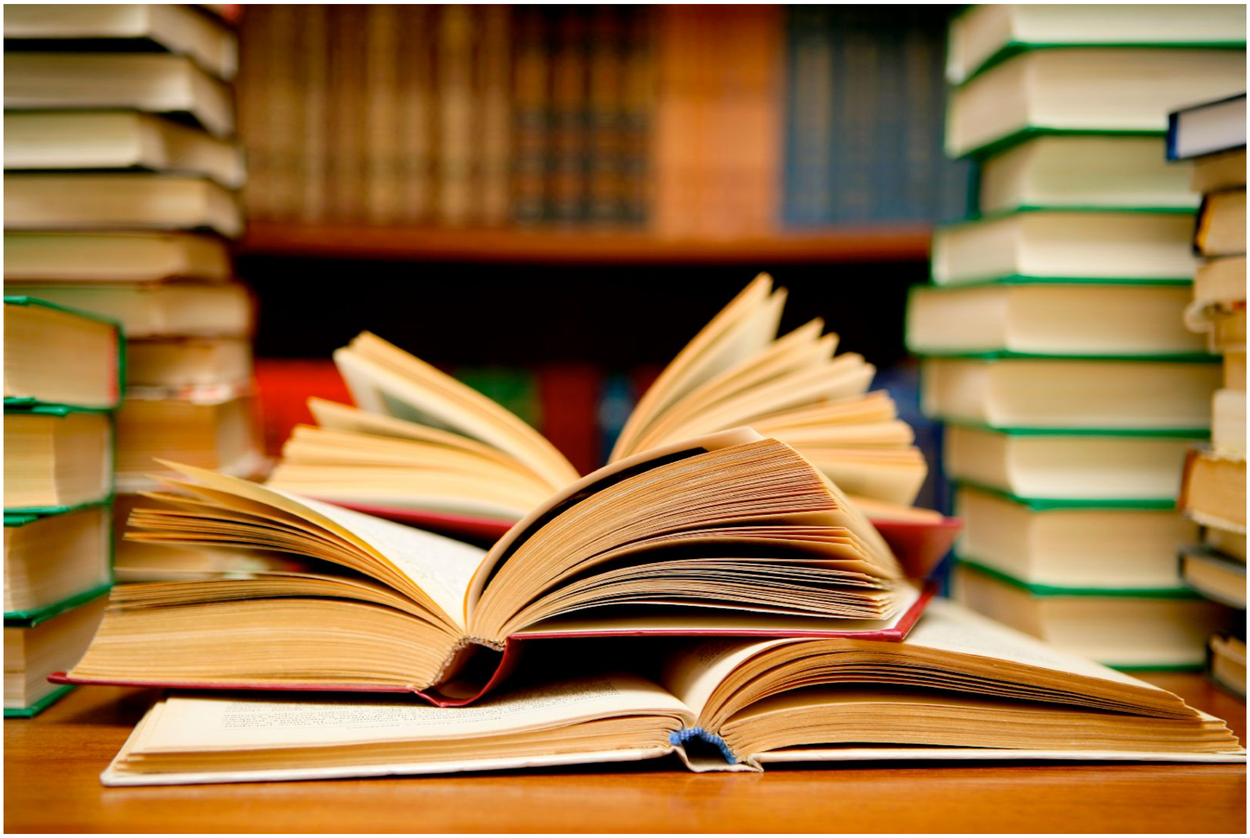


Personalized Book Recommendations

04.12.2019

Emily Strong
Manoj Karru
Nishant Gandhi

Overview

Many companies face the challenge of providing personalized recommendations to users based only on the users' interactions with content on the site. A classic approach to this problem is collaborative filtering, but this introduces what is known as the cold start problem: when decisions are based solely on interaction history, how do you provide recommendations to new users, and how do you recommend new content?

Deep learning and reinforcement learning techniques provide an alternative approach to this type of recommendation problem. In this project we will explore collaborative deep learning and deep reinforcement learning and determine which is better suited to personalized recommendations.

Goals

To determine whether collaborative deep learning or deep reinforcement learning has a stronger success in personalized recommendations and implement a recommendation pipeline with a simple web app.

Algorithms

Collaborative deep learning uses stacked autoencoders to encode and then reconstruct a user's preferences and identify items to recommend. The Collaborative Denoising Auto-Encoders framework enhances this by injecting latent user features.

Algorithm 1 Learning algorithm for CDAE

Initialize parameters with random values.
$iter \leftarrow 0$
while $iter < maxIter$ or error on validation set decreases **do**
  for all $u \in \mathcal{U}$ **do**
    Sample $\tilde{\boldsymbol{y}}_u \sim p(\tilde{\boldsymbol{y}}_u | \boldsymbol{y}_u)$ using Equation 9.
    Compute $\boldsymbol{z}_u$ using Equation 10.
    Sample negative samples $\mathcal{S}_u \subset \bar{\mathcal{O}}_u$.
    for all $i \in \mathcal{O}_u \cup \mathcal{S}_u$ **do**
      Update $\boldsymbol{W}'_i$ and b'_i using Equation 14, 15 and 20.
    end for
    Compute $\frac{\partial \ell}{\partial \boldsymbol{z}_u}$ using Equation 16.
    for all $j \in \{j | j \in \mathcal{I}\ and\ \tilde{y}_{uj} > 0\}$ **do**
      Update $\boldsymbol{W}_j$ using Equation 17 and 20.
    end for
    Update $\boldsymbol{V}_u$ using Equation 18 and 20.
    Update $\boldsymbol{b}$ using Equation 19 and 20.
  end for
  $iter \leftarrow iter + 1$
end while

Deep reinforcement learning approaches to recommendation model the problem as markov decision processes. They can then use context features of the user and content to drive deep Q-learning.

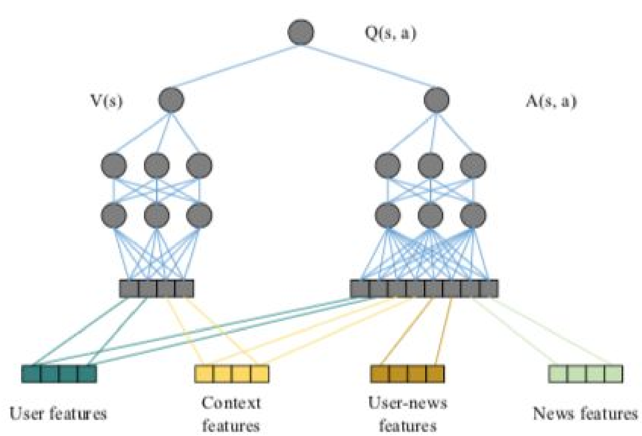


Figure 4: Q network

An example Q network for news recommendation from Zheng et al (2018).

Model Evaluation

We will evaluate each model based on replay with historic data, a method of evaluation in which only those recommendations that have historic ratings are scored. In this method, the model makes recommendation for an historic test set, and the top-k recommendations are evaluated for mean average precision and recall. We will look at the top 10 recommendations, similar to what might be displayed for a website user.

Data

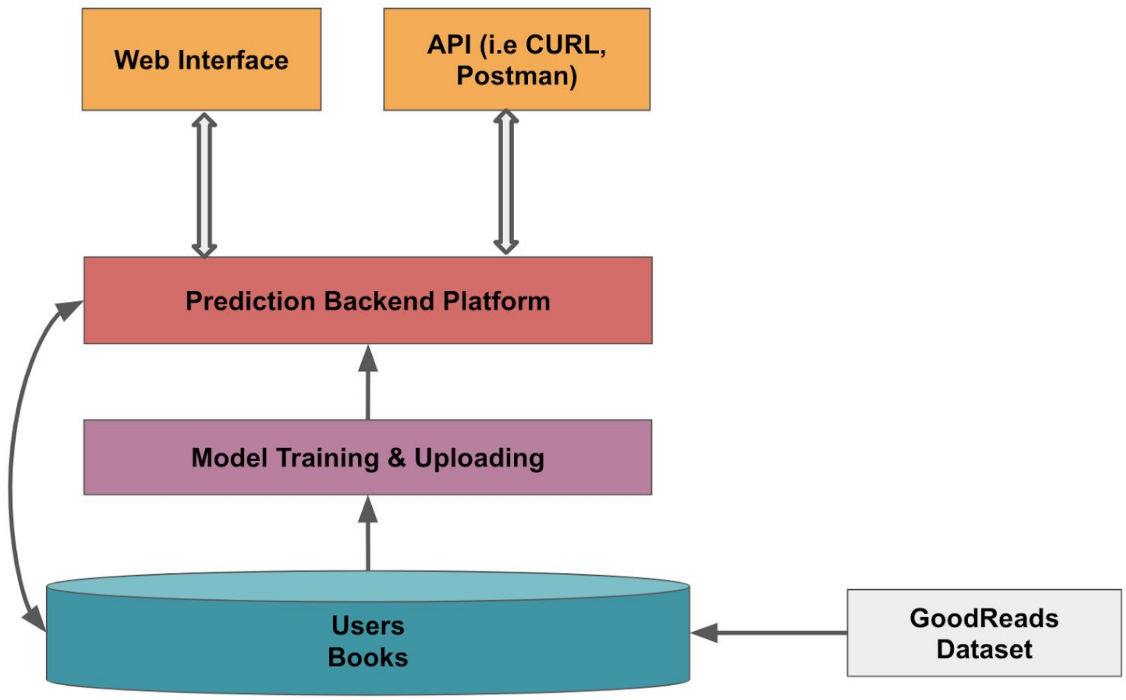
Data set: https://sites.google.com/eng.ucsd.edu/ucsdbookgraph/home

The Goodreads data set is scraped from public data on Goodreads.com. The full data set contains over 200 million records for approximately 876k users and 2.4 million books. Since it is so large and Goodreads segments its recommendations based on category, we will focus on a single category in our experiments of *Mystery, Thriller & Crime*.

Process Outline

1. Data Wrangling
2. Exploratory Data Analysis
3. Feature Engineering
    a. Text features of the books will be embedded using GloVe
    b. User-book interaction history will be used to generate additional features
4. Development of models and comparison of approaches
5. Create API for best model and deploy on cloud
6. Build web-app for user-specific recommendations

System Architecture



Milestones

Timeframe	Delivery
April 11-14	Exploratory Data Analysis & Feature Engineering
April 15-21	Model Building, Training, Tuning and Selection
April 22-24	Deployment of models on cloud and build web application
April 25-26	Documentation

Deployment Details

1. Language: Python
2. Deep Learning Framework: Keras
3. NLP Embedding Tool: GloVe
4. Cloud Tools/Platforms: AWS (Amazon WEb Services) EC2 or Google Cloud Platform
5. Application Framework: Flask;

Useful Links

Architecture Presentation: [Google Slides Link]

References

Mengting Wan, Julian McAuley, "Item Recommendation on Monotonic Behavior Chains", in Proc. of 2018 ACM Conference on Recommender Systems (RecSys'18), Vancouver, Canada, Oct. 2018.

Wu, Yao, et al. "Collaborative denoising auto-encoders for top-n recommender systems." *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*. ACM, 2016.

Zheng, Guanjie, et al. "DRN: A deep reinforcement learning framework for news recommendation." *Proceedings of the 2018 World Wide Web Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2018.