Machine Learning Engineer Nanodegree

Capstone Proposal

Lars Erik Bolstad
November 16, 2018

Book Recommendation Engine

Domain background

The goal of this project is to implement a recommendation engine for books.

Recommendation engines, or *Recommender systems*, are widely deployed to offer users recommended content or products. Such systems broadly fall into two categories depending on the model and algorithms used: *Collaborative* and *Content-based* filtering [1]. Collaborative filtering models produce recommendations based on a user's past behaviour and preferences, as well as those of other users exhibiting similar preferences. Content-based models are based on characteristics of the product or content in question to recommend items with similar properties. Many *hybrid* recommender systems combine these two approaches in various ways.

Problem Statement

Regardless of the model used a recommendation engine needs information about a given user's preferences in order to provide recommendations perceived as relevant and useful. Without such information we have what is known as the *cold-start problem*.
Our recommendation engine needs to be able to cope with this scenario, as well as the preferred case where it actually can access such data directly.

Datasets and inputs

The recommendation engine will be based on a Kaggle dataset containing user ratings for 10,000 "popular" books from Goodreads.. The dataset is available here: https://www.kaggle.com/zygmunt/goodbooks-10k/home. In this project I will use an updated

version of the dataset with duplicates removed and many more ratings (around 6 million) retrieved from this source: https://github.com/zygmuntz/goodbooks-10k.

The dataset contains limited data on each book, which means that a content-based filtering approach would need additional data from another source. The dataset does contain around 34,000 user-defined *tags* along with around 1 million combinations of these tags applied to books. An initial inspection of this data shows that the overall quality is low. Many of the user-defined tags have no relation to the books (e.g. the format, which year the book was read, etc). There are also many cases of different spellings of essentially the same tags, meaning substantial cleaning will be required to use this data in a content-based model.

In addition to this dataset I will make use of Goodreads APIs (https://www.goodreads.com/api). Recommendations for users who have a Goodreads account will be based on fetching their reading history using this API. I have personally used Goodreads actively since the service appeared almost 10 years ago and look forward to testing the end result in this project on my own account data!

Goodreads also offers API endpoints for retrieving more detailed data about a book, such as a back-cover synopsis, which could be used to enhance a content-based model. I will evaluate this approach as part of the project.

One challenge with this dataset worth noting right at the outset is that it contains a subset of the books in the Goodreads database. Being more than a year old it only contains books published before early 2017, for instance. An initial test based on fetching the list of books read from my own account via the API shows that only 98 of 200 books are present in the dataset. This is a sample of one, but it could mean that the input to the recommendation engine becomes more limited for users whose reading habits don't align well with the majority of readers.

Solution Statement

As mentioned above, the recommendation engine will need to handle two scenarios: Either we have access to a user's preferences (reading history, perhaps with a rating of each book) via Goodreads' APIs, or we will have to ask the user to indicate in some way what kind of books he or she likes to read. In the latter case the solution will be to first present the user with a number of books and ask for some kind of opinion (a rating, a thumbs-up or thumbs-down, or similar) and use this input to produce recommendations.

The dataset contains users' ratings on a scale from 1 to 5, and recommendations will be based on predicting the ratings for books that users have not read. The solution will be based on Singular Value Decomposition (SVD) [2], a matrix factorization technique that has become widely used in recommender systems. By creating a 2D *ratings matrix*, where each cell contains

the rating given by a user for a particular book, SVD can be used to produce matrices of reduced dimensionality that capture **latent factors** in the relationships between users and books. These matrices can then be used to *predict* ratings for new users and thereby produce recommendations.

The main challenge with this dataset is that the ratings matrix becomes very sparse. 10,000 books and around 50,000 books means we have a total of around 500 million ratings, but the dataset contains only around 10% of those values. For SVD to be used we will need to replace the missing rating values with actual numbers, a process known as **imputation**. I will try different strategies for imputing missing values and measure the performance of the recommendation engine for each one.

Benchmark model

The benchmark model will be based on imputing missing values with random ratings between 1 and 5. I will then expect all subsequent imputation strategies to score better than the benchmark model

Evaluation metrics

For evaluating the performance I will use the Root Mean Square Error, or **RMSE** metric. The original ratings will be split into a training matrix and a test matrix. I will then impute missing values in the training matrix and apply SVD to get the latent factor matrices, which when multiplied will produce a prediction matrix. The RMSE score will be calculated by comparing the predicted ratings with the original ratings in both the training and test matrices. The lower the RMSE the better.

Project design

The initial activity will consist of an analysis of the Goodreads dataset, including an exploration of the data, necessary data type conversions and an evaluation of missing values if any. Characteristics of the dataset will be visualized and examined with respect to suitability to the candidate machine learning models that may be applied. Cleaning and removal of data will be applied based on this analysis.

Also, an exploration of the Goodreads API will be conducted in order to determine which endpoints need to be called to extract a user's reading history as well as potentially other data that may serve as input to the machine learning models.

The main focus in this project will be on the imputation of missing values in the ratings matrix. I will try out different strategies and perform SVD on the resulting imputed ratings matrices, then use these to predict ratings and calculate the RMSE for each one.

The plan is to deploy the models as part of a web application that provide book recommendations both to users who have an active Goodreads account, and to those that don't.

[1] https://en.wikipedia.org/wiki/Recommender_system
[2] https://en.wikipedia.org/wiki/Singular_value_decomposition