Audrey Lin
DSCI549
HW 5

Book Recommendation System with Goodbooks-10k Dataset

1 Description: HW 1 recap

In HW 1, I proposed a book recommendation system that could take a book title a reader enjoyed as input, then recommend a small list of book titles to the reader as an output by analyzing user ratings of the input book and linking them to other books that users have also rated highly.

However, since that assignment, I have learned that book recommendation systems are typically classification problems. As such, my updated project proposal is a book recommendation system that classifies books as recommended or not recommended.

2 Dataset: Goodbooks-10k

The input dataset is Goodbooks-10k, representing 6,000,000 ratings from more than 50,000 users for the 10,000 most popular books on GoodReads, a social website for users to rate and recommend books. The key files of the dataset include five CSV files of tabular data, as well as an archive of 10,000 XML files of structured data (one XML file for each book). For the scope of this project, only one of the CSV files are needed: books.csv.

books.csv consists of 10,000 instances (books) and 23 features. Some features include book titles, authors, ISBN, publication year, language, and average ratings. However, since the average ratings will serve as the label for an instance, they will be excluded from the set of features.

To simplify this classification problem, the labels will be converted into binary (and disjoint) classes of recommended (positive label) or not recommended (negative label), whereby instances with average ratings of 3 or more (out of 5) are labeled as recommended and instances with average ratings less than 3 are labeled as not recommended.

3 Pre-processing

Besides converting labels (as described in the previous section), other pre-processing steps that are involved include feature selection (including features such as book titles, authors, ISBN, publication year, language; excluding features such as book ID, GoodReads ID, various other IDs, ISBN13, individual rating counts), verifying the accuracy of the data, and checking for any other errors or missing values.

Some features that would benefit from cleaning include book titles and language codes. For example, book titles in this dataset also include their series names -- denoted by enclosing parentheses following the book title -- which can be removed, i.e. “Harry Potter and the Sorcerer’s Stone (Harry Potter, #1).” On the other hand, this information could be used to create an additional feature such as whether a book is part of a series or what number book in the series it is. As for language codes, feature values require heavy standardization. Just for books published in English, language feature values include variations such as “en,” “eng,” “en-US,” “en-CA,” and more; there are some blanks as well. If selected to be used as a feature, book tags would require even more extensive cleaning including NLP tools and would benefit from being sorted into further categories (and thus, additional features), such as genres, target age groups, and author demographics.

4 Analysis: classification

The chosen data analysis task for my book recommendation system is classification, whereby book instances from the Goodbooks-10k dataset (input) are classified into binary classes of recommended or not recommended (output). Labeling occurs when users submit their book ratings, which is then calculated into a book’s average rating; as previously described, these labels are then converted into binary classes of recommended or not recommended. Features include book titles, authors, ISBN, publication year, and language. If user data were available, the book recommendation system could be personalized for individual users and include additional features such as user demographics and preferences as well as book tags.

The 10,000 book instances are partitioned into two distinct groups with no overlapping members in order to avoid contamination. Then the model is trained on one group (training set) and tested on the other group (test set). Finally, the instances are classified as recommended or not recommended.

One way to evaluate the classifier is by n-fold cross validation with 10 folds. This means that the 10,000 labeled instances are divided into 10 folds of equal size (1,000 instances per fold). The classifier is run 10 times, and each time the classifier is run, a different fold is used as the test set (1,000 instances per iteration) and the remaining 9 folds are used as the training set (9,000 instances per iteration). For example, in the first iteration, the classifier is trained on the training set, comprising of the second through tenth folds, then tested on the test set, comprising of the first fold; in the second iteration, the classifier is trained on the training set, comprising of the first and third through tenth folds, then tested on the test set, comprising of the second fold; etc. Each iteration gives an accuracy score, then all 10 scores are averaged to give the best estimate of the accuracy for the classifier.

If the accuracy is low, then it would not be ideal to take recommendations from this book recommendation system. Modifications would have to be made, such as but not limited to selecting different features, collecting higher quality data, or collecting a greater quantity of data.

5 Workflow

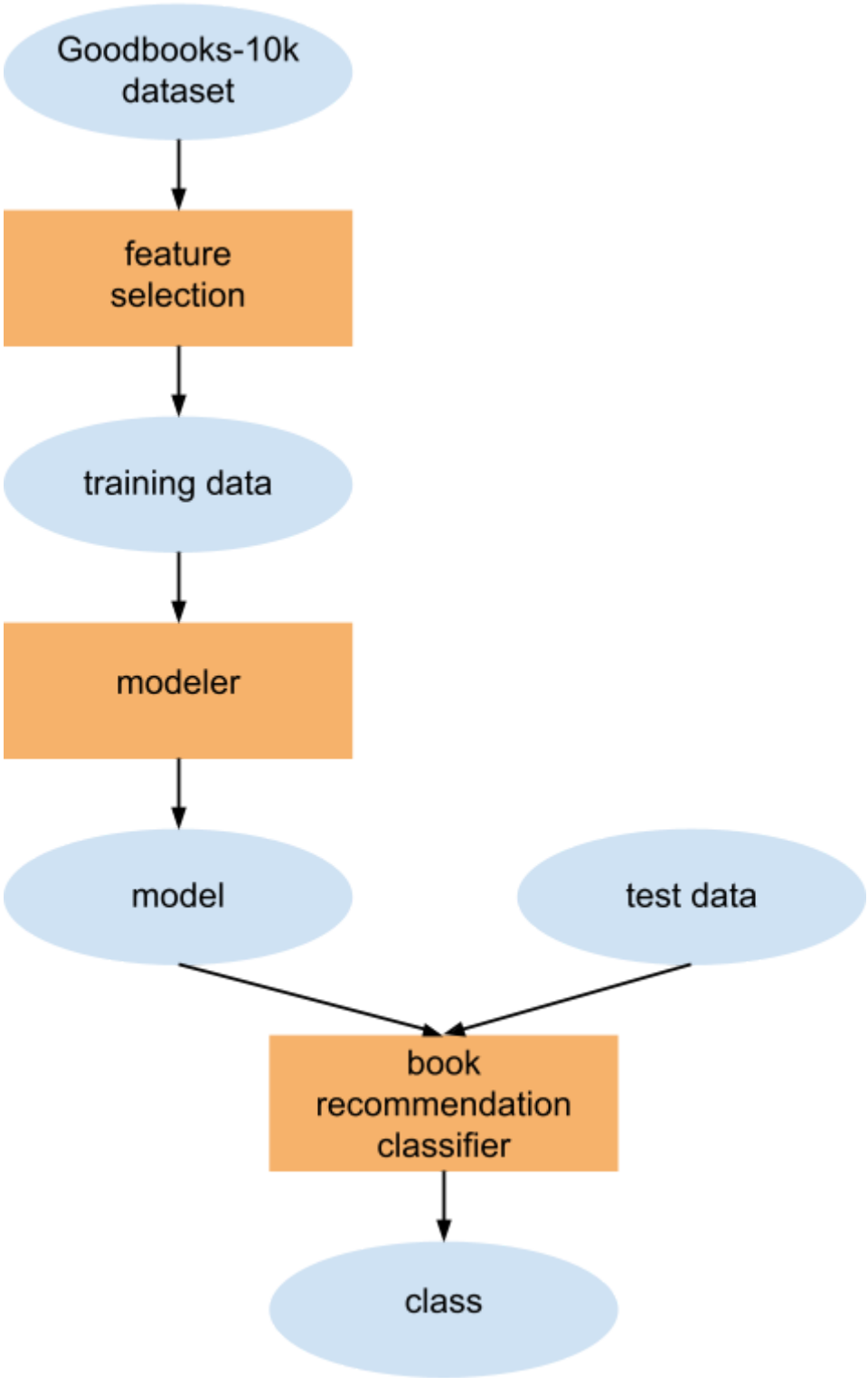

Figure 1: Workflow for book recommendation system.

6 Appendix: HW 1 revised

While there were no errors in my HW 1 to correct and my project proposal between HW 1 and HW 5 are the same, my approach to the project has changed. As such, I have revised section 6.3 Project sketch.

6.1 Introduction

At 10 years old, I moved from LA to Beijing with my family. In the expat community, everyone was always coming and going, so I couldn't escape the feeling that my time in Beijing was temporary. Anticipating my next, inevitable move, I didn't like to hold onto physical things, because I could only think of the nuisance they'd eventually be to pack and ship. Rather than accumulate possessions, I cataloged experiences through blogging, writing and photographing travels, student life, books, and more. After eight years, I left Beijing -- just as I knew I would -- and headed back to my home country for undergrad, but I never stopped blogging.

Given my history with blogging (and by extension, social media), I am fascinated with digital culture topics such as online communities, the decentralization of the internet and the widening divide of common truths, the performance inherent in the construction of self on the internet, the distance and intimacy enabled by technology, the relentless pursuit of self-optimization enabled by technology, the underlying biases encoded in technology, and more. Some of my favorite books that touch on these topics include *Trick Mirror* by Jia Tolentino and *Uncanny Valley* by Anna Wiener.

Though I double majored in computer science and linguistics during my undergraduate studies, my liberal arts education gave me space to learn in other disciplines as well -- such as user experience design, graphic design, anthropology, and more -- through which I could explore the aforementioned topics from a variety of perspectives. When I found out about the MS in Communication Data Science program at USC, I felt that it was the perfect multidisciplinary program for me, tying in my technical academic knowledge and non-technical experience with media.

Despite my experience with programming, I believe there's still a lot for me to learn specific to data science from DSCI549, especially as it is not a class geared toward programming but rather toward high level concepts specific to data science. Programming is just one part of a data science project, but by the end of the semester, I hope to have developed the skills to manage a whole data science project from beginning to end.

Beyond this course, my personal ambition is to understand how we can better use technology to make genuine connections with other people, including fostering one-on-one friendships and building online communities that improve quality of life. Part of this also involves sifting through the deluge of information available online, presenting the right information to the right people in a meaningful way, avoiding echo chambers, combating misinformation, and more.

6.2 Dataset

book_id	goodreads_book_id	best_book_id	work_id	books_count	isbn	isbn13	authors
1	2767052	2767052	2792775	272	439023483	9.78043902348e+12	Suzanne Collins
2	3	3	4640799	491	439554934	9.78043955493e+12	J.K. Rowling, Mary GrandP
3	41865	41865	3212258	226	316015849	9.78031601584e+12	Stephenie Meyer
4	2657	2657	3275794	487	61120081	9.78006112008e+12	Harper Lee
5	4671	4671	245494	1356	743273567	9.78074327356e+12	F. Scott Fitzgerald
6	11870085	11870085	16827462	226	525478817	9.78052547881e+12	John Green
7	5907	5907	1540236	969	618260307	9.7806182603e+12	J.R.R. Tolkien
8	5107	5107	3036731	360	316769177	9.78031676917e+12	J.D. Salinger
9	960	960	3338963	311	1416524797	9.78141652479e+12	Dan Brown
10	1885	1885	3060926	3455	679783261	9.78067978327e+12	Jane Austen

Figure 2: First 10 rows of data from books.csv of Goodbooks-10k dataset; first eight columns of 23 columns.

The dataset I have selected is Goodbooks-10k, representing 6,000,000 ratings from more than 50,000 users for the 10,000 most popular books on GoodReads, a social website for users to rate and recommend books. (Personally, I have used GoodReads since 2014.) Goodbooks-10k can be manually accessed by downloading the ZIP files on the Goodbooks-10k GitHub repository's release page. For convenience, a sample of the data can be previewed directly on GitHub. Otherwise, the data must be downloaded in order to be viewed, as the original files are too large to be viewed directly on GitHub. The dataset can also be accessed through the Spotlight Python library.

The key files of the dataset include five CSV files of tabular data (see Figure 1), as well as an archive of 10,000 XML files of structured data (one XML file for each book). CSV files and XML files are digital and can be read and analyzed by computer programs, so they are machine processable.

Book metadata contained in the CSV files include book titles, authors, publication year, average ratings, number of overall ratings, number of one star ratings through number of five star ratings, number of witten reviews, language of publication, tags, and more.

6.3 Project sketch

There are many possible data science projects that could use this dataset:

~~A book recommendation system could take a book title a reader has enjoyed as input, then recommend a small list of book titles to the reader as an output. This could possibly be done by analyzing user ratings of the input book and linking them to other books that users have also rated highly. It could also be done by analyzing tags of the input book and linking them to other books that have similar tags.~~

A book recommendation system could classify books as recommended or not, taking book instances as input and outputting their class based on features including titles, average ratings, and book tags.

However, it must be noted that GoodReads tags are far from standardized and could benefit from data cleaning, which could involve both standardizing tags as well as sorting tags into further categories, such as genres, target age groups, and author demographics.

Beyond cleaning up the existing data, more data can be collected as well, such as information about user preferences and book sales. Experts such as librarians, book sellers, professional book reviewers, literature professors, and those working in the publishing industry could help identify other important categories, such as writing style.

~~With more specific ways to structure the data, different methodologies for book recommendation systems could be implemented and compared, and more ways to ask classification problems could open up as well.~~ With more specific ways to structure the data, the book recommendation system could account for more features, which could improve accuracy and explainability of the results. This could also open up the possible classification problems that can be asked. Possible classification problems include classifying whether or not a book has been on the New York Times bestseller list, whether a book is classic or contemporary, or classifying a book into a specific genre. For these classification problems, a book title would be the input and the output would be the class.

Statisticians and machine learning experts would be helpful for the technical aspects of both the recommender and classification projects.

6.4 Works cited

Zajac, Zygmunt. *Goodbooks-10k: a new dataset for book recommendations*, 2017. https://github.com/zygmuntz/goodbooks-10k