DATA SCIENCE

CAPSTONE REPORT - FALL 2022

A Multi-stakeholder Book Recommender System for Libraries

Jialin Wang

supervised by
Hongyi Wen

Preface

My name is Jialin Wang and I am a senior at NYU Shanghai. I double major in Mathematics and Data Science, and my research interest lies in Machine Learning, Optimization, and Recommender Systems. The multi-stakeholder recommendation system is a new topic and our university library is also seeking a suitable recommendation system benefiting the students and the library at the same time. As a result, to combine my interest areas with real-life problems, I develop a multi-stakeholder recommendation system for university libraries using machine learning and optimization models, to provide students with desired books and expand the loan volume of under-exposed books in the library simultaneously. I hope my paper will help universities to design and improve their book recommendation schemes, and in a broader sense, help companies to improve their recommendation systems and increase the public exposure of products to consumers.

Acknowledgements

Reading two majors at the same time is hard but also joyful because many people helped me in the process of my Data Science and Mathematics learning. Firstly, I thank my parents and family for always supporting my academic career and interests, for helping me solve and overcome problems, and for providing me with mental and financial aid all the time. Secondly, I thank my Capstone project mentor, Professor Hongyi Wen, at NYU Shanghai, for his careful and meticulous instructions on my project, for answering my questions, and for offering me solutions to various technical problems patiently. Lastly, I thank my friends and senior alumni for their useful academic suggestions, considerate real-life advice, and passionate encouragement in my university life. Wish us all good luck in the future!

Abstract

The emergence of two-sided or even multi-sided marketplaces in recent years made suppliers, platforms, and society crucial stakeholders for recommender system developers to consider. And it is necessary to maximize the satisfaction of the assorted needs of all stakeholders. One such instance is the university library: students need desired books, while the library needs to expand the library's loan volume of various books, achieving a near-even distribution of lending to promote efficient allocation of public resources. Interesting as the problem seems, some issues exist in deriving the solution. We need to determine the filtering structure with better performance in our problem. Secondly, we should choose a word embedding method that functions well in differentiating the items and data sparsity issues. Lastly, the students and the library have different objectives and they may conflict with each other. To recommend books catering to students' demands and promote under-exposed books in the library simultaneously, we form a multi-objective optimization combinatorial problem and try to find the Pareto set using Genetic Algorithm. In other words, we will construct a "knapsack problem" to find a set of books, maximizing the total ratings while constraining the upper limit of the total popularity.

Keywords

Multi-stakeholder Recommender System; Multi-objective Combinatorial Optimization; GloVe; Content-based Recommendation; LASSO; Knapsack Problem; Genetic Algorithm

Contents

1 Introduction

In the past, consumer-centric recommendation systems dominated the business industry, which only recommended desired products for consumers by means of matrix factorization, tensor factorization, neural embeddings, and other traditional methods. However, with the emergence of two-sided or even multi-sided marketplaces (e.g. Amazon, Airbnb) in recent years, suppliers, platforms, and society become crucial stakeholders for system developers to consider as well. Such a multi-sided marketplace includes interactions between multiple-stakeholders, and it is necessary to maximize the satisfaction of the assorted needs of all stakeholders. One such instance is the university library: students need desired books, while the library needs to expand the library's loan volume of various books, achieving a near-even distribution of lending to promote efficient allocation of public resources.

Interesting as the problem seems, some issues exist in deriving the solution. Firstly, content-based recommendation and collaborative filtering are two mainstream methods to model the recommendation and construct the variables, and we need to determine the filtering structure with better performance in our problem. Secondly, turning the features of books and students into numerical vectors is an important process, and the embedding method we choose should function well in differentiating the items and data sparsity issues. Lastly, the students and the library have different objectives and they may conflict with each other. For example, the books students may take interest in are highly popular and students may rush into the same book, thus the popular shelves are empty while unpopular books are left unattended. Consequently, it is essential to figure out an algorithm dealing with the multi-objective optimization problem, balancing the need of students and the library.

To simultaneously recommend books catering to students' demands and promote under-exposed books in the library, we form a multi-objective optimization combinatorial problem and try to find the Pareto set in our problem setting using Genetic Algorithm. Compared with the result of only optimizing students' objective, our optimization design performs well in combining the popularity of books with students' expected ratings of the books and generally reducing the variance of the numbers of lending times of various books. We construct a "knapsack problem" to find a set of books, maximizing the total ratings while constraining the upper limit of the total popularity.

2 Related Work

In general, different stakeholders have different objectives. Some of them are consistent towards a common goal while some of them may be conflicting with each other. Our model has only students and the library as the two stakeholders. In terms of students' preference for books in university libraries, Tian et al [1] quantify students' objectives using three algorithms: collaborative filtering, content-based recommendation, and the hybrid algorithm, to find the books individual students may take interest in. These processes involve reader classification, the establishment of a rating matrix, the construction of vector space, and the calculation of similarity among users and books. However, compared with the content-based recommendation, collaborative filtering-based recommendation is mainly anchored in personal histories and suffers from sparsity problems, as Lee et al [2] describe in the book. Moreover, collaborative filtering is vulnerable to fairness problems like gender bias and racism, which makes the recommender unconvincing.

As a result, content-based recommendation is a good method to seek similarity between users' interests and the book contents because it's only based on the keywords rather than personal history. Meteren and Someren [3] discuss and present how content-based recommendation is functioning in such a structure. First use natural language processing to stem the words from book abstracts and titles to build up profiles (weight vector) for books and students using the TF-IDF method. Then calculate the cosine similarity between students' profiles and book profiles to find the books of most interest to the specific student. Moreover, Meteren and Someren also introduce the advantage of content-based recommendation when the profile vectors need to be updated.

Apart from TF-IDF, GloVe is another way to form the vectors for book contents, which is brought up by Pennington et al [4]. GloVe is an unsupervised learning algorithm for generating the vector representations for words and the training is conducted on aggregated global word-word co-occurrence statistics from a corpus. Different from TF-IDF, the vector for each book is obtained from a global pre-trained set rather than a local temporary set, and thus the acquired vectors are more meaningful. The resulting representations using GloVe showcase interesting linear substructures of the word vector space and are more effective than TF-IDF in some ways.

The problem setting of Tian et al [1] is only for the student side and they do not consider the provider side, which is a single-stakeholder recommendation system. For double-stakeholders, Zheng et al [5] introduce a personalized educational learning recommendation system for two stakeholders: students and the instructor, aiming to recommend project topics for each student in a course. The purposes of students and the instructor are a little bit conflicting because the students would like to choose topics catering to their abilities and interests while the instructor hopes students should try more advanced topics. The problem setting is similar to ours since the objective of students and the library is not collaborative. Zheng et al address the problem using collaborative filtering and a multi-task optimization approach to balance the preferences of students and the instructor, and try to find a Pareto optimal solution. Sürer et al [6] define the objective of providers as the proper distribution of recommendations across retailers.

The Knapsack Problem [7] is an example of combinatorial optimization problem, seeking to maximize the benefit of objects in a knapsack without exceeding its capacity, and our problem could fit into the structure. Limiting the upper bound of popularity, we hope to recommend books to students with higher ratings.

To deal with the multi-objective optimization, Sener and Koltun [8] bring up the multi-task learning as multi-objective optimization by means of Multiple Gradient Descent. MTL can be formulated as multi-objective optimization: optimizing a collection of possibly conflicting objectives. Sener and Koltun introduce the Multiple Gradient Descent Algorithm (MGDA), whose solution is called a Pareto stationary point. The solution to the optimization problem either satisfies the Karush-Kuhn-Tucker (KKT) conditions or gives a decent direction that improves all the tasks. In other words, it is equivalent to finding a minimum-norm point in the convex hull of the set of input points. However, our problem's domain is rather discrete so the general gradient descent is not applicable in our case.

A possible and applicable algorithm is Genetic Algorithm. Genetic Algorithm (GA) is inspired by the process of natural selection, belonging to the larger class of evolutionary algorithms (EA). Genetic algorithms are commonly used to generate optimal solutions from possible solutions to optimization and search problems by relying on biologically inspired operators such as mutation, crossover, and selection [9]. GAs was proposed and developed by John Holland, his students, and his colleagues at the University of Michigan [10]. And Genetic Algorithm has excellent performances in multi-objective optimization problems, as well as automatic programming, immune system, and population genetics.

3 Solution

3.1 Overall Architecture

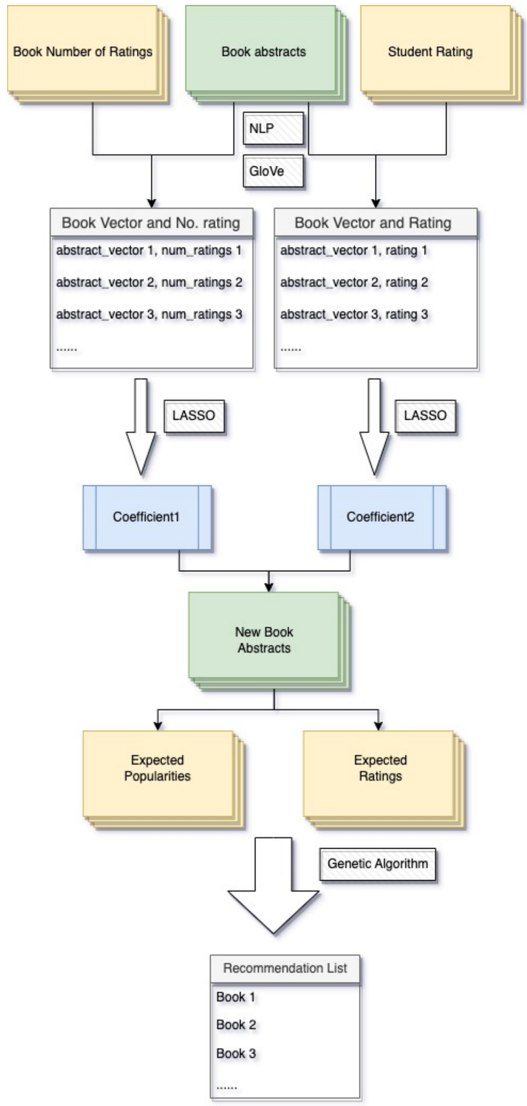

Figure 1: Architecture of our book recommendation system

We perform our training and testing of the multi-objective recommendation system on the Goodreads Datasets [11, 12], especially on the books of poetry shelves. We first collect the abstracts of books and their respective number of ratings (the measurement of their popularity) from the book information dataset. And then we investigate the borrowing information of each customer, including their borrowed books and ratings. By means of natural language processing and word embedding methods such as GloVe and TF-IDF, we turn the book abstracts into numerical vectors through the summation of each word's vector in each abstract. Set A is the abstract vectors with respect to the numbers of ratings, and set B is the abstract vectors with respect to the ratings for each individual. And we apply LASSO to both sets and train the feature coefficients, tuning the hyperparameters to get the optimal coefficient pair of the two sets with the lowest Mean Square Error. To obtain the popularity and each customer's ratings for a set of new books, we implement the trained optimal coefficients pair to the abstract vectors of the new books. As a result, for each book, we have two labels: the expected popularity and the expected rating. Based on the two labels, we then perform the Genetic Algorithm to solve the multi-objective optimization combinatorial problem: maximizing the sum of the ratings of the recommended set of books, while imposing an upper limit to the total popularity summation. Eventually, we acquire the recommended book list for each individual. By tuning the parameters, we can adjust the size of the book list. Moreover, we could change the upper limit of popularity summation to decide whether to recommend popular books or under-exposed books. The architecture of our recommendation system is also illustrated in Figure 1.

3.2 Theory

3.2.1 GloVe and TF-IDF

GloVe [4] and TF-IDF are two well-known word embedding methods and are very useful accompanied by natural language processing. TF-IDF is more of a local word embedding method. According to Meteren and Someren [3], the i-th element w_i of an abstract vector is the weight of the i-th term t_i in the abstract, indicating the importance of the word. In other words, each term is assigned a weight that is based on how often the term appears in a particular document and how frequently it appears in the entire document collection:

$$w_i = tf_i \log\left(\frac{n}{df_i}\right) \tag{1}$$

where tf_i is the number of occurrences of term t_i in document D, n is the total number of documents in the collection and df_i is the number of documents where the term t_i appears at least once. Two features of the text documents determine the rationales behind the TF-IDF method. To begin with, the more frequently a term occurs in a document, the more related it is to the theme of the document. Secondly, the more times a term appears in all of the documents in the collection, the more badly it distinguishes between documents.

For GloVe, it is a global and pre-trained word embedding method. According to Pennington et al [4], GloVe is an unsupervised learning algorithm for generating the vector representations for words and the training procedure is conducted on aggregated global word-word co-occurrence statistics from a corpus. Pre-trained based on large quantities of texts such as Common Crawl, GloVe offers the vector for each word directly. From Figure 2, we can see how Glove projects each word into a point in the space.

The vector representations generated from TF-IDF using our Goodreads dataset are very sparse, with less than one percent non-zero entries. Compared to TF-IDF, the vectors generated from GloVe, are not that sparse and can effectively distinguish between different words (since the vector distance between words is large). For GloVe, there is no relationship between the documents when performing word embedding, so we can dynamically add and delete vectors from the set. As a result, we will implement GloVe to our problem. After getting rid of all the stop words, we sum up all the vectors of words in the book abstracts and divide the summation by the total number of words, and then we obtain the abstract vector for the book.

3.2.2 Collaborative Filtering and Contented-based Recommendation

Collaborative Filtering (CF) is a method of making predictions about the interests of a user by collecting taste information from many other users. The underlying assumption of the collaborative filtering approach is that if a person A has the same preference as person B on an issue, A is more likely to have B's preference on a different issue than that of a randomly chosen person [13]. The method usually engages with Matrix Completion where we need to predict the ratings of users through dimension reduction on the user-item matrices. However, CF suffers from data sparsity problems since many recommendation systems are based on large datasets. For instance, the user-item matrix of Goodreads dataset used for collaborative filtering is extremely large and sparse, which brings about challenges in the performance of the book recommendation. Additionally, when we need to add new items into the system, items need to be rated by a substantial number of users before they could be recommended to users who have similar opinions to the ones who rated them. Similarly, when a new user comes into the system, they will need to rate a sufficient number of items to enable the system to capture their preferences accurately [13].

While for Content-based Recommendation, the new item problem does not affect the performance of the model because the recommendation of an item is based on its discrete set of descriptive properties rather than its ratings [13]. Content-based recommendation methods are based on a description of the item and a profile of the user's preferences. In our case, keywords in abstracts are used to describe the books, and a student profile is built to indicate the characteristics of books this student likes. In other words, these algorithms try to recommend items similar to those that a user liked in the past or is examining in the present. To abstract the features of items, item presentation algorithms will be applied [14]. A widely used algorithm is the GloVe Vector, so we use GloVe embedding to turn each book's abstract into a vector. Together with the two labels: the number of ratings and an individual's rating on the book, we can apply LASSO to find the relationship between the content of the book and the popularity and the student's likeness of the book. Thus for a new book, we could deduce its popularity and one's personal rating from the abstract, rather than from the blank lending information of the new book.

3.2.3 LASSO

LASSO is short for "least absolute shrinkage and selection operator", which is a regression method that performs feature selection and regularization. It can enhance the prediction accuracy and interpretability of the statistical model. LASSO was first proposed by Robert Tibshirani, a professor of Statistics at Stanford University, in 1996 based on non-negative parameter inference by Leo Breiman [15]. It is a ℓ_1 penalized model by adding the ℓ_1 norm of weights to the least squares cost function. Therefore, LASSO Regression minimizes:

$$R(\beta) = \|\mathbf{y} - X\beta\|_2^2 + \lambda\|\beta\|_1 \tag{2}$$

where $\mathbf{y}$ is the response (ground truth), $\mathbf{X}$ (feature) is a data matrix making $\mathbf{X}\beta$ be the predict value and λ is a hyperparameter for penalty. Compared with ordinary least squares, LASSO outperforms OLS in various ways. By penalizing on the ℓ_1 norm of the coefficients, LASSO can select important features out of the various variables, thus increasing the interpretability of the regression model. In our experiment, we apply LASSO to predict the expected ratings and popularity of books, to form the two labels of books to perform combinatorial recommendation. The LASSO Regression minimizes:

$$R(\beta_{\mathbf{1}}) = \|\mathbf{y_1} - X\beta_{\mathbf{1}}\|_2^2 + \lambda_1\|\beta_{\mathbf{1}}\|_1 \tag{3}$$

$$R(\beta_{\mathbf{2}}) = \|\mathbf{y_2} - X\beta_{\mathbf{2}}\|_2^2 + \lambda_2\|\beta_{\mathbf{2}}\|_1 \tag{4}$$

where $\mathbf{y_1}$ and $\mathbf{y_2}$ are the ground truth for ratings and numbers of ratings, $\mathbf{X}$ is the abstract vector making $\mathbf{X}\beta_{\mathbf{1}}$ and $\mathbf{X}\beta_{\mathbf{2}}$ be the predicted values and λ_1 and λ_2 are hyperparameters for the penalty. Since the dimension of the abstract is 50, LASSO will effectively select the important variables and boost the performance of prediction. Training on vectors with ratings through a hyperparameter grid, we find the optimal solution with the lowest MSE 0.59125 when λ_1 equals 0.000001, while for the training on vectors with the number of ratings, the optimal solution is found when λ_2 equals 0.0005. Finally, we apply the trained coefficients to new books and obtain the labels for the books for recommendation purposes.

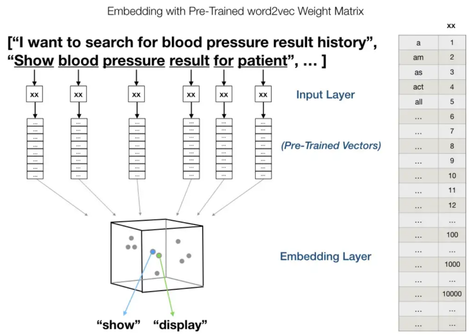

Figure 2: Graph Illustration of Glove Embedding[16]

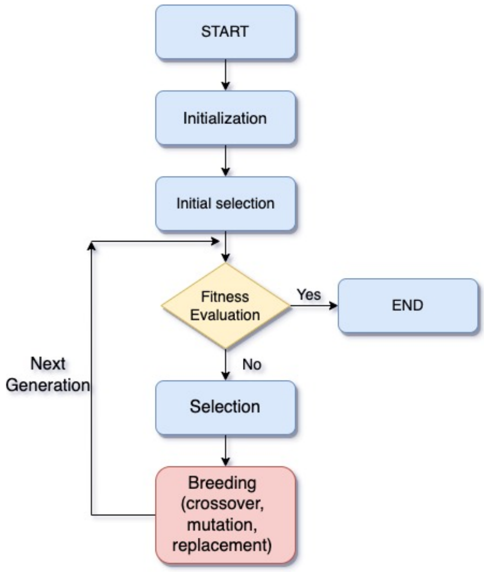

Figure 3: Flow chart of Genetic Algorithm

3.2.4 Knapsack Problem and Genetic Algorithm

The most common Knapsack problem being solved is the 0-1 Knapsack problem, which is also applicable in our scenario. The number x_i of copies of each kind of item is zero or one. Given a set of n items numbered from 1 up to n, each with a weight w_i and a value v_i, along with a maximum weight capacity W,

$$\text{maximize} \sum_{i=1}^{n} v_i x_i \quad \text{subject to} \sum_{i=1}^{n} w_i x_i \leq W \text{ and } x_i \in \{0, 1\} \tag{5}$$

where x_i is the number of instances of item i to include in the knapsack. In other words, the problem is to maximize the sum of values in the knapsack under the constraint that the total weight is less than or equal to the knapsack's capacity. In our library book recommendation case, the optimization could be formed as:

$$\text{maximize} \sum_{i=1}^{n} r_i x_i \quad \text{subject to} \sum_{i=1}^{n} n_i x_i \leq P \text{ and } x_i \in \{0,1\} \tag{6}$$

where x_i is the number of book i to include in the recommended combination (in our setting, 0 or 1), n is the number of books in the set of new books with two expected labels, r_i is the expected rating of book i for a particular student and n_i is the expected number of rating of book i, along with P as the maximum popularity sum. To deal with it, we implement the Genetic Algorithm to solve the multi-objective combinatorial optimization problem.

According to Hristakeva and Shrestha [17], the Genetic Algorithm (GA) could be implemented to solve the 0-1 Knapsack Problem (KP). GA starts with a set of possible solutions (chromosomes) called population. A new population is created from solutions of an older population, hoping to obtain a better population. Solutions that are chosen to form new solutions (offspring) are selected according to their fitness function. The more suitable the solutions are the bigger chance they have to reproduce. This process is iterated until the condition is satisfied [18]. The basic outline of GA we will implement is, where each iteration is called a generation:

1. **Start**: Randomly initiate a population of N chromosomes.
2. **Fitness**: Calculate the fitness of all the chromosomes.
3. Create a new population:
   a) **Selection**: Select chromosomes following the selection method from the population.
   b) **Crossover**: Perform crossover on the 2 chromosomes selected.
   c) **Mutation**: Perform mutation on the chromosomes acquired.
4. **Replace**: Replace the current population with the new population.
5. **Evaluation**: Evaluate whether the end condition is attained. If so, stop. If not, return the best solution in the current population and go to Step **Fitness** [18].

The above process is also shown in Figure 3.

3.3 Findings

From our experiment and trials, we find that GloVe has better word embedding performance than TF-IDF because the abstract matrix generated from GloVe is denser, and we can dynamically add and delete vectors from it. Additionally, Content-based Recommendation is more suitable than Collaborative Filtering in the Goodreads dataset because the interaction data sparsity (only 0.3 percent non-zero entries on average) poses a challenge to the accuracy of prediction performance, while Content-based Recommendation only focuses on the content of abstracts themselves. Moreover, LASSO has the lowest MSE when the penalization term is 0.000005 for training the number of ratings, while 0.0001 for the training on vectors with ratings. Lastly, the training results of Genetic Algorithm and performance comparisons between single-objective and multi-objective optimization will be presented in the following sections.

4 Results and Discussion

4.1 Experimentation Protocol

For a new set of books belonging to the poetry category (50 units), we would like to investigate which of them one particular student may take interest in, while expanding the popularity of under-exposed books, to form such a recommendation combination. To evaluate the performance of such a multi-objective combinatorial optimization, we will first select an active library user who has over 200 borrowing histories and use his or her rating data (around 250 units) to predict the expected ratings for the new set of books by means of LASSO through a grid search. To check the performance of LASSO, we will calculate the Mean Square Error for each hyperparameter, and then compare the ground truth ratings and predicted ratings with the lowest MSE in the testing set (10 units). And we perform the same procedures to evaluate predicting performance of numbers of lending for the new set of books (2000 units).

For performance comparisons between single-objective recommendation and multi-objective recommendation, we compare the books selected by the two schemes, including the distribution of recommended books, the lending variance, and the ratio of ratings and popularity. Numerical analysis and graphs are also provided.

4.2 Results

Table 1 shows the values of the hyperparameter λ with the corresponding LASSO training Mean Square Error, for the linear fittings of one user's rating data and the popularity data (numbers of ratings) with respect to the abstract content. Through the grid search, we will implement LASSO with the lowest Mean Square Error to our new book ratings and popularity prediction process. With the best penalization terms $\lambda_1 = 0.000001$ and $\lambda_2 = 0.0005$, we present the comparison in testing sets between the ground truth and predicted values for both expected ratings and expected popularity cases, via graphs, in Figure 4 and Figure 5. Their respective Mean Square Error and R-squared are 0.64753, 0.24134, and 8.2584, 0.19881. These data show that the LASSO fitting for the prediction of user ratings is reasonable since the MSE is relatively small (around 0.5),

Table 1: LASSO Penalization Term and MSE (Max Iteration: 5000)

Experiment	λ of rating training	MSE	λ of popularity training	MSE
1	0.0000001	0.59125	0.00005	85.4542
2	**0.000001**	**0.59125**	**0.0005**	**85.4542**
3	0.00001	0.59131	0.005	85.4549
4	0.0001	0.59654	0.05	85.4852
5	0.001	0.63875	0.5	85.6294
6	0.01	0.73508	1	85.7922

Table 1: LASSO penalization term and corresponding MSE

while the LASSO fitting for the prediction of book popularity is not that perfect since the MSE is relatively large, indicating that the linear regression might not be useful.

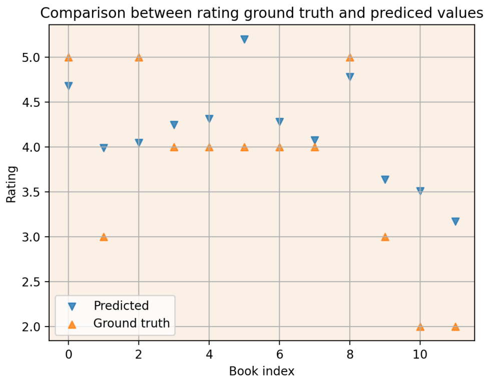

Figure 4: Comparison of rating ground truth and predicted values

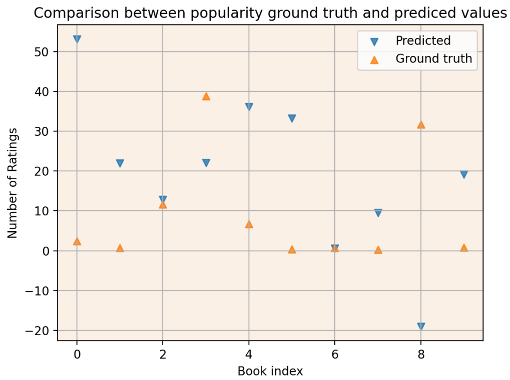

Figure 5: Comparison of popularity ground truth and predicted values

When it comes to the results of the multi-objective combinatorial optimization problem, Table 2 presents the book recommendation combinations indexes (15 books out of 40 books) generated by single-objective and multi-objective methods, for one active Goodreads user. All the numbers represent the indexes of the recommended books. The single-objective method (S) refers to just recommending books with high expected ratings for a certain amount, so the book indexes in column 'S' are based on their expected ratings in descending order. The single-objective goal only takes students' interest into account while does not consider the popularity of each book, thus the recommended books are ones that only have high expected ratings, based on the taste of the specific user. In this way, some less popular books may be left under-exposed for a long time. However, the multi-objective combinatorial method (M) achieves a balance between good ratings and reasonable popularity. For example, in column 'M_1', book 21 has a relatively high expected rating value of 4.26 which is not in the top 15, but its expected number of ratings is only approximately 8 times. Assuming the prediction is accurate if the user reads this book, he or she may surprisingly find the book interesting and then recommend the book to peers, thus increasing the popularity of the under-exposed books. Finally, the resources in the library are fully and equally exploited and used. In contrast, if students all rush to borrow popular books, the demand will exceed the real volume of accessible books, which leads to unbalanced use of books in the library.

S	M_1	M_2	M_3
37	2	2	6
35	4	6	8
29	6	8	10
17	8	10	12
33	10	12	14
3	14	17	17
11	16	18	20
...	...	...	...
27	37	37	37

Table 2: Recommended book combinations indexes for one user using S and M two methods

Remark. *The reason why M has three very similar columns is that the Genetic Algorithms use random sampling methods to create generations of random candidate solutions. It can get "stuck" on local optima, and if other local optima (or the global optimum) is too far away, operations such as crossing and mutation might not provide sufficient variation to get "unstuck" from the original place [19]. As a result, the above M_1, M_2, and M_3 are three similar possible results of optimal solutions. Improvements need to be done in tuning the hyperparameters (increasing crossing rate, mutation rate), to obtain more variation in training, and get a convergent and unique optimal solution in further experiments.*

The recommended books using single-objective and multi-objective combinatorial methods have different distributions in terms of their expected ratings and expected popularity. According to Figure 6, we could see that books recommended by single-objective method tend to have high expected ratings but expected popularity varies. While for the books recommended by multi-objective method, not only do they tend to have relatively high ratings, but some of them also tend to have relatively low popularity. They cluster between Popularity 2.5 - 3.5, where blue points seldom occur, achieving a balance between good ratings and reasonable popularity.

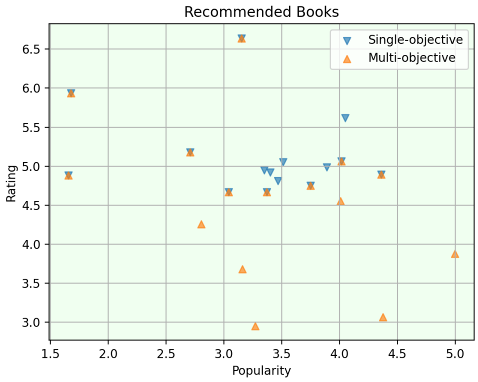

Figure 6: The Label Distribution of Books Using Two Methods

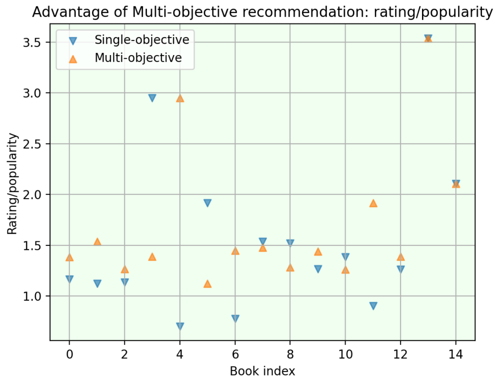

Figure 7: Ratio Comparison between Two Methods

The ratio of expected ratings and expected popularity is another performance measurement of how multi-objective method outperforms single-objective method. According to Figure 7, the multi-objective method usually results in a higher ratio (on average 1.7) of expected ratings and expected popularity, which means that the recommended books generated by multi-objective method generally have high ratings and low popularity, which effectively contributes to the balance between users' interest and volumes of books in the library. While the ratio of expected ratings and expected popularity for single-objective method is 1.5.

Performing this multi-objective recommendation procedure for several Goodreads users and counting the total lending times for each book, we could also find that this method reduces the long-term variance of book lending times among the candidate books. Figure 8 illustrates the long-term lending frequency of the candidate books using the two methods, where the single-objective method has a variance of 227.1 and the multi-objective method has a variance of 192.5. This shows that multi-objective combinatorial optimization reduces the variance of book lending times by 16 percent (we assume the users borrow all the recommended books generated by the two methods). In Figure 9, we can see that the distribution curve for the multi-objective method is narrower. In the longer term, the loan volumes may achieve a near-even distribution among books at a faster speed, using the multi-objective method.

Figure 8: Variance of Lending Times Using Two Methods

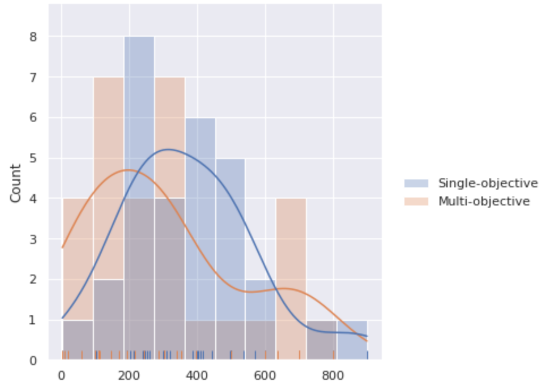

Figure 9: Distribution of Lending Times Using Two Methods

To conclude, the multi-objective combinatorial optimization method outperforms the single-objective method concerning the distribution of recommended books, the long-term lending variance, and the ratio of ratings and popularity. Some issues of the multi-objective method still exist and we will discuss them in the following sections.

5 Discussion

Our library book recommender system is an innovative, multi-stakeholder system that can satisfy both students' and libraries' needs simultaneously. It can offer students desired books while promoting the under-exposed books in the library, achieving a near-even distribution of book lending times. Compared with previous research which only considers students' benefits like [1], our model outperforms them in considering both sides. Moreover, our method for forming book abstract vectors using GloVe is more effective than TF-IDF mentioned in [3] because GloVe fixes the problem of vector sparsity. What is more, our use of content-based recommendation also successfully deals with the data sparsity problem, compared with [5] which uses collaborative filtering.

Several issues with the multi-objective method still exist. To begin with, except for getting rid of the stop words and punctuation marks, more natural language processing should be done on the books' abstracts. Stemming the word and matching the stemmed word vector will make the final abstract vector more accurate and effective in expressing the content of each book. For example, the word "duplicate" should be semantically equal to the word "duplication", and natural language processing could match them together rather than separate them. Additionally, using LASSO for the prediction of numbers of ratings for each book does not fit well because the Mean Square Error is a little bit large. We should try some nonlinear methods such as neural networks to better fit the regression problem. Regarding the multi-objective combinatorial optimization using Genetic Algorithm, the hyperparameters (crossing rate and mutation rate) should be tuned more intensively to give out a more convergent recommendation combination. In terms of the optimization algorithm, we could also try to implement the multiple gradient descent, turning our problem into a problem with "continuous" step functions.

6 Conclusion

In conclusion, our multi-stakeholder book recommender system for libraries can effectively provide students with desired books while promoting the under-exposed books in the university library, thus achieving a near-even distribution of book lending times. Though there are some similar attempts in different fields of study, the multi-stakeholder book recommender system for libraries is the first in this category. And it outperforms other similar multi-stakeholder recommendation models in terms of:

1. dealing with the book abstract vector sparsity problem using Glove vector embedding
2. fixing the user-book interaction data sparsity problem by content-based recommendation
3. transforming the unusual optimization problem with a discrete domain into a well-known "Knapsack Problem", thus we can implement Genetic Algorithm to give optimal solutions

In the future, we will put our focus on improving the semantic quality of the abstract vectors using higher-level natural language processing; and tuning our Genetic Algorithm model to obtain a more convergent solution, or searching for a new algorithm (deep learning) to better deal with the multi-objective combinatorial optimization problem.

References

[1] Y. Tian, B. Zheng, Y. Wang, Y. Zhang, and Q. Wu, "College library personalized recommendation system based on hybrid recommendation algorithm," *Procedia CIRP*, vol. 83, pp. 490–494, 2019, 11th CIRP Conference on Industrial Product-Service Systems. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2212827119307401

[2] S. Lee, J. Yang, and S.-Y. Park, "Discovery of hidden similarity on collaborative filtering to overcome sparsity problem," in *Discovery Science*, E. Suzuki and S. Arikawa, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 396–402.

[3] R. Meteren, "Using content-based filtering for recommendation," 06 2000.

[4] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543. [Online]. Available: http://www.aclweb.org/anthology/D14-1162

[5] Y. Zheng, N. Ghane, and M. Sabouri, "Personalized educational learning with multi-stakeholder optimizations," in *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*, ser. UMAP'19 Adjunct. New York, NY, USA: Association for Computing Machinery, 2019, p. 283–289. [Online]. Available: https://doi.org/10.1145/3314183.3323843

[6] O. Sürer, R. Burke, and E. C. Malthouse, "Multistakeholder recommendation with provider constraints," in *Proceedings of the 12th ACM Conference on Recommender Systems*, ser. RecSys '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 54–62. [Online]. Available: https://doi.org/10.1145/3240323.3240350

[7] G. B. Mathews, "On the partition of numbers," *Proceedings of the London Mathematical Society*, vol. s1-28, no. 1, pp. 486–490, 1896. [Online]. Available: https://londmathsoc.onlinelibrary.wiley.com/doi/abs/10.1112/plms/s1-28.1.486

[8] O. Sener and V. Koltun, "Multi-task learning as multi-objective optimization," 2018. [Online]. Available: https://arxiv.org/abs/1810.04650

[9] M. Mitchell, *An introduction to genetic algorithms.* MIT press, 1998.

[10] J. H. Holland, "Genetic algorithms," *Scientific american*, vol. 267, no. 1, pp. 66–73, 1992.

[11] M. Wan and J. J. McAuley, "Item recommendation on monotonic behavior chains," in *Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018, Vancouver, BC, Canada, October 2-7, 2018*, S. Pera, M. D. Ekstrand, X. Amatriain, and J. O'Donovan, Eds. ACM, 2018, pp. 86–94. [Online]. Available: https://doi.org/10.1145/3240323.3240369

[12] M. Wan, R. Misra, N. Nakashole, and J. J. McAuley, "Fine-grained spoiler detection from large-scale review corpora," in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, A. Korhonen, D. R. Traum, and L. Màrquez, Eds. Association for Computational Linguistics, 2019, pp. 2605–2610. [Online]. Available: https://doi.org/10.18653/v1/p19-1248

[13] Wikipedia contributors, "Collaborative filtering — Wikipedia, the free encyclopedia," https://en.wikipedia.org/w/index.php?title=Collaborative_filtering&oldid=1092795851, 2022, [Online; accessed 3-December-2022].

[14] ——, “Recommender system — Wikipedia, the free encyclopedia,” https://en.wikipedia.org/w/index.php?title=Recommender_system&oldid=1121815715, 2022, [Online; accessed 3-December-2022].

[15] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996. [Online]. Available: https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.2517-6161.1996.tb02080.x

[16] J. Mangiavacchi. (2018) Coreml with glove word embedding and recursive neural network — part 2. [Online]. Available: https://medium.com/@JMangia/coreml-with-glove-word-embedding-and-recursive-neural-network-part-2-ab238ca90970

[17] M. Hristakeva and D. Shrestha, “Solving the 0-1 knapsack problem with genetic algorithms,” in *Midwest instruction and computing symposium*, 2004, pp. 16–17.

[18] M. Obitko, “Genetic algorithms,” *Internet publication*, 1998.

[19] A. Cheong. (2012) Why genetic algorithm gives different results for optimization of one objective function with same parameters in matlab optimization toolbox? [Online]. Available: https://stackoverflow.com/a/14060943