Goodreads Bestseller Detector System

Lokman Mheni
EPFL
Electrical Engineering
Sciper: 326680

Fabio Palmisano
EPFL
Electrical Engineering
Sciper: 296708

***Abstract*—This project investigates predicting book bestseller status using various machine learning models and graph-based features, highlighting the potential of linear models, the challenges faced by neural networks, and the benefits of incorporating statistical book features.**

***Keywords*—** Network Machine Learning, Logistic Regression, Book recommender, GNN, GCN

I. Introduction

This paper aims to describe the work done in the context of the Course Project for the EE-452 "Network Machine Learning" class. The project guidelines are to understand and use the tools learned during the lectures and the various assignments completed during the semester.

This project investigates the application of graph network analysis to determine if a book could become a bestseller.

After giving a bit of theoretical context in Section II, Section III presents some data analysis needed to compute results with a reduced dataset due to the relatively heavy computational cost for local processing. The study primarily focuses on analyzing the properties of these different graphs and some statistics about the ratings. The goal is to use these properties to identify which features have the most impact on a book's success.

Using the goodbooks-10k dataset, multiple graphs will be constructed, including a user-to-user graph based on the books rated by each user, a to-read book-to-book graph where each pair of nodes is linked by a user who made a "to read" suggestion to both books, and the most utilized graph, the book-to-book graph based on user ratings. More details on these graphs will be provided in the specific section IV.

Subsequently, as described in Section V, machine learning techniques, including linear models and Graph Neural Networks, will be used to refine the determination of blockbusters. As is common in this type of study, defining what constitutes a bestseller is not straightforward [2] [3]. Subjective considerations had to be made to determine what qualifies as a blockbuster and what does not.

All data analysis and model computations can be found in the following GitHub Repository.

This paper will present the issues encountered during the process and propose solutions to address those problems.

II. Theoretical Elements

Some theoretical aspects need to be explained to understand the main interest of this project. The first concepts that are really interesting to recall when analyzing graphs are the different properties of graphs. As seen in the lectures [1], the goal is to obtain some properties for each graph, considering that each kind of graph can be better represented by one or more features. This allows us to extract structural information at different levels for each graph.

- **Node Degree**: Node degree represents the level of connectivity for one node (degree = sum of the edge weights).
- **Betweenness Centrality**: Betweenness Centrality measures how often a node appears on the shortest paths between other nodes.
- **Closeness Centrality**: Closeness Centrality shows how close a node is to the other nodes in the network.
- **Eigenvector Centrality**: Eigenvector Centrality illustrates the influence of a node in the network, i.e., how much it is surrounded by important neighbors.
- **Clustering Coefficient**: Clustering Coefficient measures the tendency of nodes to cluster together, or, in other words, the fraction of a node's neighbors that are connected.
- **Page Rank** : The PageRank algorithm measures the "importance" of nodes in a graph. Each node receives a rank that corresponds to the probability that a "random walker" will visit the node.

These are some of the main features in analyzing graph properties, and they are used in this project.

III. Data Analysis

Before delving into building graphs and models, it is essential to examine the data to understand the problem. The data consist of five datasets. Prior to any model training, insightful analysis were done on each of these different datasets.

One of the datasets contains all the ratings given by users, amounting to nearly six million ratings. The number of users who have made ratings is $53,424$. This large volume of data needs to be analyzed, and some reductions should be made to handle it more easily.

The second dataset, "to read book", contains less data and includes all the books that users consider interesting and recommend to read. The books dataset consists of all the information about the books, such as authors, ratings, etc. There are 10,000 books in this dataset.

Additionally, there are two more datasets that are less used in this project: "tags" and "book tag".

The initial exploration of the datasets involved basic analyses, such as computing the mean ratings or examining the distribution of ratings for each user. After that, two distinct models are built and specific data analyses tailored to each approach are conducted.

A. First Model

Initially, a straightforward model creation process was chosen. In this model, there will be the construction of a book-to-book graph where edges represent recommendations made by users. If User X recommends both Book A and Book B a "To Read" tag is assigned to both books by User X. Consequently, an edge connects the nodes representing Book A and Book B, with User X's recommendation serving as the edge's attribute.

During the construction of this graph, significant variations were observed in the number of "To Read" suggestions for different books. Some books received numerous recommendations, while others received relatively few. For instance, investigating the correlation between the frequency of "To Read" suggestions and the number of ratings received by each book revealed a strong positive linear relationship (correlation coefficient of 0.72). This indicates that as the number of ratings a book receives increases, so does the number of

"To Read" suggestions for that book, and vice versa. Conversely, no significant correlation was found between the frequency of "To Read" suggestions and the average rating received by a book. Furthermore, an increase in the number of ratings for a book does not lead to a proportional increase in the percentage of "To Read" suggestions. On average, 20% of the raters give a "To Read" tag to books.

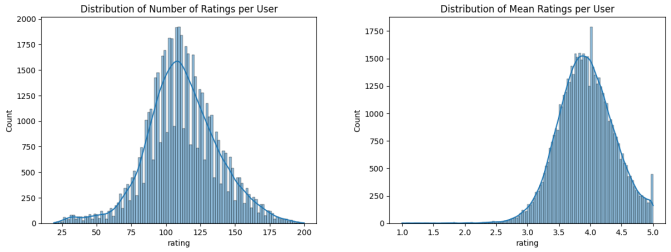
Following this analysis, the first book-to-book graph was constructed, wherein nodes were labeled based on the frequency of "To Read" suggestions by users. An arbitrary threshold was applied to create a balanced dataset, with approximately 50% of the books labeled as bestsellers (label 1) and the remaining labeled as non-bestsellers (label 0). This model was then used to train logistic regression, random forest tree regressor, and SVM models, utilizing various graph features such as degree centrality and betweenness centrality. Notably, the random Forest achieved an F1 score of 80%, demonstrating promising results.

Despite achieving satisfactory performance with the initial model, inherent limitations were recognized, due to the construction of the graph, where "to read" tages were chosen to be the edges connecting the books. Predicting bestseller status appeared trivial, as it could be directly inferred from the dataset without the need for machine learning models: the information about the number of "to read" tags received by a book is intrinsectly contained in the graph itself.

Consequently, it was necessary to refine the approach and develop a more robust and worth model.

B. Second Model

For the second model, statistical analyses were revisited to identify meaningful nodes for training the model. Recognizing the subjective nature of defining a bestseller, the aim was to identify books that best represented the population of readers. To achieve this, statistics such as mean ratings per book, mean ratings per user, and the number of ratings per user were analyzed. Notably, two distinct Gaussian distributions were observed for mean ratings per user and the number of ratings.



(a) Distribution of the number of ratings per user

(b) Distribution of the mean of ratings per user

Fig. 1: Different statistical distributions

With a quick calculation, it became evident that handling a network with one billion possible edges was impossible. To make it more computable, reducing the data was necessary. However, randomly reducing the data would be pointless.

Initially, an attempt was made to reduce the dataset by selecting values between the 25th and 75th percentiles. However, due to computational constraints and prolonged processing times, the approach was revised. Instead, a representative subset comprising 20% of users was selected, specifically those falling between the 40th and 60th percentiles for mean ratings and then the 20% of them was filtered again by selecting those falling between 40th and 60th percentiles of the number of ratings as well. This subset effectively captured the mean behavior of the population of readers. The threshold can be adjusted, and increasing the amount of data can improve the model's performance, as well known in data science in the majority of the cases.

With these refined datasets, the construction of the book-to-book graph proceeded, incorporating only the selected subset of users.

This analysis was crucial to capture more representative data in the constructed graphs. Now, the construction of the different graphs needs to be considered. By reducing the dataset in this fashion instead of using random samples, the analysis becomes more robust.

IV. Graphs

A. User-to-User

Before delving into the details of the two main graphs used in this project, it's important to mention that additional graphs were created to explore specific features or characteristics that could help determine whether a book is a bestseller or not. For instance, a user-to-user graph was generated based on ratings and another one based on "to-read" tags. However, the insights obtained from these graphs were not particularly relevant for the project. Additionally, due to computational constraints, it was not feasible to construct entire graphs. For example, the initial attempt to build the complete user-to-user graph for users who rated the same book took 50 hours. Making those exhaustive and cumbersome graphs wasn't practical nor useful. The only discernible findings from these graphs were certain interesting clustering patterns observed in the user-to-user graphs based on "to-read" tags, which are presented in Fig. 2.

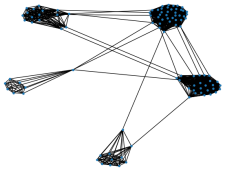


Fig. 2: Clustering in User-to-User Graph

B. Book-to-Book

For both the first and second models, the structure of the graphs remains largely the same. The key differences lie in how the edges are constructed and the features they possess. The nodes and their labels that correspond still to the discussed zeros and ones, are representing books. The construction process for these graphs is depicted in Figures 3 and 4.

Both the "to read" and "ratings" datasets contain information about users and books. To construct the graphs, each user is considered individually, and for each user, all the books they have rated or respectively marked as "to read" are taken into account.

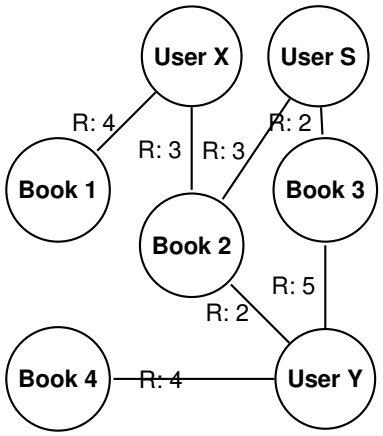


Fig. 3: Undirected Graph to link Books rated by same user.

Then, edges are created between all possible pairs of these nodes. As previously outlined, a connection between two books exists if a user has rated both books. In the case of Book-to-Book connections with ratings, the edges are weighted based on the number of users who have rated both books. For example, in this case, both Book 2 and Book 3 are rated by User S and User Y, so the weight of the edges between these two nodes will be 2.

Please note that here just the fact that a two books have been rated by the same user is included as information in the edge connection betweek the corresponding two nodes. The rate itself is not taken into account because it would not add any value, considering users were sampled by taking only the 20% of them most close to the mean rate (approximately 3.8%).

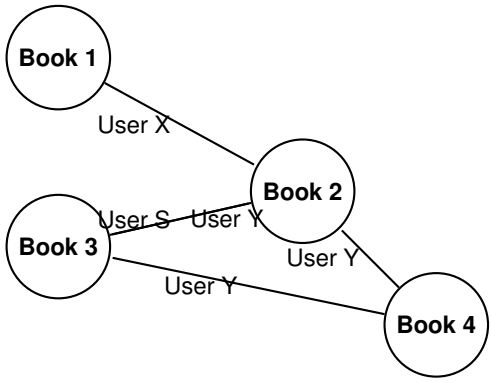


Fig. 4: Book To Book Graph Based on the ratings.

As discussed earlier, graphs aren't fully constructed due to computational constraints. The "to read" data, being smaller, accommodates more nodes and edges in the model. However, ultimately, the model was trained with fewer nodes compared to the "ratings" type, and the results are satisfactory. With more nodes, it could likely achieve even greater accuracy. Nonetheless, optimizing it doesn't make much sense given its inherently trivial task. Nonetheless, for the book-to-book connections based on ratings, a data preprocessing step was undertaken to select the most representative nodes, as detailed in section III-B.

After this step, a graph of 9,823 nodes and more than 6 million edges was constructed. Considering that the dataset contains 10,000 books, we obtained a graph with over 98% of the books and, finally, only 2,578 users, which confirms the density of the data present in the dataset. For the node labels, as specified earlier with the arbitrary threshold value, we ended up with a balanced dataset.

V. Models

The first model, which has been discussed extensively in previous sections, is the book-to-book model using the to-read dataset. In this model, edges connect nodes corresponding to books that have been labeled by a user as "to read." It has been noted that the limitations of this model are clearly identified, and all computations for this model are trivial tasks that can be directly deduced from the dataset without any machine learning model. Therefore, it makes more sense to merge the datasets to train more robust models, which is done in the second model. All derivations for the first model can be found in the file https://github.com/fpalmisa/NML_2024_Project/blob/main/Simple_Model.ipynb. The structure of the first and second graphs is slightly similar.

After processing the data referenced in section III and creating the graphs in section IV, models were created using basic machine learning techniques such as Logistic Regression, Random Forests, and SVM. Neural Networks and Graph Neural Networks are trained only for the second graph, which is more significant in what concerns the data used and more representative of the definition of a bestseller. Therefore, in this section, only what was done with the second model is detailed, considering that the first model is slightly similar. The results of the first model are however presented in the next section VI.

A. Considering the features

As recalled in section II, considering the properties of the graph in analysis networks is crucial for understanding the structure of graphs. In this case, the first model trained was a logistic regression, using graph properties as features. The properties considered in the first model included degree, degree centrality, closeness centrality, betweenness centrality, eigenvector centrality, clustering coefficient, PageRank, and average neighbor degree. We aimed to incorporate as many features as possible to capture the structure of the graphs and achieve good predictions.

Initially, a smaller model was created due to the computational cost of calculating these features, particularly for properties like betweenness centrality, which are time-consuming. Working with a smaller model was faster, and then, in a six-hours run, the other features were added. All these features were saved in a CSV file to avoid recomputation and save time.

The most representative features, yielding the best results, were degree, degree centrality, closeness centrality, betweenness centrality, and clustering coefficient. Using these features, we trained a logistic regression model and a simple neural network model with two layers initially, and then with three layers, achieving slight improvements but not significant differences.

B. Using the info of Books

As discussed in section VI, the results from the logistic regression are satisfactory but can be improved. Some features are more representative than others. For example, having a high degree means a book is one of the most rated, but does that imply that the most rated books are really blockbusters? Not really as seen in the data processing in section III.

After considering all the datasets, we decided to use various characteristics of the books from the books dataset as features. Features like average rating, ratings count, text review count, ratings of 1, ratings of 5, and so on, were now considered. The results clearly improved, which makes sense because we added more statistical information about how the books are rated. Meaningful results were obtained using 14 of the 16 features, confirming that each feature contributes to determining if a book will be a bestseller or not.

Following the logistic regression, SVM and Random Forest models were trained as well. A grid search was conducted for the Random Forest model to find the best parameters that fit the model. Bootstraping was utilized, and various depths and leaf configurations were trained to compute the best model. Using the best results obtained from the grid search, the best overall model was standardized using the best features and retrained with the optimal parameters of the Random Forest Classifier. Then, a neural network was built once again.

C. Neural Networks

Different kinds of neural networks were created. The process started with a very simple model featuring a few layers, ReLU activation, and a final sigmoid layer. Then, several GNN models were trained with different learning rates, parameters, and loss functions. However, the main issues of this project concern these models, and all the details will be explained in the next section VI.

VI. Results

A. First Graph

As described in the table I, Logistic Regression, Random Forest, and SVM were employed. No neural network was utilized due to the

trivial nature of the task, to avoid unnecessary time loss. Here for the first Graph all the model where trained with the same number of epochs.

TABLE I: Graph one Book-to-Book Trivial Case

Model	Features	Accuracy	F1-score
Logistic Regression	All	0.744	0.803
Logistic Regression	Selecting the Best	0.716	0.813
Random Forest	Best Features	0.739	0.805
Random Forest	All	0.748	0.814
SVM	All	0.718	0.807

For this trivial task, the results obtained here are not outstanding, but they fall within the expected range for this type of model. Further explanation is unnecessary, considering that this model does not contain a significant amount of interesting information.

B. Second Graph

With the second graph, there are two parts described in tables II and III. The first table shows the results of the second graph using only the properties of this graph as features. On the ther side, the second table, includes the statistics of the books as explained in section V-B.

TABLE II: Graph Book-to-Book Only with graph properties

Model	Features	Accuracy	F1-score
Logistic Regression	Some properties	0.746	0.683
Logistic Regression	All properties	0.746	0.683
Logistic Regression	Best features	0.754	0.692
Simpler Neural Model	Best features (2 L.)	0.747	0.65
Simpler Neural Model	Best features (3 L.)	0.754	0.69

Overall, the results suggest that Logistic Regression and the simpler neural model with three layers performed similarly well, achieving the highest accuracy and F1-score when using the best features. Indeed, it appears that there isn't a significant change in the results when using all features versus using only the best features. This suggests that no single feature can entirely describe whether a book will be a bestseller or not. Instead, it's likely a combination of multiple factors that contribute to a book's success. As usual, all combinations of the features were tested in order to obtain the best combination and achieve the best results.

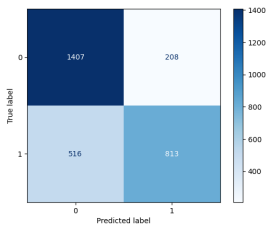


Fig. 5: Confusion Matrix for graph properties as features

TABLE III: Graph Book-to-Book adding books statistics

Model	Features	Accuracy	F1-score
Logistic Regression	All Features	0.809	0.775
Logistic Regression	Best Features	0.821	0.787
Random Forest	Best Features	0.829	0.800
Random Forest	Grid Search	0.830	0.799
Random Forest	Best + Standardize	0.833	0.803
MLP	All + Standardize	0.755	0.687
GCN	All	0.443	0.613

From these results, conclusion is that Random Forest with the best features and standardization achieved the highest accuracy and F1-score, indicating its effectiveness in predicting whether a book will be a bestseller or not. The results reveal an ongoing issue with GCN and GNN predictions mostly being 1 , indicating a potential problem with the model architecture or training process. The best model reach an accuracy of 64.5% but a bad F1 score. Further investigation into model design, hyperparameters, and data preprocessing was conducted to address this issue. Several solutions were attempted, as documented in https://github.com/fpalmisa/NML_2024_Project/blob/main/GNN.ipynb, but unfortunately, the problem persists.

Possible sources of this issue in machine learning tasks include overfitting, which was addressed by testing simpler models to check for it. Additionally, different learning rates and optimizers were explored, but the problem persisted. Early stopping was implemented to assess its impact, but without success. The problem may also lie in the model architecture, prompting the testing of various models, all without success. Data processing methods were also examined as a potential source of the issue, including different approaches to managing nodes and edges.

VII. Critical Analysis

First of all, the results are within the expected range for linear models. Significantly better accuracy could be expected with neural networks, but issues with predictions are a real limitation in obtaining more precise results. It is well-known in machine learning that convolutional and neural networks can lead to outstanding results.

Several aspects could be improved in this project. For example, the threshold to determine if a book is a bestseller is set arbitrarily. It might make more sense to determine it by the percentage of "to read" ratings instead of just using an arbitrary number of ratings. Another limitation that could lead to better results is the handling of the edges. More features could be added to the edges, such as the mean ratings per user as edge embeddings, and so on.

Furthermore, reducing the size of the dataset decreases the quality of the data and, consequently, the quality of the results. With access to the complete dataset and more resources, the results could potentially be much better. This reduction in dataset size likely explains the less convincing outcomes observed for all models.

VIII. Summary

This project aimed to predict whether a book would be a bestseller using various machine learning models and graph-based features. The results obtained were within the expected range for linear models, and while neural networks showed potential for higher accuracy, issues with prediction consistency limited their effectiveness.

Key findings :

- The use of logistic regression and random forest models yielded reasonable accuracy and F1-scores.
- Adding statistical features of books improved model performance.
- Graph-based features provided useful insights but were limited by the size and complexity of the dataset.
- Neural network models, including GNNs, struggled with prediction consistency, indicating potential areas for further investigation and improvement

Several areas for improvement were identified, including refining the threshold for determining bestsellers, enhancing edge feature management, and utilizing the complete dataset to improve data quality and model performance.

In conclusion, the project more or less successfully demonstrated the application of various machine learning models to predict book bestsellers.

IX. GitHub Link Repository

https://github.com/fpalmisa/NML_2024_Project/tree/main

REFERENCES

[1] Pascal Frossard Dorina Thanou. Network machine learning, 2024. EPFL Network machine learning, Fall 2024. URL: https://edu.epfl.ch/coursebook/en/network-machine-learning-EE-452.

[2] Inconnu. Qu'est-ce qu'un best-seller pour un livre ?, 2024. URL: https://www.chronobook.fr/best-seller-livre/.

[3] R.C. Quand parle-t-on d'un best-seller pour un livre ?, 2022. URL: https://www.commentecrire.fr/blog/quand-parle-t-on-dun-best-seller-pour-un-livre/.