**A MINI PROJECT REPORT**

**ON**

**“Goodreads: Books analysis & similar books recommendation”**



**Sinhgad College of Engineering**

**Department of Computer Engineering**

**SKILL DEVELOPMENT LABOTATORY**

**(T.E Computer Engineering)**

**SUBMITTED BY**

**STUDENT NAME** **ROLLNO**

Nitin A. Dixit 305073

Vasant R. Donukar 305074

Sameer Shah 305075

**YEAR 2019-20**

**CERTIFICATE**



**“Goodreads: Books analysis & similar books recommendation”**

**Sinhgad College of Engineering**

**Department of Computer Engineering**

**STUDENT NAME** **ROLLNO**

Nitin A. Dixit 305073

Vasant R. Donukar 305074

Sameer Shah 305075

**Prof.Nalini Mhetre**   **Prof. M. P. Wankhade**

Internal Guide Head of Dept.

Department of Computer Engineering Department of Computer Engineering

**Dr. S.D. Lokhande**

Principal SCOE, Pune

**ACKNOWLEDGEMENT**

We feel great pleasure in expressing our deepest sense of gratitude and sincere thanks to our guide **Prof.Nalini Mhetre** for her valuable guidance during the Project work, without which it would have been very difficult task. I have no words to express my sincere thanks for valuable guidance, extreme assistance and cooperation extended to all the **Staff Members** of our Department.

This acknowledgement would be incomplete without expressing our special thanks to **Prof.** **M.P.Wankhade** **Head of the Department (Computer Engineering)** for his support during the work.

We would also like to extend our heartfelt gratitude to our **Principal, Dr. S. D. Lokhande** who provided a lot of valuable support, mostly being behind the veils of college bureaucracy.

Last but not least we would like to thank all the Teaching, Non- Teaching staff members of our Department, our parents and colleagues those who helped us directly or indirectly for completing this Project successfully.

**ABSTRACT**

The approach of recommender systems are based on the ground of users history of choices, likings and reviews, each of which is interpreted in order to project the future choices of the user. In this study, a novel and strong recommender system for the books is proposed. A content based book recommendation application was developed which makes recommendations according to user's taste and choices.

The basic idea behind analysing the Goodreads dataset is to get a fair idea about the relationships between the multiple attributes a book might have, such as the aggregrate rating of each book, the most rated books and books with more pages, etc.

We've always conisdered the magical persona books seem to hold, and with this notebook, we step out on a journey to see what kind of books really drives people to read in this era of modern smart devices.

With such a vast, overwhelming number of factors, we'll go over such demographics:

* Does any relationship lie between average\_ratings and the total ratings given?
* Where do majority of the books lie, in terms of ratings - Does reading a book really bring forth bias for the ratings?
* Do number of pages make an impact on reading styles, ratings and popularity?

Can books be recommended based on ratings? Is that a factor which can work?

**INDEX**

**1.Introduction**

1.1.Introduction

1.2.Problem Statement

1.3.Project Scope

**2.Project Planning and Management**

2.1.Software Requirements

2.2.Hardware Requirements

2.3.Process model

**3. Analysis and design**

3.1 Use-Case Diagrams

3.2 State Transition diagram

3.3 Sequence Diagrams /DFD & CFDs

**4. Implementation and coding**

* 1. Methodology
  2. Database schema
  3. Algorithms/Flow Charts
  4. GUI design/screen shots

**5. Results and discussion**

5.1. Visualization of results (Graphs, Charts, etc.)

**6. Conclusion**

**7. References**

**List of Figures**

|  |  |  |
| --- | --- | --- |
| Sr.No | Name | Page No |
| 1 | Use case diagram | **4** |
| 2 | State Transition diagram | **5** |
| 3 | Data flow diagram | **6** |
| 4 | Goodreads\_dataset | **8** |
|  |  |  |

**1. INTRODUCTION**

**1.1. Background And Basics :**

|  |  |  |  |
| --- | --- | --- | --- |
| |  | | --- | | The basic idea behind analysing the Goodreads dataset is to get a fair idea about the relationships between the multiple attributes a book might have, such as the aggregrate rating of each book, the most rated books and books with more pages, etc.  We've always conisdered the magical persona books seem to hold, and with this notebook, we step out on a journey to see what kind of books really drives people to read in this era of modern smart devices.  With such a vast, overwhelming number of factors, we'll go over such demographics:   * Does any relationship lie between average\_ratings and the total ratings given? * Where do majority of the books lie, in terms of ratings - Does reading a book really bring forth bias for the ratings? * Do number of pages make an impact on reading styles, ratings and popularity? * Can books be recommended based on ratings? Is that a factor which can work? | | **1.2.Problem Statement :**  The need to analyze Goodreads dataset is to provide simple view towards complicated books data. This project analyzes goodreads books data to get a fair idea about the relationships between the multiple attributes a book might have, such as the aggregrate rating of each book, the most rated books and books with more pages, etc. The project should be able to give accurate visualization of CSV data used. Another aspect of this project is to help avid readers find books similar to their taste and save their time from searching online and reading the recommendations. This project would help avid reader to decide what to read next without wasting much time. As the user gives input of books name or the book id ( in case he/she doesn’t remembers complete title..as not everyone always rememberes) then we are able to recommend 5 different books name which are similar to that book. | |  | |

**1.3. Project Scope:**

This project is focused on giving the clear and simple visualization of books recommendation in text form. It does clustering based on relationship between average\_rating and rating\_count. This will require dataset of Goodreads in CSV format and packages like dash (GUI), plotly and matplotlib, sklearn, etc. In future we are planning to include recommendation based on more complex and accurate machine learning methods like hybrid algorithm which combines both collaborative and content based filtering methods. Besides, we are planning to recommend books to users based on his/her past behaviour, like the books which they have already read and recommending the similar taste books.

**2. PROJECT PLANNING & MANAGEMENT**

**2.1.Software Requirements :**

* Anaconda Spyder

**2.2.Hardware Requirements :**

* Intel® Core™  i5 processor 3210M at 2.50 GHz
* 8 GB of RAM
* Disk space:2 to 3 GB
* Operating system: Windows 10 or Linux

**2.3.Process Model :**

1. Read CSV data of goodreads.

2. Calculate and analyse the data present in dataset.

3. Display infographics in visual form using bar graphs, histograms,scatter plots, etc.

4. recommend books based on user input.

**Process Model in Visual Form:**

recommend books based on user input.

Display certain infographics( graphs, bar plot)

Calculate and

analyse data

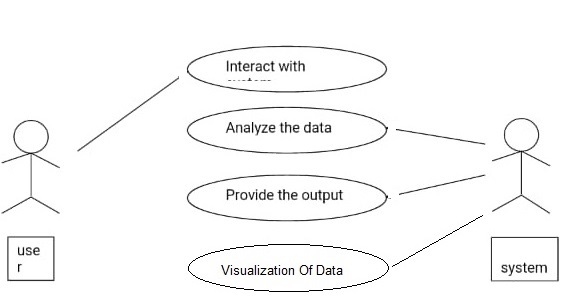
Reading

CSV file

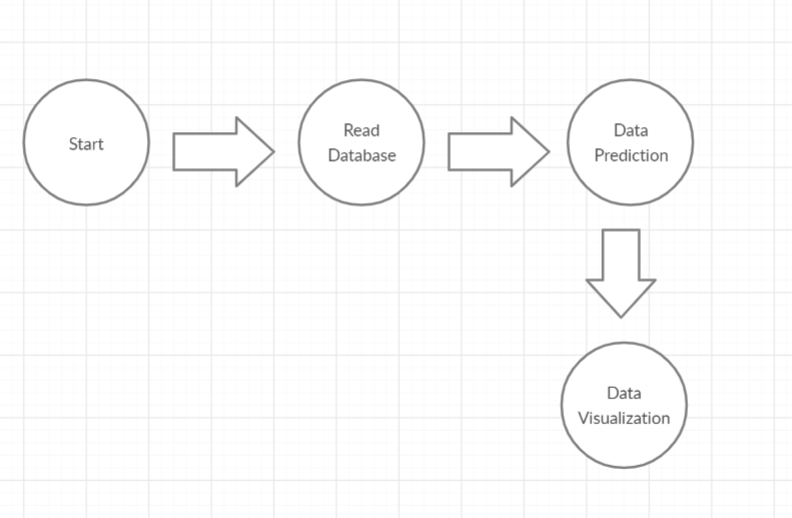
CSV data of Goodreads

**3. ANALYSIS & DESIGN**

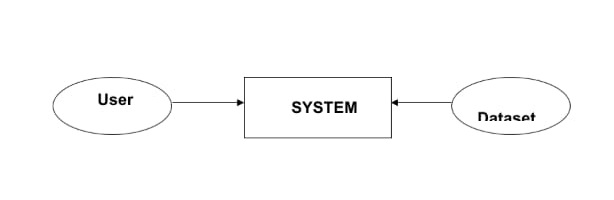
**3.1 Use-Case Diagrams:**



**3.2 State Transition Diagram :**

****

**3.3 DFD Diagram :**

****

**4. IMPLEMENTION & CODING**

**4.1. Methodology :**

This project is a dash application which represents data visualization of goodreads books data.

The basic idea behind analysing the Goodreads dataset is to get a fair idea about the relationships between the multiple attributes a book might have, such as the aggregrate rating of each book, the most rated books and books with more pages, etc.

Project implements dash framework to visualize demographics such as:

1.top 30 authors with most books

2. top 30 books with more pages

3. top 30 most rated books.. etc.

# Topic Modelling

### KMeans Clustering without outliers:

KMeans clustering is a type of unsupervised learning which groups unlabelled data. The goal is to find groups in data.

With this, we attepmt to find a relationship or groups between the rating count and average rating value.

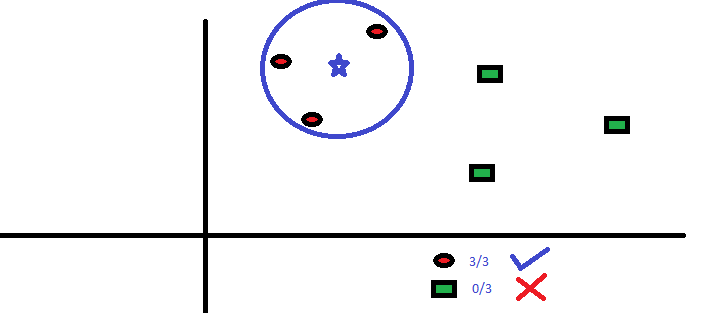
## Recommendation Engine:

Having seen the clustering, we infered that there can be some recommendations which can happen with the relation between Average Rating and Ratings Count.

Taking the Ratings\_Distribution (A self created classifying trend), this recommendation system works with the algortihm of K Nearest Neighbors.

Based on a book entered by the user, the nearest neighbours to it would be classified as the books which the user might like.

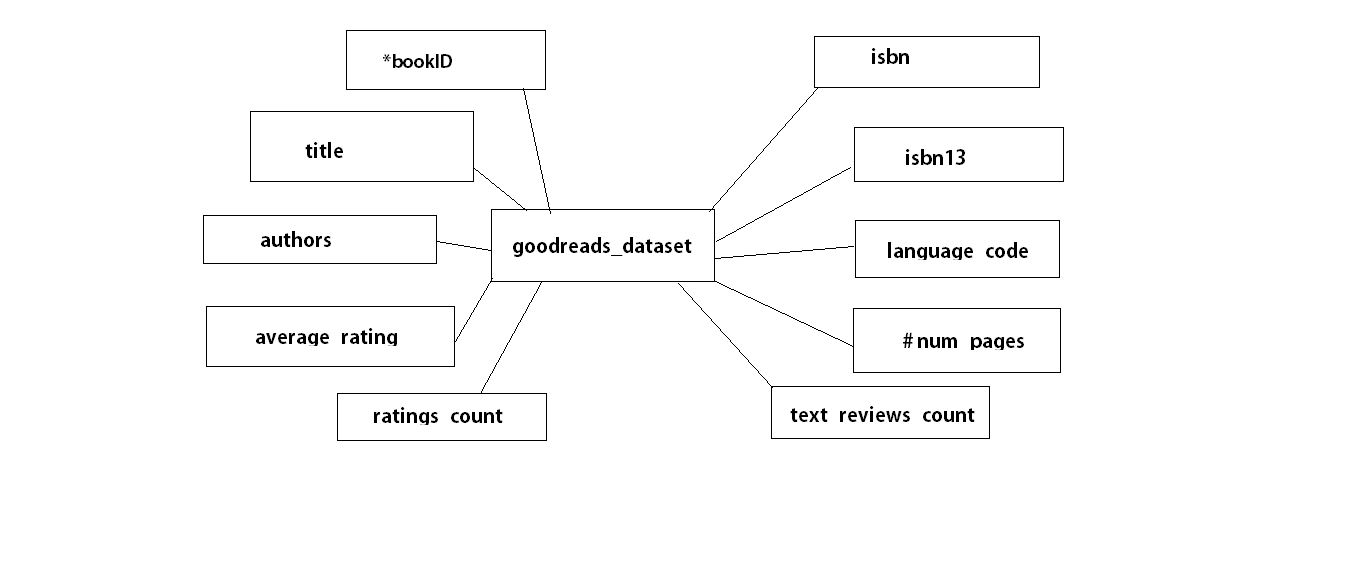
KNN is used for both classification and regression problems. In classification problems to predict the label of a instance we first find k closest instances to the given one based on the distance metric and based on the majority voting scheme or weighted majority voting(neighbors which are closer are weighted higher) we predict the labels.



In a setting such as this, the unsupervised learning takes place, with the similar neighbors being recommended. For the given list, if the user ask recommendations for "The Catcher in the Rye", five books related to it would appear.

**4.2 Database schema :**

* **Goodreads\_dataset :**

****

**4.3.Algorithms/Flowcharts :**

K-means clustering:-

1. Read CSV data of goodreads\_dataset.

df = pd.read\_csv(“books.csv”)

2. Calculate and analyse the data present in dataset.

trial = df[['average\_rating', 'ratings\_count']]

data = np.asarray([np.asarray(trial['average\_rating']), np.asarray(trial['ratings\_count'])]).T3. 3.Display result in visual form using bar charts, histograms,scatter plot etc.

go.Bar(x=most\_rated.index, y=most\_rated['ratings\_count'], marker\_color='#ff6c03' );

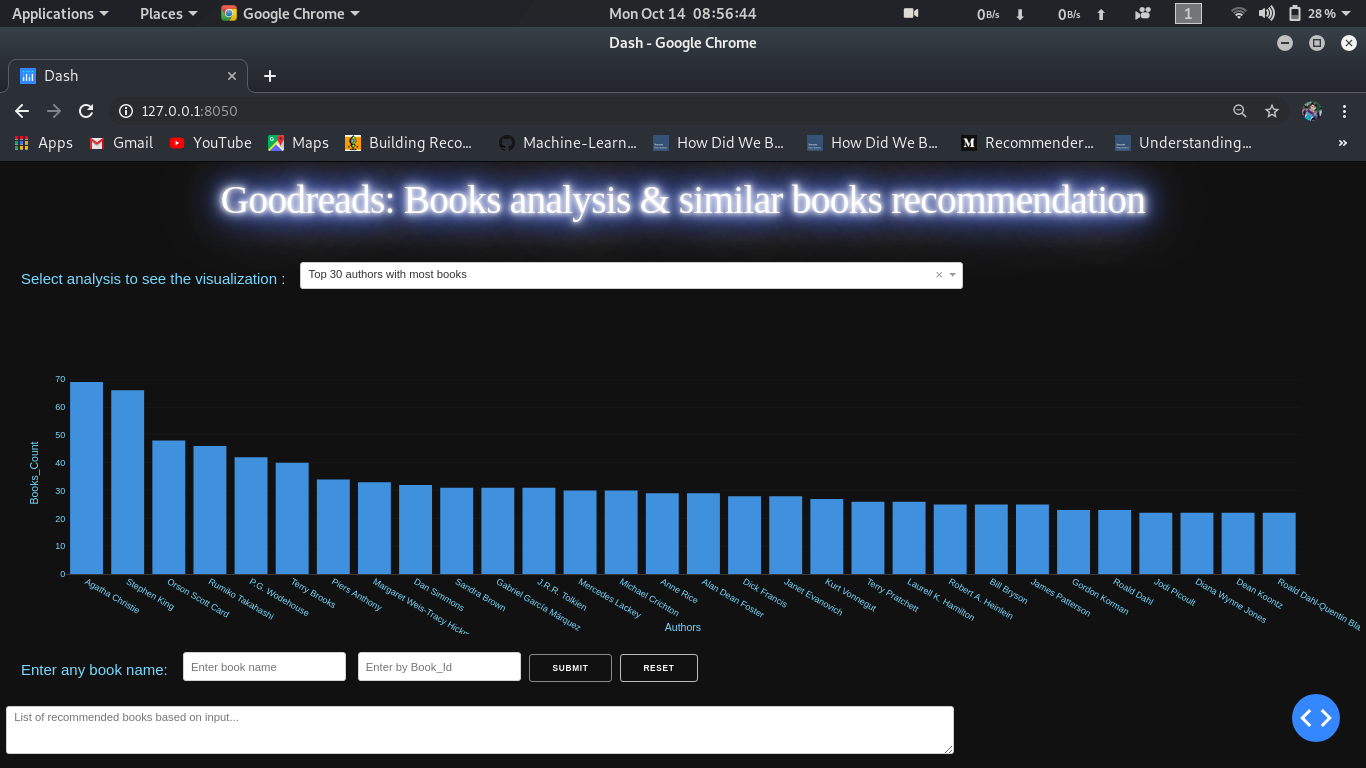
4. recommend books based on user input.

Input= “book\_title”/”book\_id”

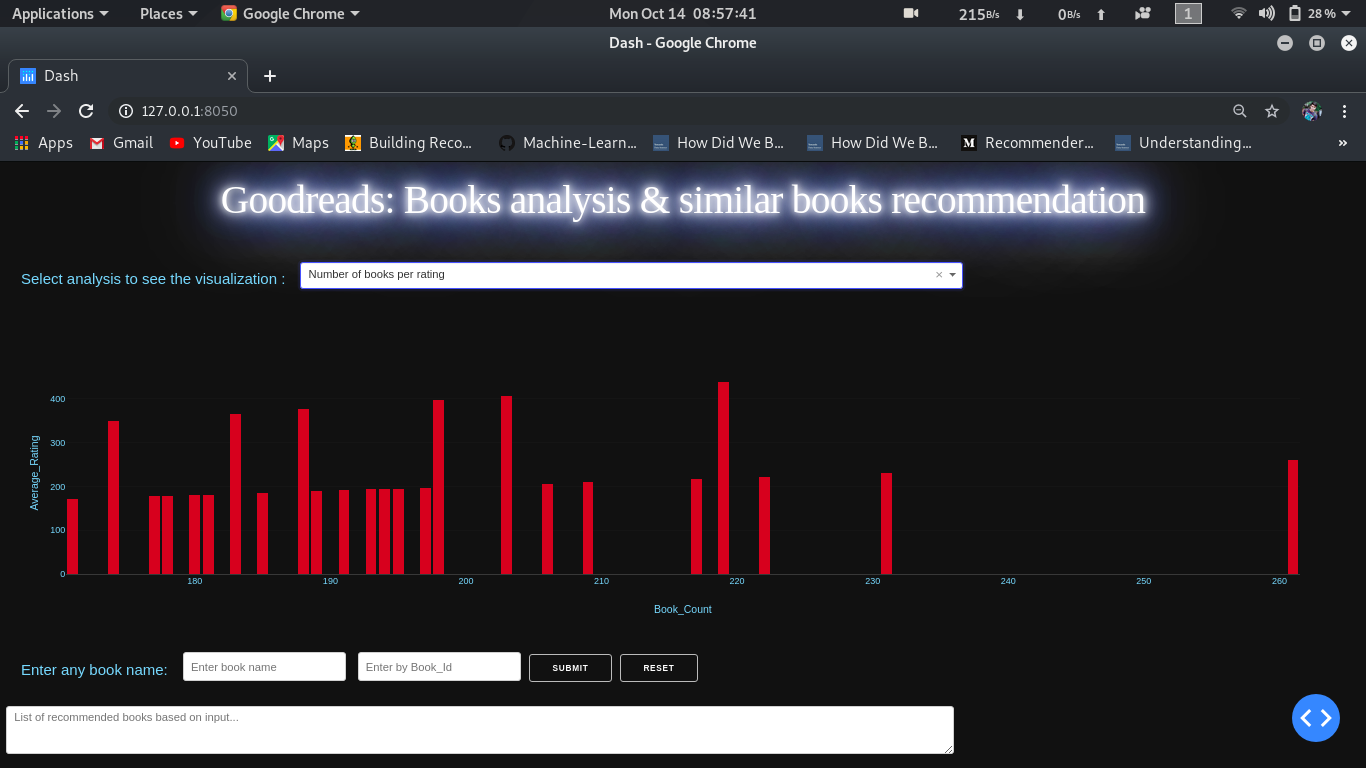
Output= 5 similar books recommendation

**4.4 GUI design/screen shots :**

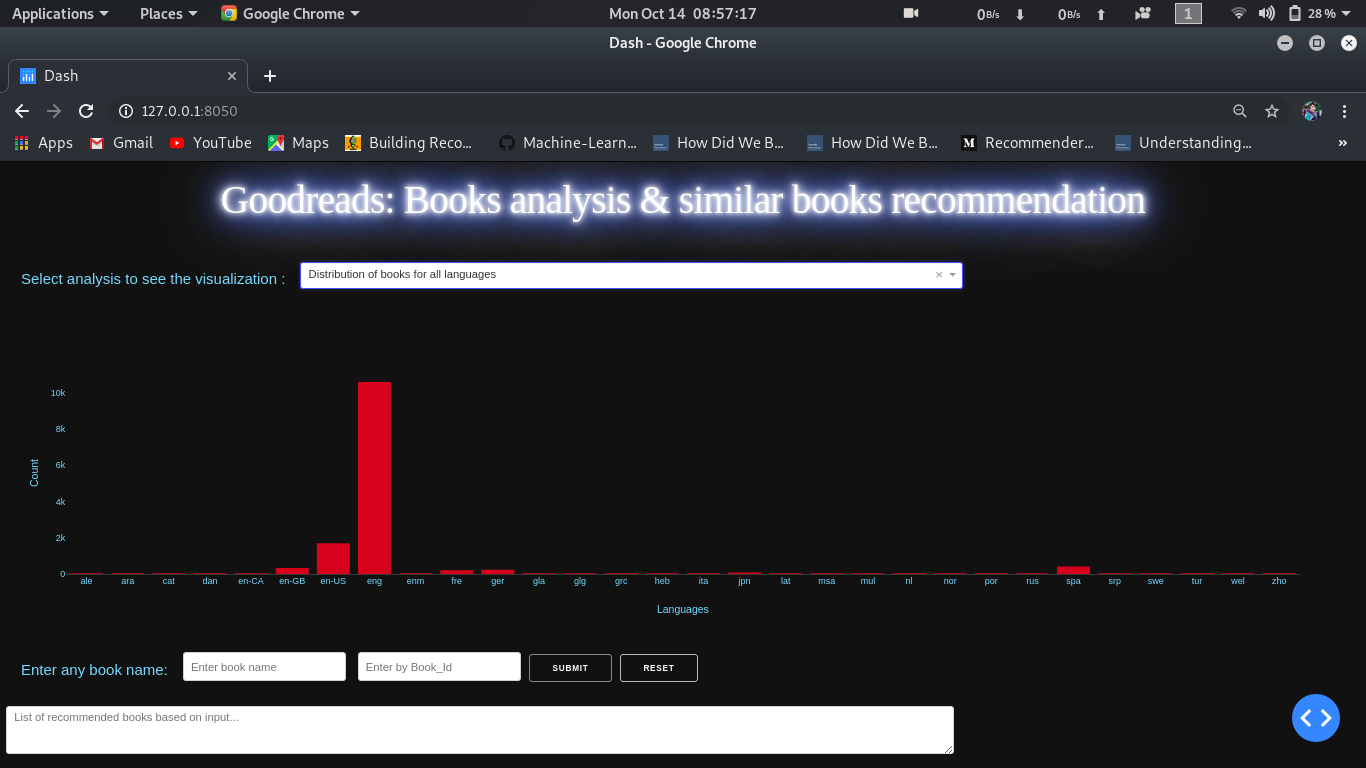
**4.4.1. Visualization of demographics:**



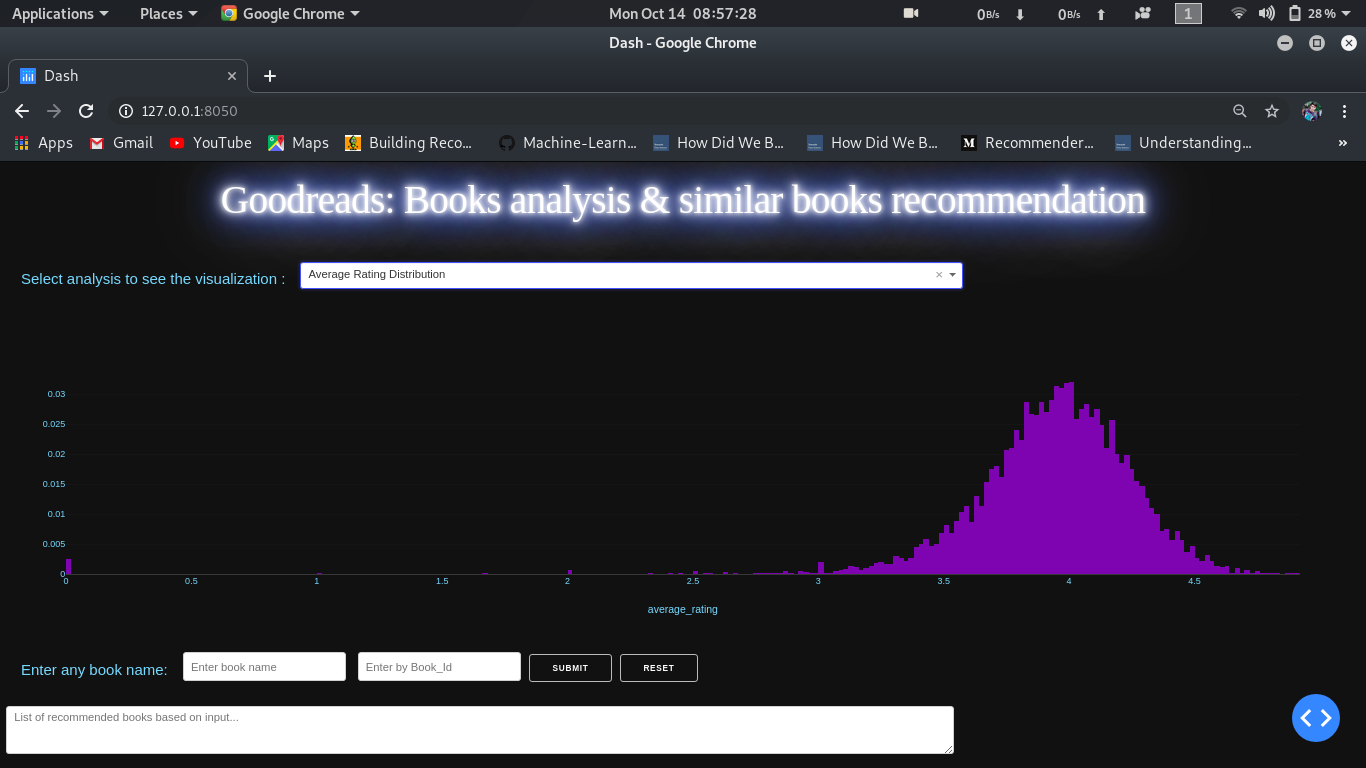
**Fig 4.4.1: Top 30 authors with most books**

The above screenshot is the homepage of the project. It shows 30 authors with most books. We can see from the above plot that Agatha Christie has the most number of books in the list - although a lot of them might be just various publications for the same book, considering the fact that her work has been here for quite a while, spanning decades. On selecting one of the visualization in dropdown menu , demographics are displayed according to the selection in bar chart, histograms and scatter plots. 

**Fig 4.4.2: Number of books per rating**

****

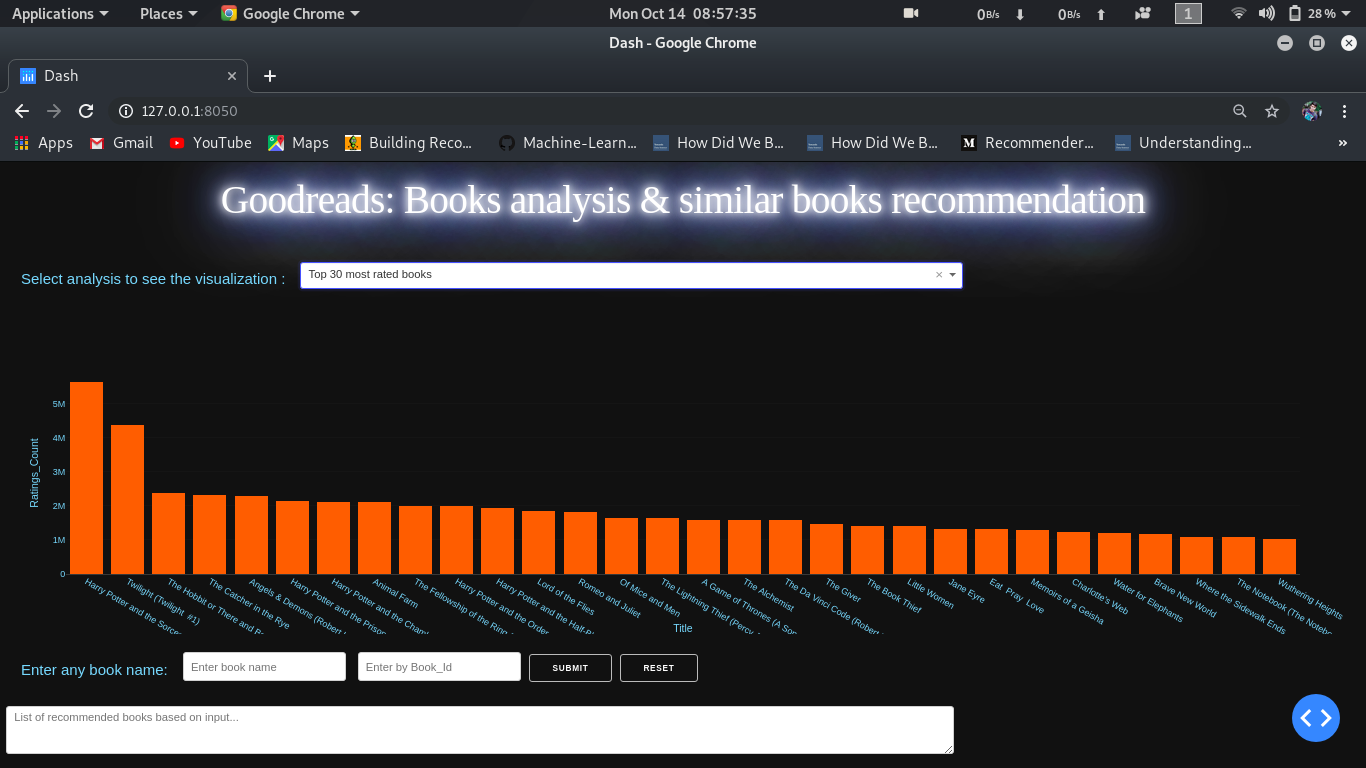
**Fig 4.4.2:Distribution of books for all lang**



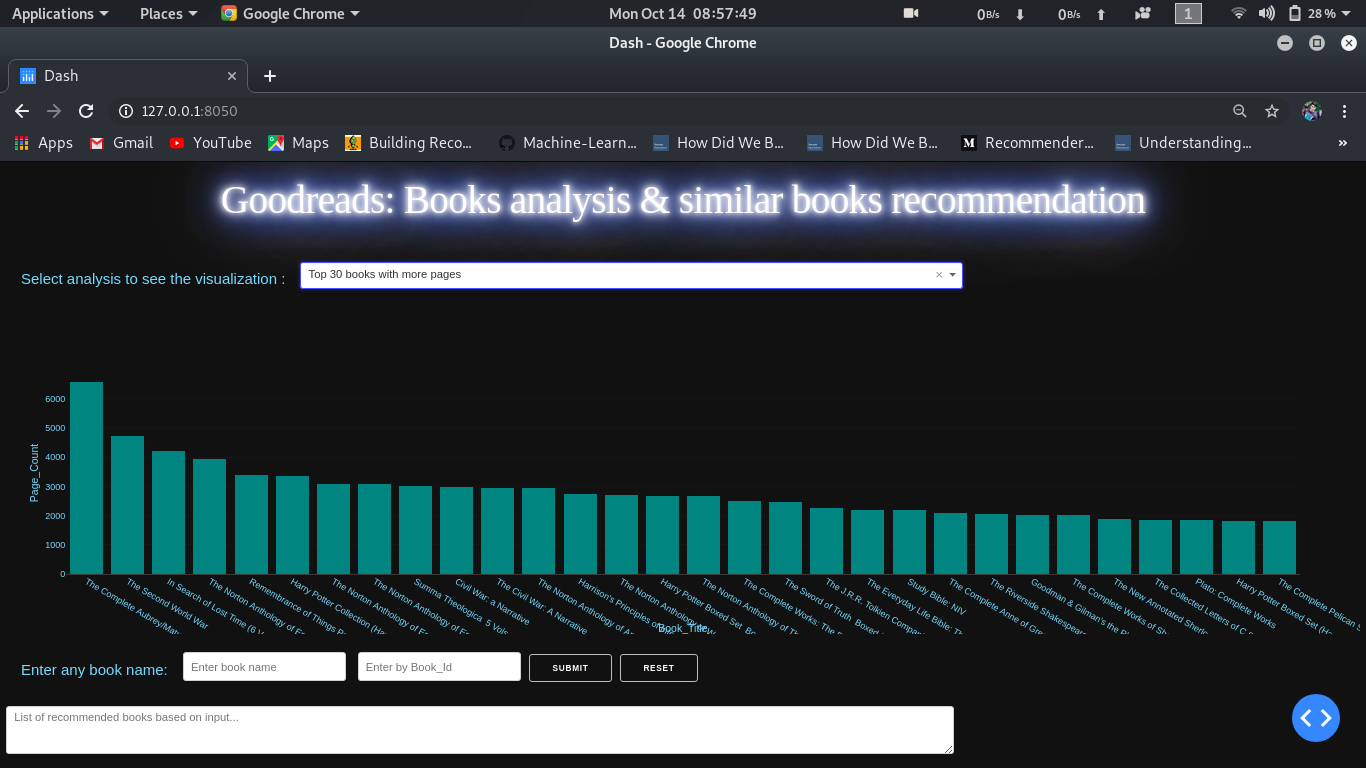
**Fig 4.4.3: Average rating distribution**

From the given plot, we can infer that:

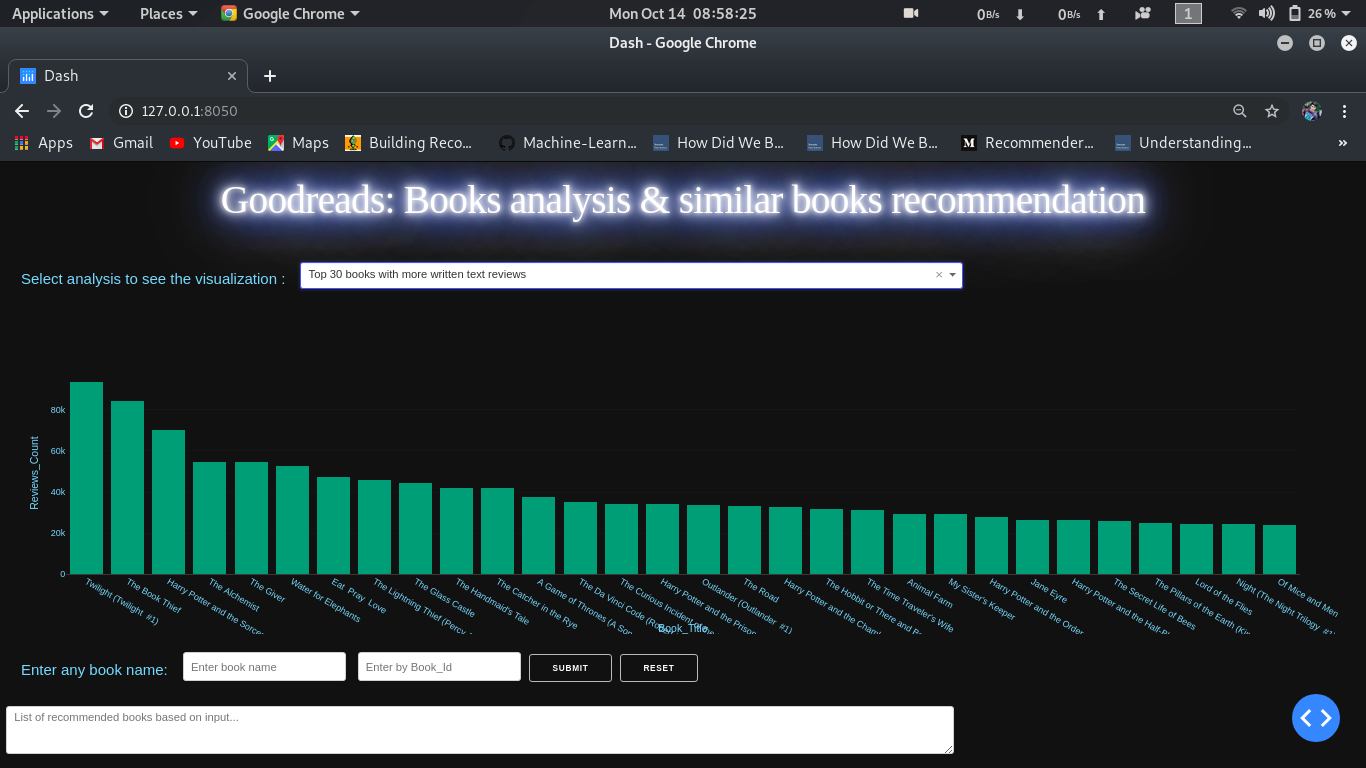
* Majority of the ratings lie near 3.7-4.3, approximately.
* Books having scores near 5 are extremely rare

****

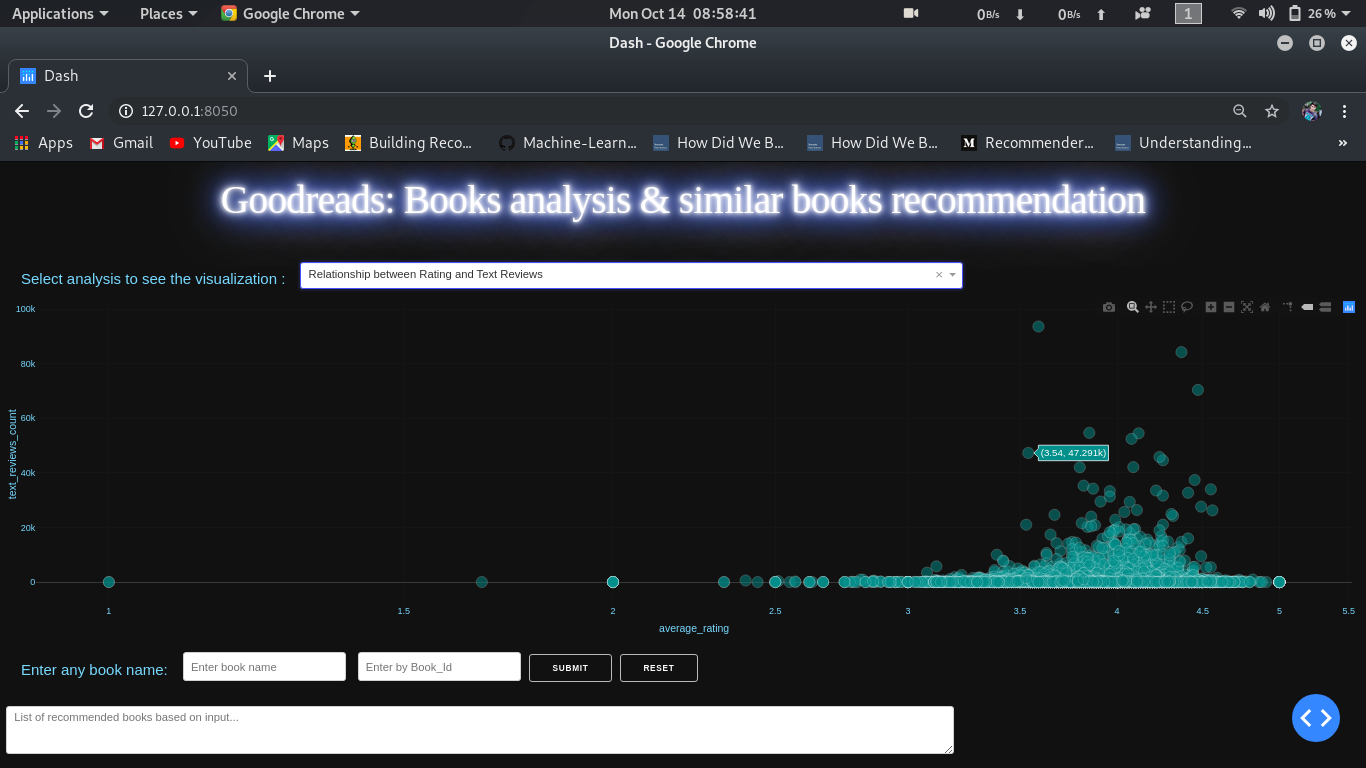
**Fig 4.4.4: top 30 most rated books**

****

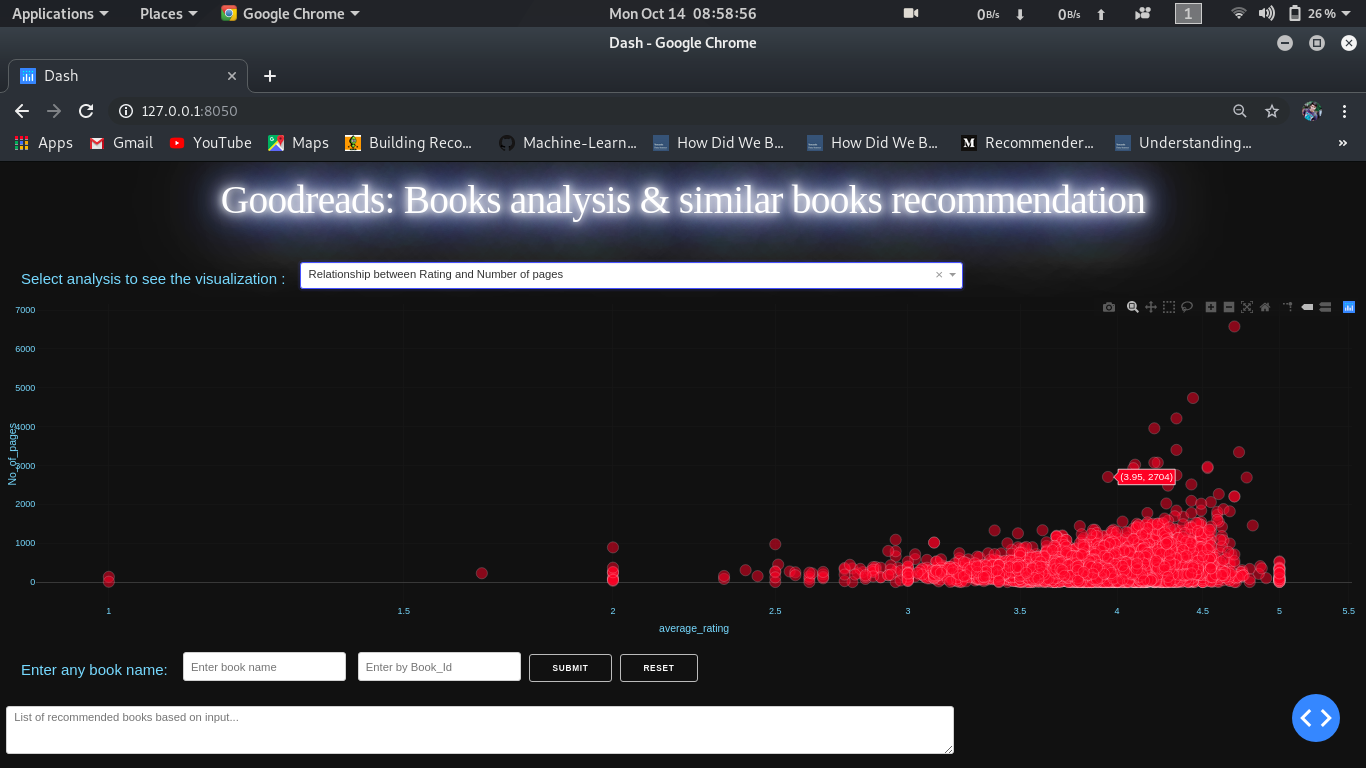
**Fig 4.4.5: top 30 books with more pages**

****

**Fig 4.4.6: top 30 books with text reviews**

****

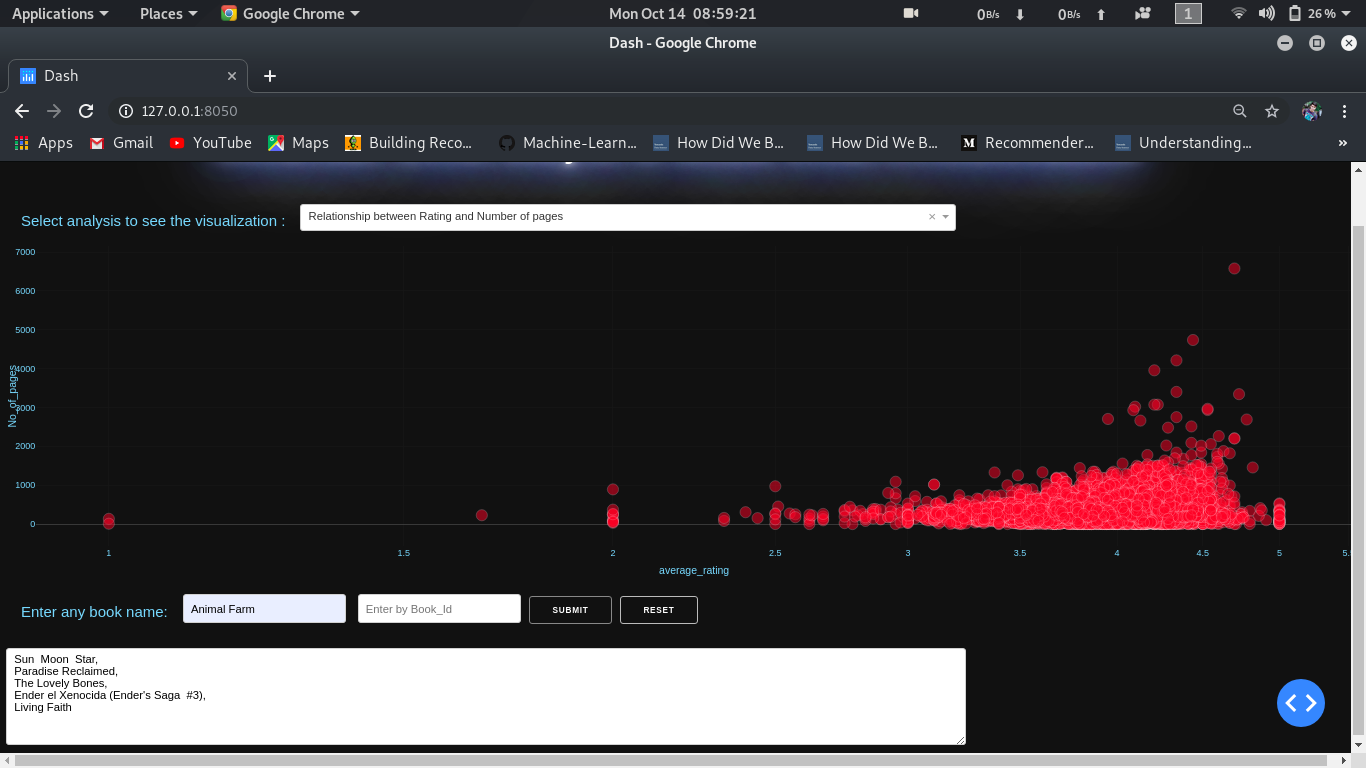
**Fig 4.4.7: relationship between rating and text reviews**

****

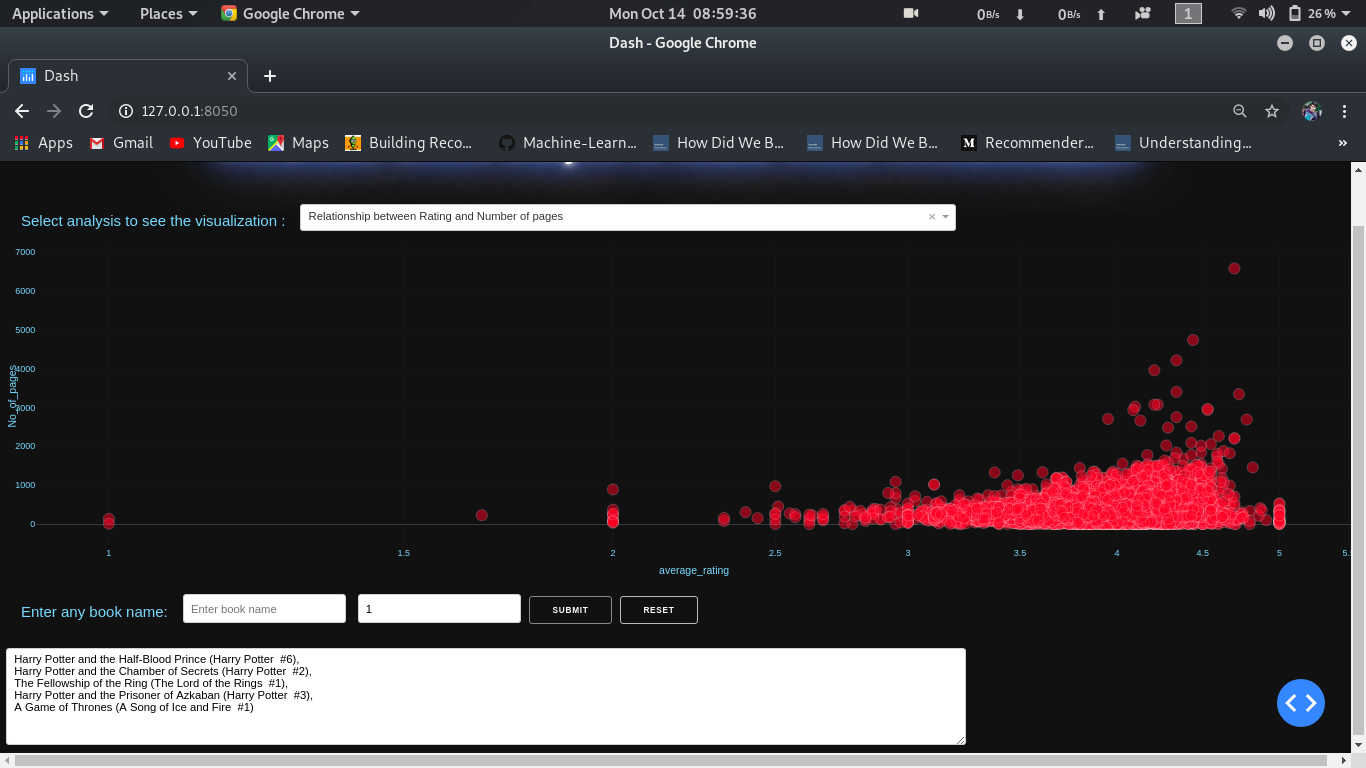
**Fig 4.4.8: relationship between rating and no. of pages**

**5. RESULTS AND DISSCUSSION**

**5.1. Visualization of results:**



**Fig 5.1.2: Recommending books based on user input books title**

****

**Fig 5.1.2: Recommending books based on user input books id**

6. CONCLUSION

There are various recommendation system available but not much for book recommendation. This simple and easy book recommendation uses k-means clustering and nearest neigbors method to recommend similar taste books to avid users. Saving their time and efforts to ask others what to read next

The approach of recommender systems are based on the ground of average\_rating and rating count relationship and clustering each of which is interpreted in order to project the future recommendation of the user. In this study, a novel and strong recommender system for the books is proposed. A content based book recommendation application was developed which makes recommendations according to user's taste and choices. K-mean clustering improves the accuracy of clustering algorithm and is suitable.

This project is considered as the easily available and beneficial in terms of to get recommendation of similar taste books based on user input.

7. REFERENCES

1. <https://towardsdatascience.com/how-did-we-build-book-recommender-systems-in-an-hour-part-2-k-nearest-neighbors-and-matrix-c04b3c2ef55c>
2. <https://www.kaggle.com/jealousleopard/goodreadsbooks/kernels>
3. <https://towardsdatascience.com/how-did-we-build-book-recommender-systems-in-an-hour-the-fundamentals-dfee054f978e>
4. <https://www.youtube.com/watch?v=AJZJgrmr4bQ>