VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANA SANGAMA, BELAGAVI – 590 018

An Internship Project Report on

Predict The Price Of Books

Submitted in partial fulfilment of the requirements for the VIII Semester of degree of **Bachelor of Engineering in Information Science and Engineering** of Visvesvaraya Technological University, Belagavi

by

Vaishnavi S M

1RN18IS115

Under the Guidance of

Mrs. Shwetha G N

Associate Professor

Department of ISE

Department of Information Science and Engineering

RNS Institute of Technology

Dr. Vishnuvaradhan Road, Rajarajeshwari Nagar post, Channasandra, Bengaluru-560098

2021-2022

RNS INSTITUTE OF TECHNOLOGY

Dr. Vishnuvaradhan Road, Rajarajeshwari Nagar post,

Channasandra, Bengaluru - 560098

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the Internship work entitled ***Predict The Price Of Books*** has been successfully completed by **Vaishnavi S M (1RN18IS115)** bonafide student of **RNS Institute of Technology, Bengaluru** in partial fulfilment of the requirements of 7th semester for the award of degree in **Bachelor of Engineering in Information Science and Engineering** of **Visvesvaraya Technological University, Belagavi** during academic year **2021-2022**. The internship report has been approved as it satisfies the academic requirements in respect of internship work for the said degree.

Mrs. Shwetha GN	**Dr. Suresh L**	**Dr. M K Venkatesha**
Internship Guide	Professor and HoD	Principal
Assistant Professor	Department of ISE	RNSIT
Department of ISE	RNSIT	

External Viva

Name of the Examiners		**Signature with Date**
1. _____	**1.** _____	_____
2. _____	**2.** _____	_____

DECLARATION

I, **Vaishnavi S M [USN: 1RN18IS115]** student of VII Semester BE, in Information Science and Engineering, RNS Institute of Technology hereby declare that the Internship work entitled ***Predict The Price of Books*** has been carried out by us and submitted in partial fulfilment of the requirements for the *VII Semester degree of* ***Bachelor of Engineering in Information Science and Engineering*** *of Visvesvaraya Technological University, Belagavi* during academic year 2021-2022.

Place : Bengaluru

Date : 12-01-21

VAISHNAVI S M (1RN18IS115)

ABSTRACT

As each and every sector of the market is growing, data is building up day by day, we need to keep the record of the data which can be helpful for the analytics and evaluation. Now we don't have data in gigabyte or terabyte but in zeta byte and petabyte and this data cannot be handled with the day to day software such as Excel or Matlab. Therefore, in this report we will be dealing with large data sets with the high-level programming language 'Python'.

The main goal of this project is to aggregate and analyze the data collected from the different data sets. This project mainly focuses on the usage of the python programming language in the field of data analysis and outcome prediction. This language has not only it's application in the field of just analyzing the data but also for the prediction of the upcoming scenarios.

The purpose of using this specific language is due to its versatility, vast libraries (Pandas, Numpy, Matplotlib, etc.), speed limitations, and ease of learning. We will be analyzing large data sets in this project which cannot be easily analyzed in other tools as compared to python. Python does not have its limitation to only data analytics but also in many other fields such as Artificial intelligence, Machine learning, and many more.

ACKNOWLEDGMENT

At the very onset I would like to place our gratefulness to all those people who helped me in making the Internship a successful one.

Coming up, this internship to be a success was not easy. Apart from the sheer effort, the enlightenment of the very experienced teachers also plays a paramount role because it is they who guided me in the right direction.

First of all, I would like to thank the **Management of RNS Institute of Technology** for providing such a healthy environment for the successful completion of internship work.

In this regard, I express sincere gratitude to our beloved Principal **Dr. M K Venkatesha,** for providing us all the facilities.

We are extremely grateful to our own and beloved Professor and Head of Department of Information science and Engineering, **Dr. Suresh L**, for having accepted to patronize me in the right direction with all her wisdom.

We place our heartfelt thanks to **Mrs. Shwetha GN,** Assistant Professor, Department of Information Science and Engineering for having guided internship and all the staff members of the department of Information Science and Engineering for helping at all times.

I thank **Mr. Satyendra Nath, Data Scientist at LocalHost Tecnology**, for providing the opportunity to be a part of the Internship program and having guided me to complete the same successfully.

I also thank our internship coordinator **Dr. R Rajkumar,** Associate Professor, Department of Information Science and Engineering. I would thank my friends for having supported me with all their strength and might. Last but not the least, I thank my parents for supporting and encouraging me throughout. I have made an honest effort in this assignment.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVATIONS

SQL	-	Structured Query Language
SAS	-	Static Analysis System
API	-	Application programming interface
XGboost	-	eXtreme Gradient Boosting
UI	-	User Interface
MATLAB	-	MATrix LABoratory
SDK	-	Software Development Kit

INTRODUCTION

1.1 Data Science

Data science is the field of data analytics and data visualization in which raw data or the unstructured data is cleaned and made ready for the analysis purpose. Data scientists use this data to get the required information for the future purpose. “Data science uses many processes and methods on the big data, the data may be structured or unstructured”. Data frames available on the internet is the raw data we get. It may be either in unstructured or semi structured format. This data is further filtered, cleaned and then number of required task are performed for the analysis with the use of the high programming language. This data is further analyzed and then presented for our better understanding and evaluation.

One must be clear that data science is not about making complicated models or making awesome visualization neither it is about writing code but about using the data to create an impact for your company, for this impact we need tools like complicated data models and data visualization. There are many tools used to handle th**e big data available to us.** “Data scientists use programming tools such as Python, R, SAS, Java, Perl, and C/C++ to extract knowledge from prepared data”.

Data scientists use many algorithms and mathematical models on the data. Following are the stages and their cycle performed on the unstructured data.

- Identifying the problem.
- Identify available data sources
- Identify available data sources
- Identify if additional data sources are needed.
- Statistical analysis
- Implementation, development
- Communicate results
- Maintenance

Data science finds its application in many fields. With the assistance of data science, it is easy to get the search query on search engines in plenty of time. A role of the data scientist is to have a deep understanding of the data as well as a good command on the programming language, he should also know how to work with the raw data extracted from the data source. Many programming languages are used to analyze and evaluate the data such as Python, Java, MATLAB, Scala, Julia, R., SQL and TensorFlow. Among which python is the most user friendly and vastly used programming language in the field of data science. This life cycle is applied in each and every field, in this project we will be considering all this seven stages of data science to analyze the data. The process will be starting from data collection, data preparation, data modeling and finally data evaluation. Data Science continues to be a hot topic among skilled professionals and organizations that are focusing on collecting data and drawing meaningful insights out of it to aid business growth. A lot of data is an asset to any organization, but only if it is processed efficiently. The need for storage grew multifold when we entered the age of big data. Until 2010, the major focus was towards building a state of the art infrastructure to store this valuable data, that would then be accessed and processed to draw business insights. With frameworks like Hadoop that have taken care of the storage part, the focus has now shifted towards processing this data. Let us see what is data science, and how it fits into the current state of big data and businesses. Broadly, Data Science can be defined as the study of data, where it comes from, what it represents, and the ways by which it can be transformed into valuable inputs and resources to create business and IT strategies.

.

1.2 Proposed System

The proposed system uses XGBoost for regression. Extreme Gradient Boosting (XGBoost) is an open-source library that provides an efficient and effective implementation of the gradient boosting algorithm. Shortly after its development and initial release, XGBoost became the go-to method and often the key component in winning solutions for a range of problems in machine learning competitions. Regression predictive modeling problems involve predicting a numerical value such as a dollar amount or a height. XGBoost can be used directly for regression predictive modeling.

Gradient boosting refers to a class of ensemble machine learning algorithms that can be used for classification or regression predictive modeling problems. Ensembles are constructed from decision tree models. Trees are added one at a time to the ensemble and fit to correct the prediction errors made by prior models. This is a type of ensemble machine learning model referred to as boosting. Models are fit using any arbitrary differentiable loss function and gradient descent optimization algorithm. This gives the technique its name, “gradient boosting,” as the loss gradient is minimized as the model is fit, much like a neural network. The two main reasons to use XGBoost are execution speed and model performance.

ANALYSIS

2.1 Introduction

“Python is an interpreted, object-oriented, high-level programming language with dynamic semantics”. This language consists of mainly data structures which make it very easy for the data scientists to analyze the data very effectively. It does not only help in forecasting and analysis it also helps in connecting the two different languages. Two best features of this programming language is that it does not have any compilation step as compared to the other programming language in which compilation is done before the program is being executed and other one is the reuse of the code, it consists of modules and packages due to which we can use the previously written code anywhere in between the program whenever is required. There are multiple languages for example R., Java, SQL, Julia, Scala, MATLAB available in market which can be used to analyze and evaluate the data, but due to some outstanding features python is the most famous language used in the field of data science.

Python is mostly used and easy among all other programming languages. Books are the most important friends in one’s life. The so-called paradoxes of an author, to which a reader takes exception, often exist not in the author’s book at all, but rather in the reader’s head. — Friedrich Nietzsche Books are open doors to the unimagined worlds which is unique to every person. It is more than just a hobby for many. There are many among us who prefer to spend more time with books than anything else. Here we explore a big database of books. Books of different genres, from thousands of authors. In this article, we will use the dataset to build a Machine Learning model to predict the price of books based on a given set of features.

Fig1.1 XGBoost Algorithm

2.2 XGBoost Regression

Extreme Gradient Boosting, or XGBoost for short, is an efficient open-source implementation of the gradient boosting algorithm. As such, XGBoost is an algorithm, an open-source project, and a Python library. It was initially developed by Tianqi Chen and was described by Chen and Carlos Guestrin in their 2016 paper titled "XGBoost: A Scalable Tree Boosting System." It is designed to be both computationally efficient (e.g. fast to execute) and highly effective, perhaps more effective than other open-source implementations. The two main reasons to use XGBoost are execution speed and model performance. XGBoost dominates structured or tabular datasets on classification and regression predictive modeling problems. The evidence is that it is the go-to algorithm for competition winners on the Kaggle competitive data science platform.

Now XGboost owns the ability to handle both types of situations, whether you need to go with regression or classification modelling. So, we can consider XGboost both as a classification and regression algorithm.

2.3 XGBoost Features

- **Regularized Learning:** Regularization term helps to smooth the final learnt weights to avoid over-fitting. The regularized objective will tend to select a model employing simple and predictive functions.
- **Gradient Tree Boosting:** The tree ensemble model cannot be optimized using traditional optimization methods in Euclidean space. Instead, the model is trained in an additive manner.
- **Shrinkage and Column Subsampling:** Besides the regularized objective, two additional techniques are used to further prevent overfitting. The first technique is shrinkage introduced by Friedman. Shrinkage scales newly added weights by a factor η after each step of tree boosting. Similar to a learning rate in stochastic optimization, shrinkage reduces the influence of each tree and leaves space for future trees to improve the model.

2.4 XGBoost Regression API

XGBoost can be installed as a standalone library and an XGBoost model can be developed using the scikitlearn API. The first step is to install the XGBoost library if it is not already installed. This can be achieved using the pip python package manager on most platforms; for example: sudo pip install xgboost

You can then confirm that the XGBoost library was installed correctly and can be used by running the following script.

import xgboost print(xgboost.__version__)

Running the script will print your version of the XGBoost library you have installed.
Your version should be the same or higher. If not, you must upgrade your version of the XGBoost library.
It is possible that you may have problems with the latest version of the library. It is not your fault.
Sometimes, the most recent version of the library imposes additional requirements or may be less stable. If you do have errors when trying to run the above script, I recommend downgrading to version 1.0.1 (or lower). This can be achieved by specifying the version to install to the pip command, as follows:

sudo pip install xgboost==1.0.1

The XGBoost library has its own custom API, although we will use the method via the scikit-learn wrapper classes: XGBRegressor and XGBClassifier. This will allow us to use the full suite of tools from the scikit-learn machine learning library to prepare data and evaluate models.

An XGBoost regression model can be defined by creating an instance of the XGBRegressor class;
for example:

```
...
# create an xgboost regression model model = XGBRegressor()
```

You can specify hyperparameter values to the class constructor to configure the model. Perhaps the most commonly configured hyperparameters are the following:

- n_estimators: The number of trees in the ensemble, often increased until no further improvements are seen.
- max_depth: The maximum depth of each tree, often values are between 1 and 10.

- eta: The learning rate used to weight each model, often set to small values such as 0.3, 0.1, 0.01, or smaller.
- subsample: The number of samples (rows) used in each tree, set to a value between 0 and 1, often 1.0 to use all samples.
- colsample_bytree: Number of features (columns) used in each tree, set to a value between 0 and 1, often 1.0 to use all features.

2.5 Software requirement specification

The best thing about using Flutter for creating cross-platform native mobile apps is the fact that you can build those on almost any OS.

Here are some System Requirements for Android Studio which is needed for running an Android simulator.

Windows:

- Microsoft® Windows® 7/8/10 (64-bit)
- 4 GB RAM minimum, 8 GB RAM recommended
- 2 GB of available disk space minimum,
- 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)
- 1280 x 800 minimum screen resolution

To install and run Jupyter, your development environment must meet these minimum requirements:

- **Operating Systems**: Windows 7 SP1 or later (64-bit), x86-64 based.
- **Disk Space**: 1.64 GB (does not include disk space for IDE/tools).
- With PySpark (Team Studio version 6.2 and later)
  Memory and disk space required per user: 1GB RAM + 1GB of disk + .5 CPU core.
  Server overhead: 2-4GB or 10% system overhead (whatever is larger), .5 CPU cores, 10GB disk space.
  Port requirements: Port 8000 plus 5 unique, random ports per notebook.

- Without PySpark (Team Studio version 6.0 or 6.1)
  Memory and disk space required per user: 512MB RAM + 1GB of disk + .5 CPU core.
  Server overhead: 2-4GB or 10% system overhead (whatever is larger), .5 CPU cores, 10GB disk space.
  Port requirements: Port 8000.SYSTEM DESIGN

SYSTEM DESIGN

3.1 Process Flow

Process Flow Diagrams (PFDs) are **a graphical way of describing a process, its constituent tasks, and their sequence**. A PFD helps with the brainstorming and communication of the process design. The following diagrams help us understand the algorithm and its working.

Algorithm 1: XGboost algorithm

Data: Dataset and hyperparameters

Initialize $f_0(x)$;

for $k = 1, 2, \ldots, M$ **do**

Calculate $g_k = \frac{\partial L(y,f)}{\partial f}$;

Calculate $h_k = \frac{\partial^2 L(y,f)}{\partial f^2}$;

Determine the structure by choosing splits with maximized gain

$\mathbf{A} = \frac{1}{2}\left[\frac{G_L^2}{H_L} + \frac{G_R^2}{H_R} - \frac{G^2}{H}\right]$;

Determine the leaf weights $w^* = -\frac{G}{H}$;

Determine the base learner $\hat{b}(x) = \sum_{j=1}^{T} wI$;

Add trees $f_k(x) = f_{k-1}(x) + \hat{b}(x)$;

end

Result: $f(x) = \sum_{k=0}^{M} f_k(x)$

Fig3.1 Basic XGboost Algorithm mathematically represented

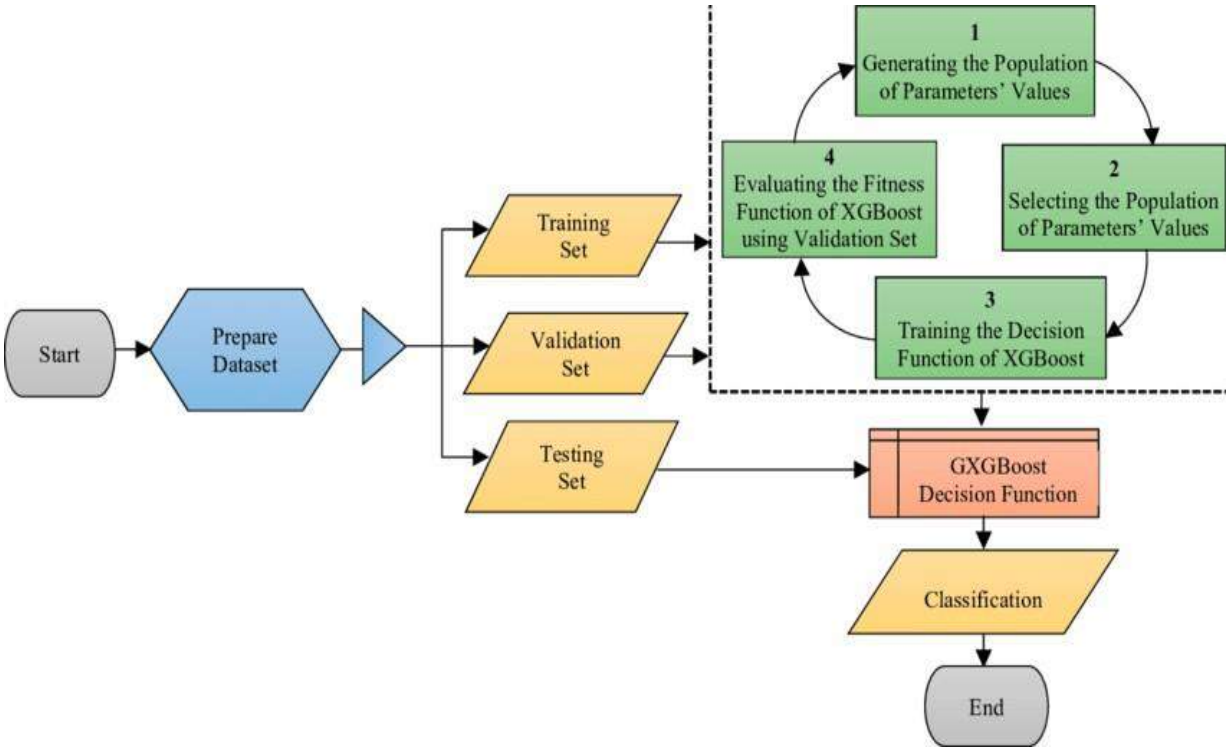


Fig3.2 Flowchart for XGboost Algorithm

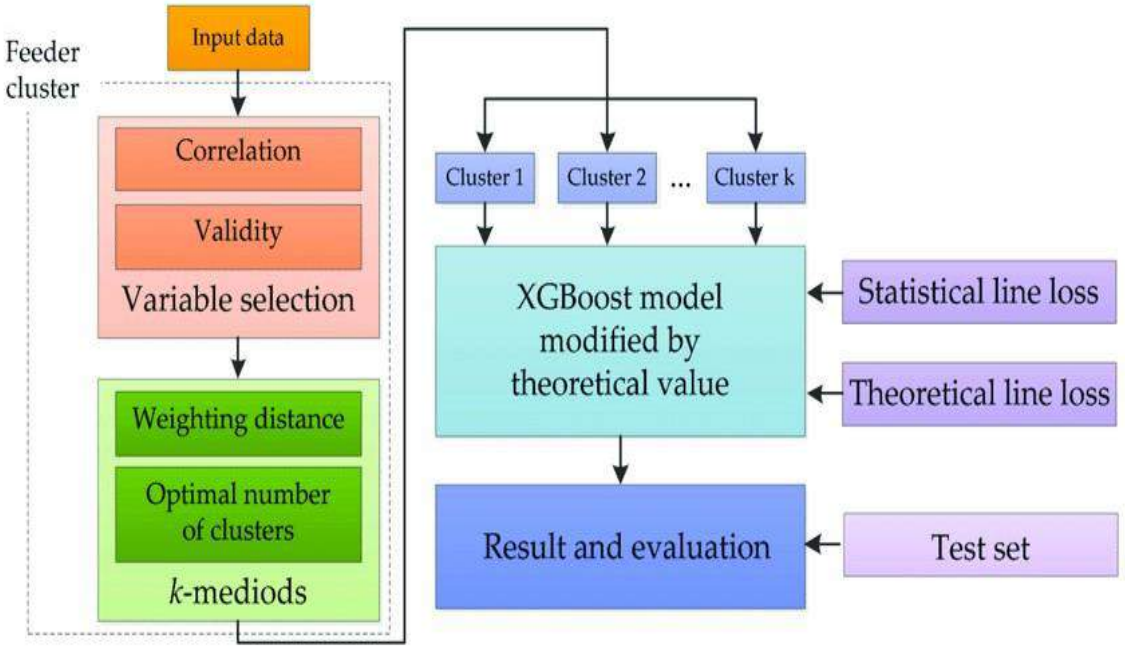


Fig3.3 Flowchart for XGboost model

IMPLEMENTATION DETAILS

4.1 Implementation

Project implementation (or project execution) is **the phase where visions and plans become reality**. This is the logical conclusion, after evaluating, deciding, visioning, planning, applying for funds and finding the financial resources of a project. The following set of snippets help us understand the working of the project in a better way:

```
train = pd.read_excel("C:/Users/RAMANA MURTHY/Downloads/Participants_Data/Data_Train.xlsx")
test = pd.read_excel("C:/Users/RAMANA MURTHY/Downloads/Participants_Data/Data_Test.xlsx")
```

Fig4.1

The above snippet is used to import the training and test data sets.

```
#A method to clean and restructure the Edition column

def split_edition(data):

  edition  = list(data)

  ed_type = [i.split(",- ")[0].strip().upper() for i in edition]

  edit_date = [i.split(",- ")[1].strip() for i in edition]

  m_y = [i.split()[-2:] for i in edit_date]


  for i in range(len(m_y)):
    if len(m_y[i]) == 1:
      m_y[i].insert(0,'NA')

  # Based on the given dataset below is the list of possible values for Months

  months =  ['Apr','Aug','Dec','Feb', 'Jan', 'Jul','Jun','Mar','May','NA','Nov','Oct','Sep']

  ed_month = [m_y[i][0].upper() if m_y[i][0] in months else 'NA' for i in range(len(m_y))]
  ed_year = [int(m_y[i][1].strip()) if m_y[i][1].isdigit() else 0 for i in range(len(m_y))]

  return ed_type, ed_month, ed_year
```

Fig4.2

The above snippet is used to clean and restructure the edition column.

```
#Identifying the maximum number of authors for a single book from the given datasets
authors_1 = list(train['Author'])
authors_2 = list(test['Author'])

authors_1.extend(authors_2)

authorslis = [i.split(",") for i in authors_1]

max = 1
for i in authorslis:
  if len(i) >= max:
    max = len(i)
print("Max. number of authors for a single boook = ",max)

for i in range(len(authorslis)):
  if len(authorslis[i]) == max:
    print(i)

all_authors = [author.strip().upper() for listin in authorslis for author in listin]
```

Fig4.3

The above snippet is used to identify the maximum number of authors for a single book from given data set.

```
#Identifying the maximum number of Genres for a single book from the given datasets

genre_1 = list(train['Genre'])
genre_2 = list(test['Genre'])

genre_1.extend(genre_2)

genre_lis = [i.split(",") for i in genre_1]


max = 1
for i in genre_lis:
  if len(i) >= max:
    max = len(i)
print("Max. number of genres for a single boook = ",max)

all_genres = [genre.strip().upper() for listin in genre_lis for genre in listin]
```

Fig4.4

The above snippet is used to identify the maximum number of genres for a single book from given data set.

```
# A method to split the Genre column in to 7 new columns

def split_genres(data):

  genres = list(data)

  G1 = []
  G2 = []

  for i in genres:

    try :
      G1.append(i.split(',')[0].strip().upper())

    except :
      G1.append('NONE')

    try :
      G2.append(i.split(',')[1].strip().upper())
    except :
      G2.append('NONE')



  return G1,G2

all_genres.append('NONE')
```

Fig4.5

The above snippet is implemented to split the genre column into 7 new columns.

```python
# A method to clean and restructure the datasets

import re

def restructure(data):

  #Cleaning Title Column
  titles = list(data['Title'])
  titles = [title.strip().upper() for title in titles]

  #Cleaning & Restructuring Author Column
  a1,a2,a3,a4,a5,a6,a7 = split_authors(data['Author'])

  #Cleaning & Restructuring Edition Column
  ed_type, ed_month, ed_year = split_edition(data['Edition'])

  #Cleaning Ratings Column
  ratings = list(data['Reviews'])
  ratings = [float(re.sub(" out of 5 stars", "", i).strip()) for i in ratings]

  #Cleaning Reviews Column
  reviews = list(data['Ratings'])
  plu = ' customer reviews'
  reviews = [re.sub(" customer reviews", "", i) if plu in i else re.sub(" customer review", "", i) for i in reviews  ]
  reviews = [int(re.sub(",", "", i).strip()) for i in reviews ]


  #Cleaning & Restructuring Genre Column
  g1, g2 = split_genres(data['Genre'])

  #Cleaning & Restructuring BookCategory Column
  c1,c2 = split_categories(data['BookCategory'])

  # Forming the Structured dataset
  structured_data = pd.DataFrame({'Title': titles,
                                  'Author1': a1,
                                  'Author2': a2,
                                  'Author3': a3,
                                  'Author4': a4,
                                  'Author5': a5,
                                  'Author6': a6,
                                  'Author7': a7,
                                  'Edition_Type': ed_type,
                                  'Edition_Month': ed_month,
                                  'Edition_Year': ed_year,
                                  'Ratings': ratings,
                                  'Reviews': reviews,
                                  'Genre1': g1,
                                  'Genre2': g2,
                                  'Category1': c1,
                                  'Category2': c2

                                 })

  return structured_data
```

Fig4.6

The above snippet is used to clean and restructure the datasets.

```
pip install xgboost
```

Fig4.7

The above snippet is used to install xgboost.

```
!pip install bayesian-optimization
```

Fig4.8

The above snippet is used to install bayesian optimization.

```
#Extracting the best parameters
params = xgb_bo.max['params']

print(params)

#Conversting the max_depth and n_estimator values from float to int
params['max_depth']= int(round(params['max_depth']))
params['n_estimators']= int(round(params['n_estimators']))

print(params)
```

Fig4.9

The above snippet is used to extract the best parameters.

```
#Initialize an XGB with the tuned parameters and fit the training data
from xgboost import XGBRegressor
reg = XGBRegressor(**params).fit(X_train,Y_train)

y_pred_reg = sc.inverse_transform(reg.predict(X_test))
```

Fig 4.9.1

The above snippet to initialize an XGB with the tuned parameters and fit the raining data.

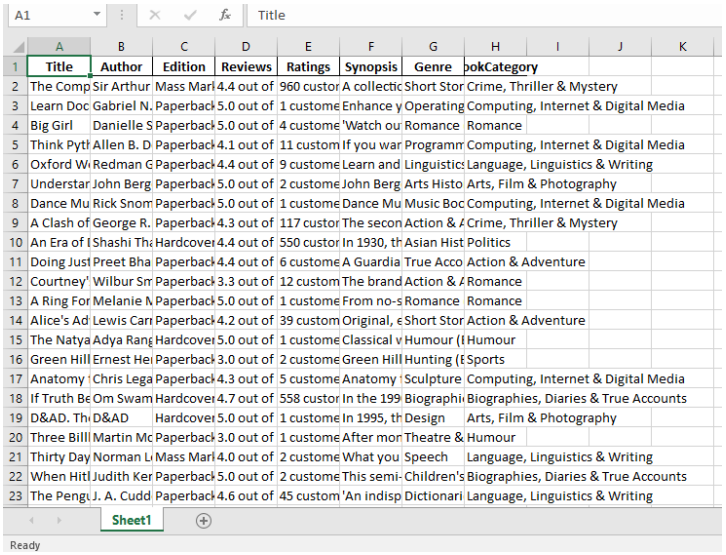
TESTING

5.1 Testing and Training

Testing is the process of evaluating and verifying that a software product or application does what it is supposed to do. In machine learning, a common task is the study and construction of algorithms that can learn from and make predictions on data. Such algorithms function by making data-driven predictions or decisions, through building a mathematical model from input data.

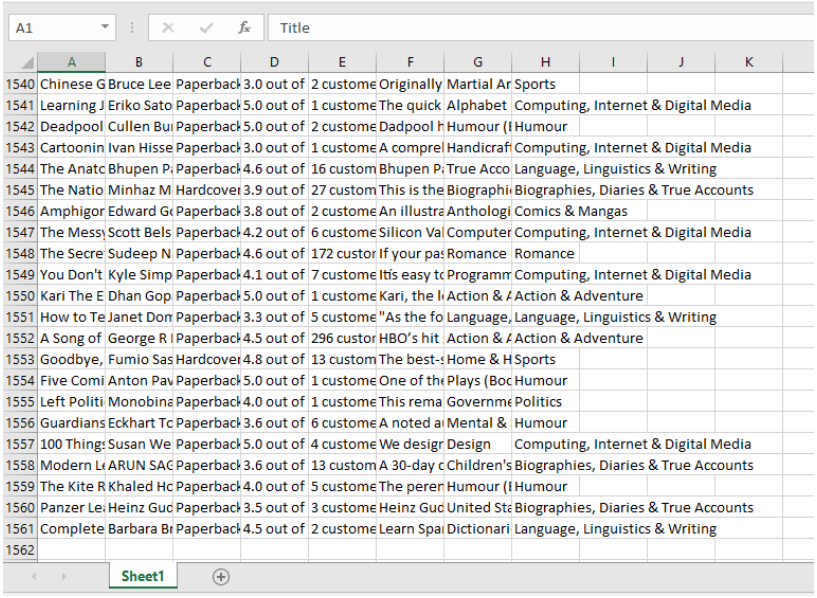
A validation dataset is a sample of data held back from training your model that is used to give an estimate of model skill while tuning model's hyperparameters. The validation dataset is different from the test dataset that is also held back from the training of the model, but is instead used to give an unbiased estimate of the skill of the final tuned model when comparing or selecting between final models. There is much confusion in applied machine learning about what a validation dataset is exactly and how it differs from a test dataset.

A simple evaluation method is a train test dataset where the dataset is divided into a train and a test dataset, then the learning model is trained using the train data and performance is measured using the test data. In a more sophisticated approach, the entire dataset is used to train and test a given model. After the model is built, testing data once again validates that it can make accurate predictions. If training and validation data include labels to monitor performance metrics of the model, the testing data should be unlabeled. Test data provides a final, real-world check of an unseen dataset to confirm that the XGBoost algorithm was trained effectively.



	Title	Author	Edition	Reviews	Ratings	Synopsis	Genre	bookCategory
2	The Comp	Sir Arthur	Mass Mari	4.4 out of	960 custor	A collectic	Short Stor	Crime, Thriller & Mystery
3	Learn Doc	Gabriel N.	Paperback	5.0 out of	1 custome	Enhance y	Operating	Computing, Internet & Digital Media
4	Big Girl	Danielle S	Paperback	5.0 out of	4 custome	'Watch ou	Romance	Romance
5	Think Pyt	Allen B. D	Paperback	4.1 out of	11 custom	If you war	Programm	Computing, Internet & Digital Media
6	Oxford Wi	Redman G	Paperback	4.4 out of	9 custome	Learn and	Linguistics	Language, Linguistics & Writing
7	Understar	John Berg	Paperback	5.0 out of	2 custome	John Berg	Arts Histo	Arts, Film & Photography
8	Dance Mu	Rick Snom	Paperback	5.0 out of	1 custome	Dance Mu	Music Boc	Computing, Internet & Digital Media
9	A Clash of	George R.	Paperback	4.3 out of	117 custor	The secon	Action & A	Crime, Thriller & Mystery
10	An Era of I	Shashi Tha	Hardcover	4.4 out of	550 custor	In 1930, th	Asian Hist	Politics
11	Doing Just	Preet Bha	Paperback	4.4 out of	6 custome	A Guardia	True Acco	Action & Adventure
12	Courtney'	Wilbur Sm	Paperback	3.3 out of	12 custom	The brand	Action & A	Romance
13	A Ring For	Melanie N	Paperback	5.0 out of	1 custome	From no-s	Romance	Romance
14	Alice's Ad	Lewis Carr	Paperback	4.2 out of	39 custom	Original, e	Short Stor	Action & Adventure
15	The Natya	Adya Rang	Hardcover	5.0 out of	1 custome	Classical v	Humour (I	Humour
16	Green Hill	Ernest Hei	Paperback	3.0 out of	2 custome	Green Hill	Hunting (E	Sports
17	Anatomy	Chris Lega	Paperback	4.3 out of	5 custome	Anatomy	Sculpture	Computing, Internet & Digital Media
18	If Truth Be	Om Swam	Hardcover	4.7 out of	558 custor	In the 199	Biographi	Biographies, Diaries & True Accounts
19	D&AD. Th	D&AD	Hardcover	5.0 out of	1 custome	In 1995, th	Design	Arts, Film & Photography
20	Three Bill	Martin Mc	Paperback	3.0 out of	1 custome	After mon	Theatre &	Humour
21	Thirty Day	Norman L	Mass Mari	4.0 out of	2 custome	What you	Speech	Language, Linguistics & Writing
22	When Hitl	Judith Ker	Paperback	5.0 out of	2 custome	This semi-	Children's	Biographies, Diaries & True Accounts
23	The Pengu	J. A. Cudd	Paperback	4.6 out of	45 custom	'An indisp	Dictionari	Language, Linguistics & Writing

Fig5.1 Testing Dataset



	A	B	C	D	E	F	G	H
1540	Chinese G	Bruce Lee	Paperback	3.0 out of	2 custome	Originally	Martial Ar	Sports
1541	Learning J	Eriko Sato	Paperback	5.0 out of	1 custome	The quick	Alphabet	Computing, Internet & Digital Media
1542	Deadpool	Cullen Bu	Paperback	5.0 out of	2 custome	Dadpool h	Humour (I	Humour
1543	Cartoonin	Ivan Hisse	Paperback	3.0 out of	1 custome	A compre	Handicraf	Computing, Internet & Digital Media
1544	The Anato	Bhupen P	Paperback	4.6 out of	16 custom	Bhupen P	True Acco	Language, Linguistics & Writing
1545	The Natio	Minhaz M	Hardcover	3.9 out of	27 custom	This is the	Biographi	Biographies, Diaries & True Accounts
1546	Amphigor	Edward Gc	Paperback	3.8 out of	2 custome	An illustra	Anthologi	Comics & Mangas
1547	The Mess	Scott Bels	Paperback	4.2 out of	6 custome	Silicon Val	Computer	Computing, Internet & Digital Media
1548	The Secre	Sudeep N	Paperback	4.6 out of	172 custor	If your pa	Romance	Romance
1549	You Don't	Kyle Simp	Paperback	4.1 out of	7 custome	Itís easy to	Programm	Computing, Internet & Digital Media
1550	Kari The E	Dhan Gop	Paperback	5.0 out of	1 custome	Kari, the l	Action & A	Action & Adventure
1551	How to Te	Janet Don	Paperback	3.3 out of	5 custome	"As the fo	Language,	Language, Linguistics & Writing
1552	A Song of	George R	Paperback	4.5 out of	296 custor	HBO's hit	Action & A	Action & Adventure
1553	Goodbye,	Fumio Sas	Hardcover	4.8 out of	13 custom	The best-s	Home & H	Sports
1554	Five Comi	Anton Pav	Paperback	5.0 out of	1 custome	One of the	Plays (Boc	Humour
1555	Left Politi	Monobina	Paperback	4.0 out of	1 custome	This rema	Governme	Politics
1556	Guardians	Eckhart Tc	Paperback	3.6 out of	6 custome	A noted a	Mental &	Humour
1557	100 Things	Susan We	Paperback	5.0 out of	4 custome	We design	Design	Computing, Internet & Digital Media
1558	Modern Le	ARUN SAC	Paperback	3.6 out of	13 custom	A 30-day c	Children's	Biographies, Diaries & True Accounts
1559	The Kite R	Khaled Ho	Paperback	4.0 out of	5 custome	The peren	Humour (I	Humour
1560	Panzer Le	Heinz Guc	Paperback	3.5 out of	3 custome	Heinz Guc	United Sta	Biographies, Diaries & True Accounts
1561	Complete	Barbara Bı	Paperback	4.5 out of	2 custome	Learn Spa	Dictionari	Language, Linguistics & Writing
1562								

Fig5.2 Testing Dataset showing maximum number of data input

RESULTS

Project results are **the changes or effects expected to take place after implementing the project**. The results are generally positive improvements to the lives of the beneficiaries. XGBoost is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework. ... The algorithm differentiates itself in the following ways: A wide range of applications: Can be used to solve regression, classification, ranking, and user-defined prediction problems.

The XGBoost library implements **the gradient boosting decision tree algorithm**. This algorithm goes by lots of different names such as gradient boosting, multiple additive regression trees, stochastic gradient boosting or gradient boosting machines.

The dataset will help us determine the prediction of book prices as much as possible. The dataset contains Title, Author, Edition, Review, Synopsis, Genre and Category.

In the following figure we can notice the 10 book prices being predicted as the output for the respective input.

	Price
0	227.955902
1	1925.048828
2	286.947144
3	1020.582458
4	351.452637
5	546.595825
6	610.366028
7	310.343689
8	361.530396
9	444.550079

Fig6.1

The predicted prices of the books.

CONCLUSION AND FUTURE ENHANCEMENT

7.1 Conclusion

XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. This project can be used to predict the price of books with the help of XGboost algorithm in the field of Data Science and in the later stages, its accuracy can be increased eventually. The dataset being used in this project is huge and hence it takes a long time for showing the output as it analyzes a lot of combinations.

All in all the project is very helpful and makes use of Machine Learning with Data Science to give us the required ouputs.

7.2 Future Enhancement

The project has a very vast scope in future. The project **can be implemented on intranet in future**. Project can be updated in near future as and when requirement for the same arises, as it is very flexible in terms of expansion. In particular, we can:

- Further increase the efficiency of the algorithm.
- Improve the accuracy of prediction.
- Try to decrease the execution time.

REFERENCES

- https://machinehack.com/hackathon/predict_the_price_of_books/data
- https://medium.com/analytics-vidhya/books-price-prediction-via-python-31dc358ad8d8
- https://machinelearningmastery.com/xgboost-for-regression
- https://www.geeksforgeeks.org/xgboost-for-regression