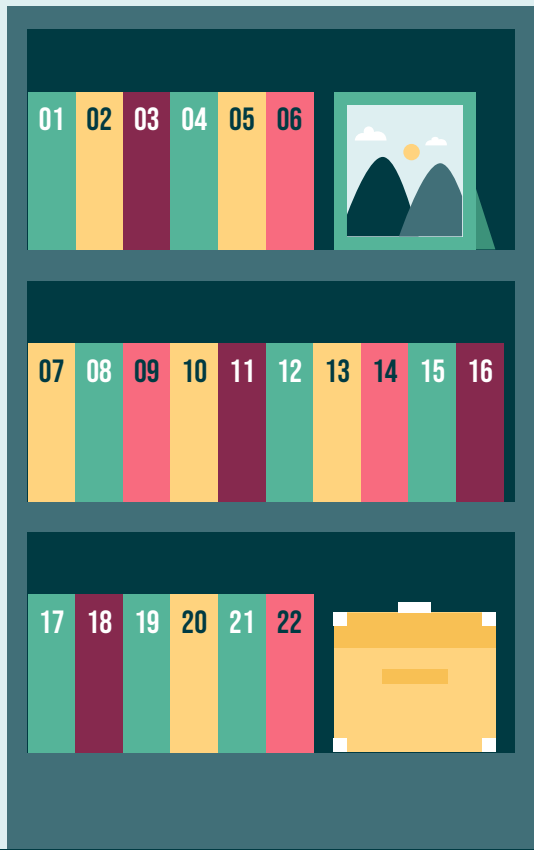


SC1015: GENRE-RATOR

SC12 - A Multilabel Classifier of Genres

- Nathaniel Chin Yi Xuan [REDACTED]
- Marcus Soh Yi Qing [REDACTED]
- Tan Yan Chi [REDACTED]



<https://github.com/natisaver/GoodReads-Multilabel-Genre-Prediction>

TABLE OF CONTENTS

1. Problem Formulation
2. Data Pre-Processing & Cleaning
3. Exploratory Data Analysis
4. Train Test Split / Standardisation
5. Pipeline (Encoder + Classification Algorithms)
6. App Demonstration & Conclusions
7. Future Work

PROBLEM FORMULATION

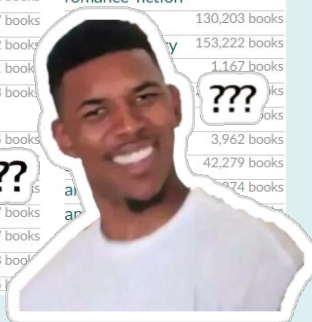
Real-Life Problem:

- Over 900 genres on Goodreads website =>
Unstandardized definition of different genres
- Confusing for user navigation

Project Resolution:

- Aims to produce an algorithm that is able to predict the combination of genres of a book given its **plot description** and **features** to create **standardisation**

10th-century	7,223 books	adult-fiction	1,772,022 books	american-classics	117,722 books
11th-century	8,739 books	adventure	2,345,871 books	american-fiction	201,858 books
12th-century	13,368 books	adventurers	8,242 books	american-history	414,817 books
13th-century	11,838 books	aeroplanes	906 books	american-novels	54,505 books
14th-century	19,292 books	africa	334,114 books	american-revolution	35,761 books
15th-century	22,449 books	african-american	199,808 books	american-revolutionary-war	3,533 books
16th-century	63,414 books	african-american-literature	31,547 books	americana	258,767 books
17th-century	78,865 books	african-american-romance	14,292 books	amish	72,916 books
1864-shenandoah-campaign	37 books	african-literature	38,781 books	amish-fiction	19,365 books
18th-century	138,519 books	agender	307 books	amish-romance-fiction	959 books
1917	2,272 books	agriculture	28,077 books		
19th-century	531,321 books	ahistory	52 books		
1st-grade	86,092 books	aircraft	3,747 books		
20th-century	1,045,119 books	airliners	22 books		
21st-century	701,272 books	airships	2,721 books		
2nd-grade	86,558 books	albanian-literature	2,163 books		
40k	24,297 books	alchemy	35,105 books		
ableism	3,666 books	alcohol	42,279 books		
abuse	477,476 books	alexandria	3,207 books		
academia	214,259 books	algebra	11,107 books		
academic	478,765 books	algeria	1,308 books		
academics	84,704 books	algiers	7,935 books		
accounting	7,880 books	algorithms			
accra	615 books				
action	999,753 books				
activism	101,814 books				
adaptations	121,362 books				
addis-ababa	68 books				



DATASET

The Zenodo Dataset we have chosen contains the following columns.

```
Data columns (total 25 columns):
#  Column      Non-Null Count  Dtype
---  -
0  bookId       52478 non-null    object
1  title        52478 non-null    object
2  series       23470 non-null    object
3  author       52478 non-null    object
4  rating       52478 non-null    float64
5  description   51140 non-null    object
6  language     48672 non-null    object
7  isbn         52478 non-null    object
8  genres       52478 non-null    object
9  characters   52478 non-null    object
10 bookFormat   51005 non-null    object
11 edition     4955 non-null     object
12 pages       50131 non-null    object
13 publisher   48782 non-null    object
14 publishDate 51598 non-null    object
15 firstPublishDate 31152 non-null    object
16 awards      52478 non-null    object
17 numRatings  52478 non-null    int64
18 ratingsByStars 52478 non-null    object
19 likedPercent 51856 non-null    float64
20 setting     52478 non-null    object
21 coverImg    51873 non-null    object
22 bbeScore     52478 non-null    int64
23 bbeVotes    52478 non-null    int64
24 price       38113 non-null    object
```

PROJECT PROCESS OVERVIEW

Preparing Data

- Setting top 30 genres
- Clean Text Data
- Hot Binary Encoding of Categories
- Extract RGB Features & Relative Brightness of Cover Images

Input data

Train Test Split

- Iterative Stratification
- Followed by Vectorisation of textual data via TF-IDF encoder

Train Data

Test Data

Scale

- MinMax or Standard Scaler
- Dependent on classification model used

Vectorise & Scale

All Pipelines

- Encoder + Classification Method + Model

TF-IDF + Binary Relevance + Logistic Regression

TF-IDF + Label Power Set + Naïve Bayes

TF-IDF + Clustered Label Power Set + Linear SVC

Evaluate Accuracy & F1-Score

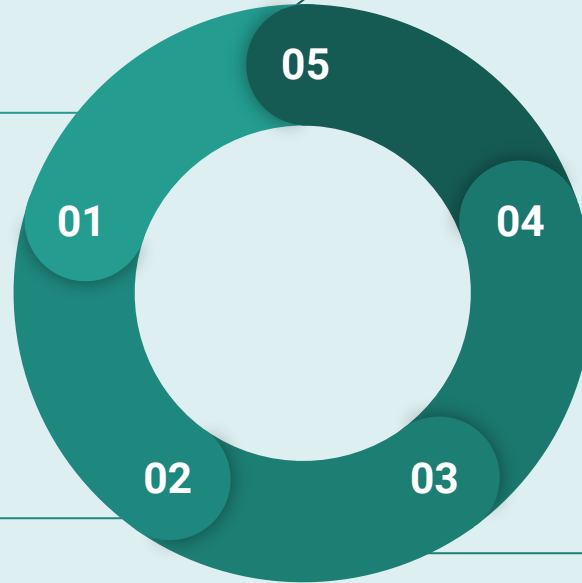
DATA CLEANING

Removing Non-English Books

- To ensure that the description of the books are **purely English content**
- Ensuring **consistency** when analysing textual data.

Dropping columns that are irrelevant

- Columns like 'price', 'publisher' contain too many **NULL** values to be useful for analysis
- Columns like 'language', 'bookFormat' are **irrelevant** to predicting 'genres'



Multi-Hot Binary Encoding of Genre

- Each of the 30 genres were encoded to a **one-hot binary representation**
- If a book belongs to a genre, the value is 1 ("hot") else 0

Cleaning Genre Column

- Reducing the initial **967** "genres" to fixed **top 30 genres**

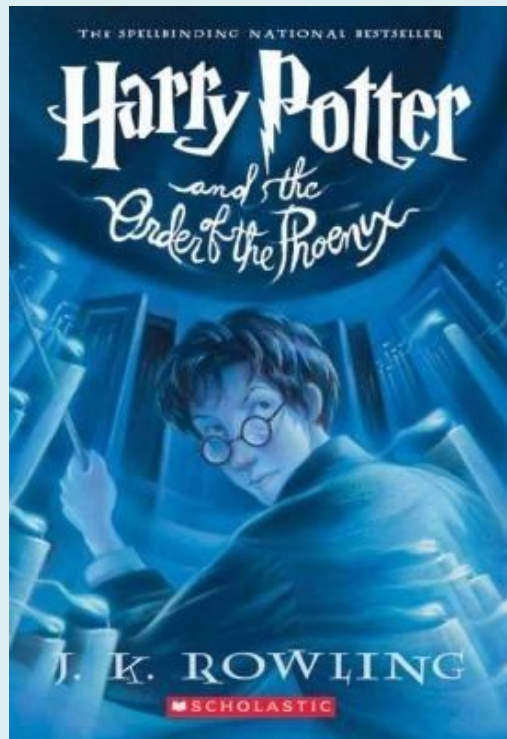
Cleaning Description Column

- Cleaning NULL
- Lemmatizing
- Stop Words
- Accented Characters
- Punctuations
- Lower Case

PROCESSING COVER IMAGE

- To get additional features, we did image processing to extract the **Red, Green, Blue** Values of the cover images for all the books
- Luminance was then calculated using the formula:

$$\bigcirc \text{math.sqrt}(0.241 * (\text{row.r}^{**2}) + 0.691 * (\text{row.g}^{**2}) + 0.068 * (\text{row.b}^{**2}))$$



MODIFIED GENRE LIST

Our team manually analysed the top 60 genres and reduced them to a “top 30 genre list”

- **Purpose**

- Standardisation
- Combine overlapped genres

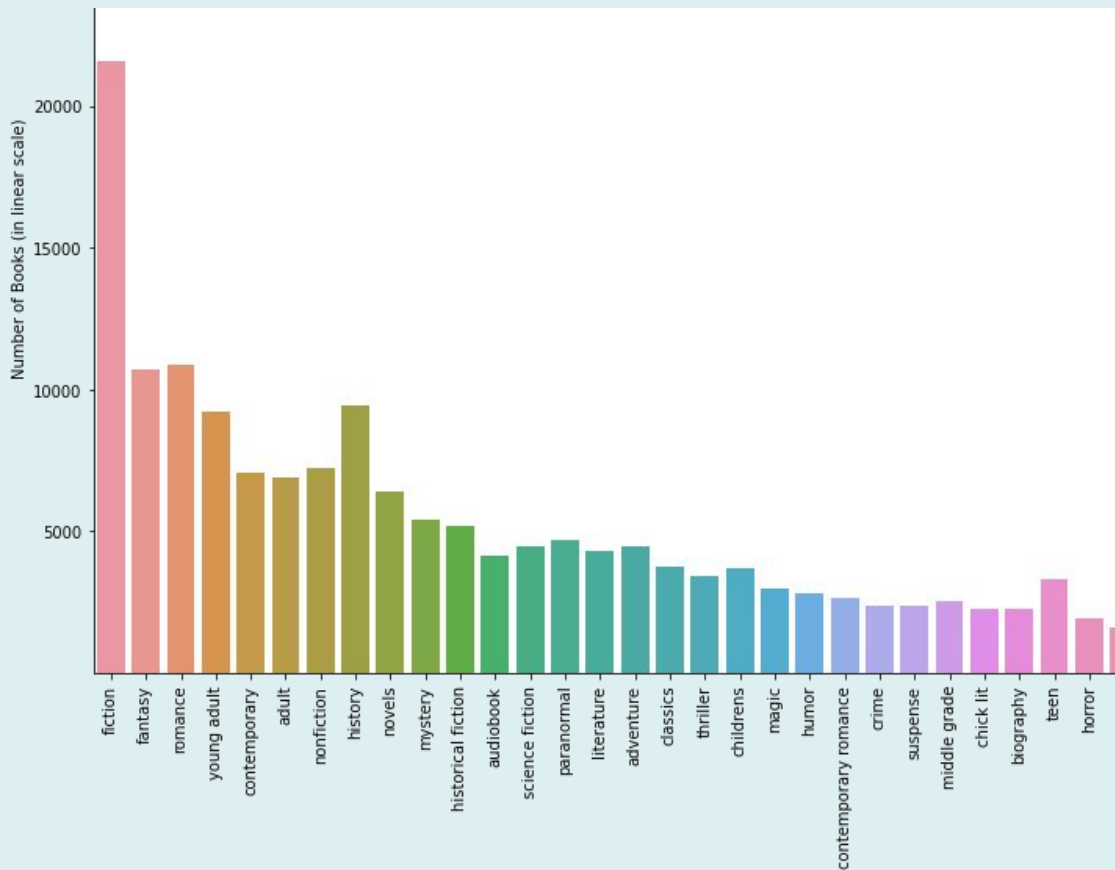
- **Example**

- History, 11th Century, Historical → History
- Adult Fiction → Adult and Fiction

```
top30genrelist = ['fiction',  
'fantasy',  
'romance',  
'young adult',  
'contemporary',  
'adult',  
'nonfiction',  
'history',  
'novels',  
'mystery',  
'historical fiction',  
'audiobook',  
'science fiction',  
'paranormal',  
'literature',  
'adventure',  
'classics',  
'thriller',  
'childrens',  
'magic',  
'humor',  
'contemporary romance',  
'crime',  
'suspense',  
'middle grade',  
'chick lit',  
'biography',  
'teen',  
'horror',  
'philosophy']
```


EXPLORATORY DATA ANALYSIS

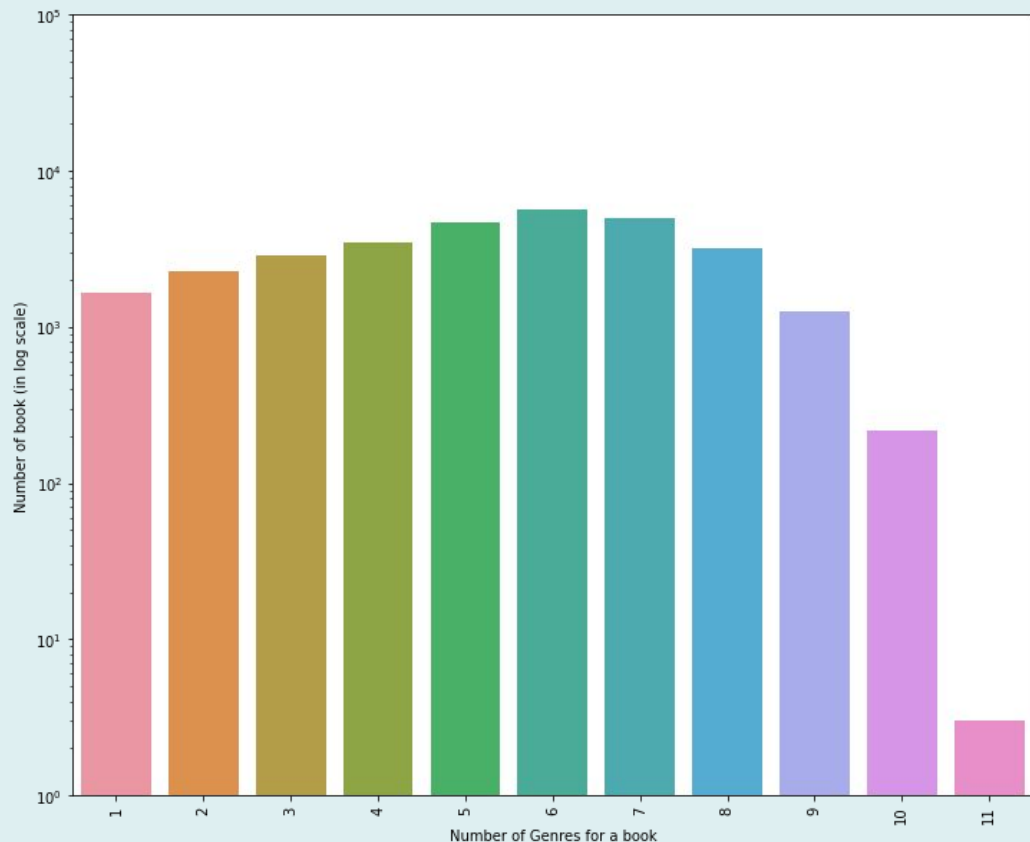
EDA - NUMBER OF BOOKS PER GENRE



Key Observations

- **Fiction** has the highest number of books with **21590**
- Followed by **Romance** which is less than half of fiction at **10862**
- The genre with the least number of books is **philosophy** with **1582** books

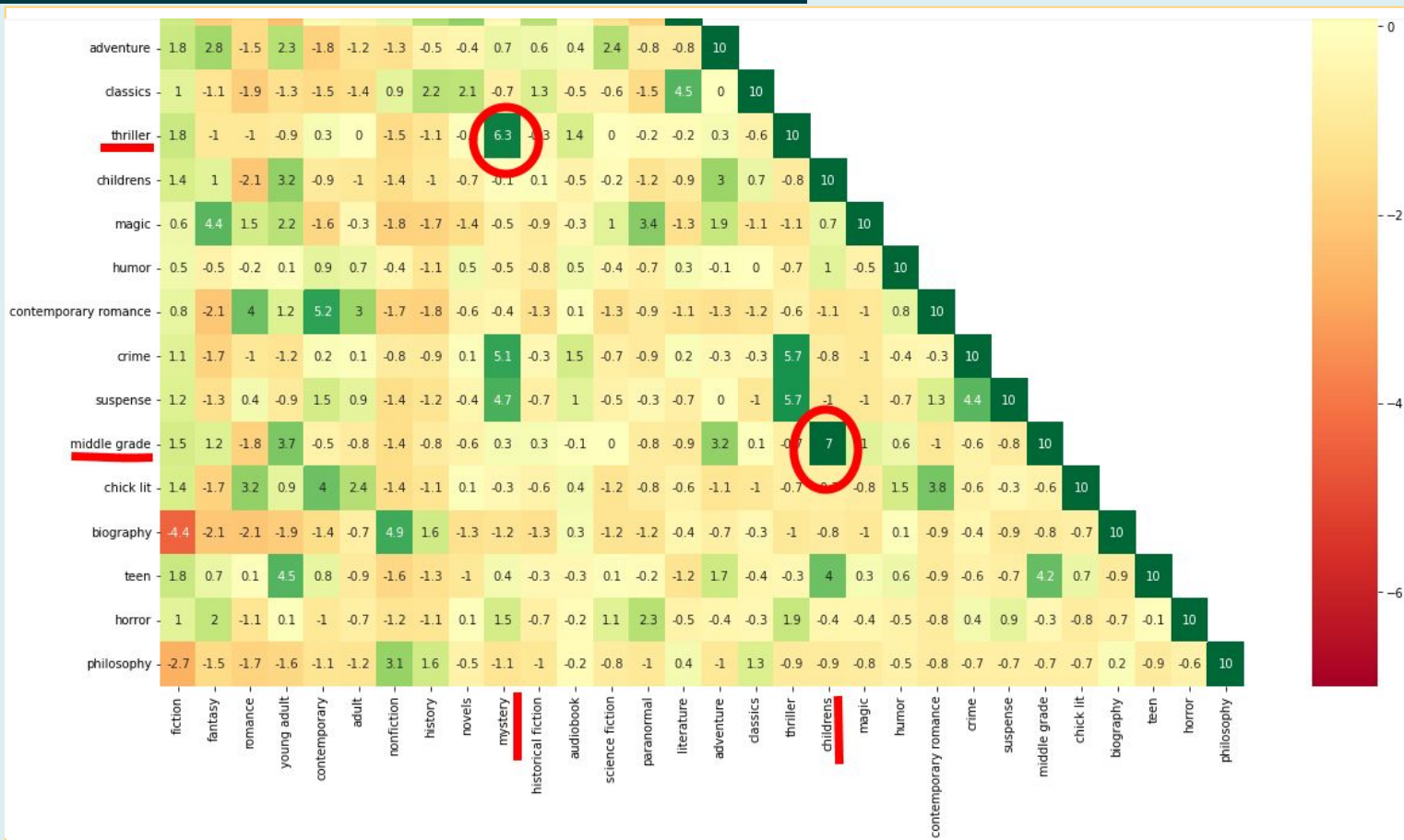
EDA - NUMBER OF GENRE PER BOOK



Key Observations

- On average, books have **5.28** genres.
- There are **3 books** that is associated with **11** total different genres!

EDA - NUMBER OF GENRE PER BOOK

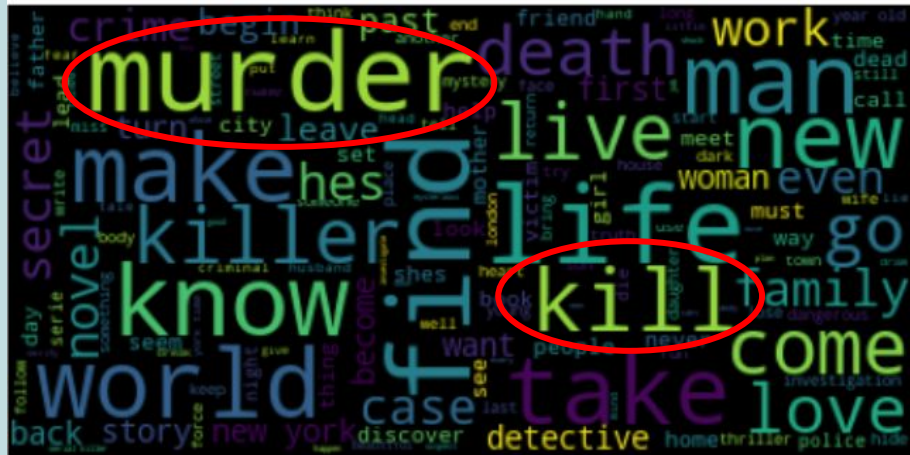


EDA - NUMBER OF GENRE PER BOOK



EXPLORATORY DATA ANALYSIS

Book Genre: crime



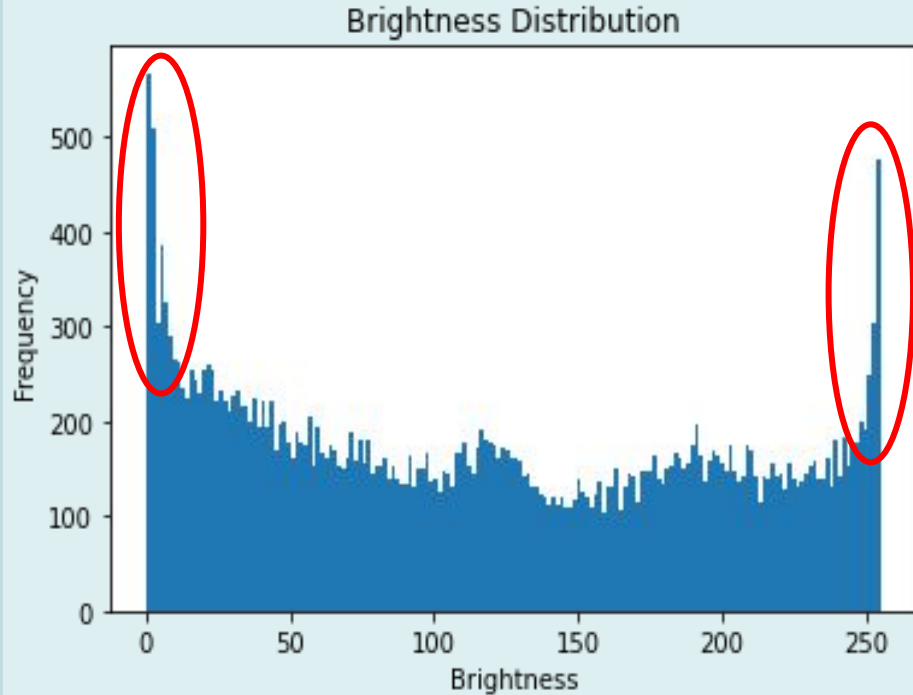
Book Genre: fiction



Key Observations

Certain words such as “**kill**”, “**murder**” appears in Crime, Mystery, Suspense and Thriller “**world**” and “**life**” seems to be the most common words that appear in book descriptions

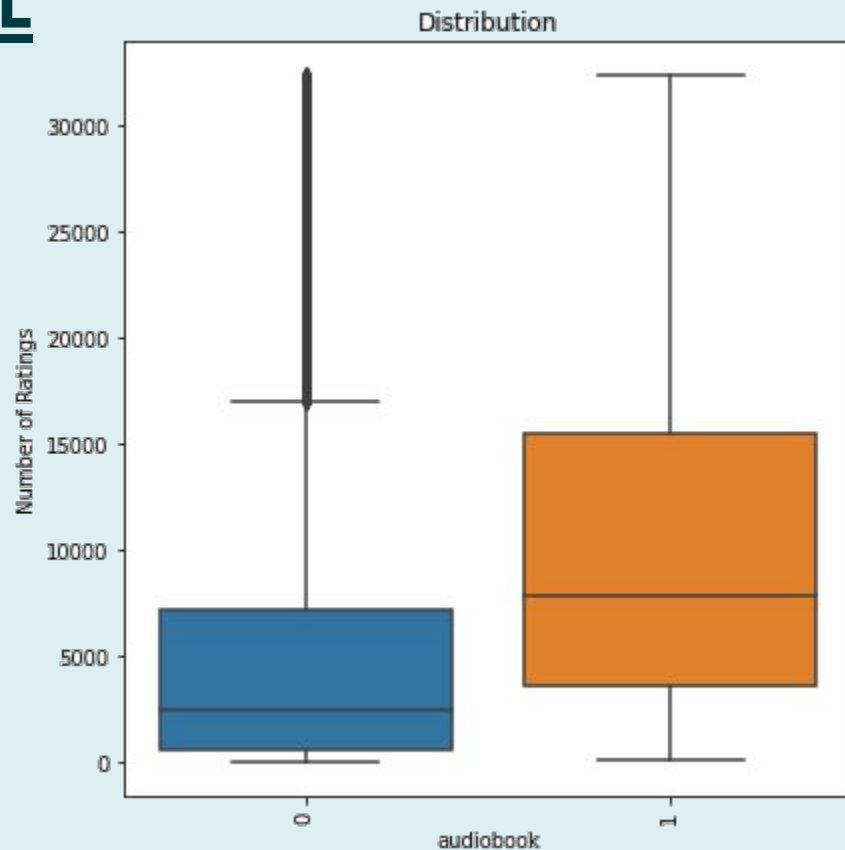
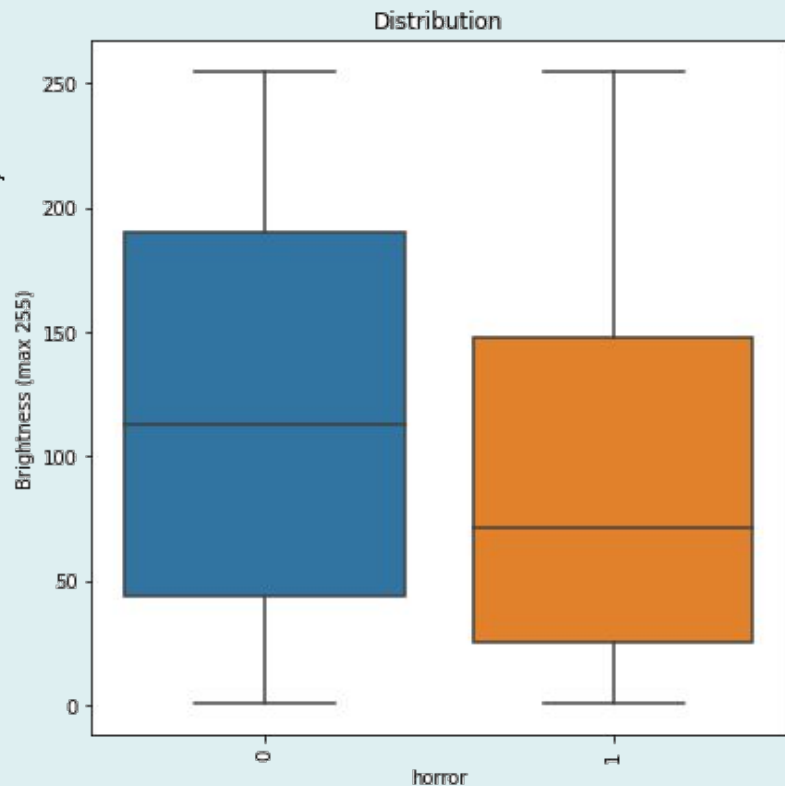
EDA - BRIGHTNESS



Key Observations

- Book covers tend to have **very high brightness** or **very low brightness**

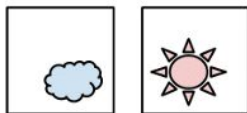
FEATURE RELEVANCE TO GENRE



CLASSIFYING & MACHINE LEARNING

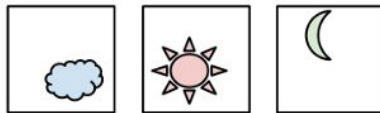
TYPES OF CLASSIFICATION

Binary Classification



- Spam
- Not spam

Multiclass Classification

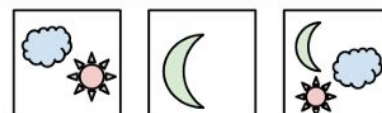


Labels (t)

[0 0 1] [1 0 0] [0 1 0]

- Dog
- Cat
- Horse
- Fish
- Bird

Multi-label Classification



Labels (t)

[1 0 1] [0 1 0] [1 1 1]

- Dog
- Cat
- Horse
- Fish
- Bird

PROJECT PROCESS OVERVIEW

Preparing Data

- Setting top 30 genres
- Clean Text Data
- Hot Binary Encoding of Categories
- Extract RGB Features & Relative Brightness of Cover Images

Input data

Train Test Split

- Iterative Stratification
- Followed by Vectorisation of textual data via TF-IDF encoder

Train Data

Test Data

Scale

- MinMax or Standard Scaler
- Dependent on classification model used

Vectorise & Scale

All Pipelines

- Encoder + Classification Method + Model

TF-IDF + Binary Relevance + Logistic Regression

TF-IDF + Label Power Set + Naïve Bayes

TF-IDF + Clustered Label Power Set + Linear SVC

Evaluate Accuracy & F1-Score



STRATIFICATION

One labelset

Stratification Based on Labelsets

instance	λ_1	λ_2	λ_3
i_1	1	0	1
i_2	0	0	1
i_3	0	1	0
i_4	1	0	0
i_5	0	1	1
i_6	1	1	0
i_7	1	0	1
i_8	1	0	1
i_9	0	0	1

labelset
5
1
2
4
3
6
5
5
1

1 st Fold				
i_1	1	0	1	5
i_2	0	0	1	1
i_3	0	1	0	2

2 nd Fold				
i_7	1	0	1	5
i_9	0	0	1	1
i_4	1	0	0	4

3 rd Fold				
i_8	1	0	1	5
i_5	0	1	1	3
i_6	1	1	0	6

STRATIFICATION

Example

Instance	λ_1	λ_2	λ_3
i_1	1	0	1
i_2	0	0	1
i_3	0	1	0
i_4	1	0	0
i_5	0	1	1
i_6	1	1	0
i_7	1	0	1
i_8	1	0	1
i_9	0	0	1
sum	5	3	6

Firstly
Distribute the
positive examples
of λ_2

1 st Fold			
desired	1.7	1	2
2 nd Fold			
desired	1.7	1	2
3 rd Fold			
desired	1.7	1	2

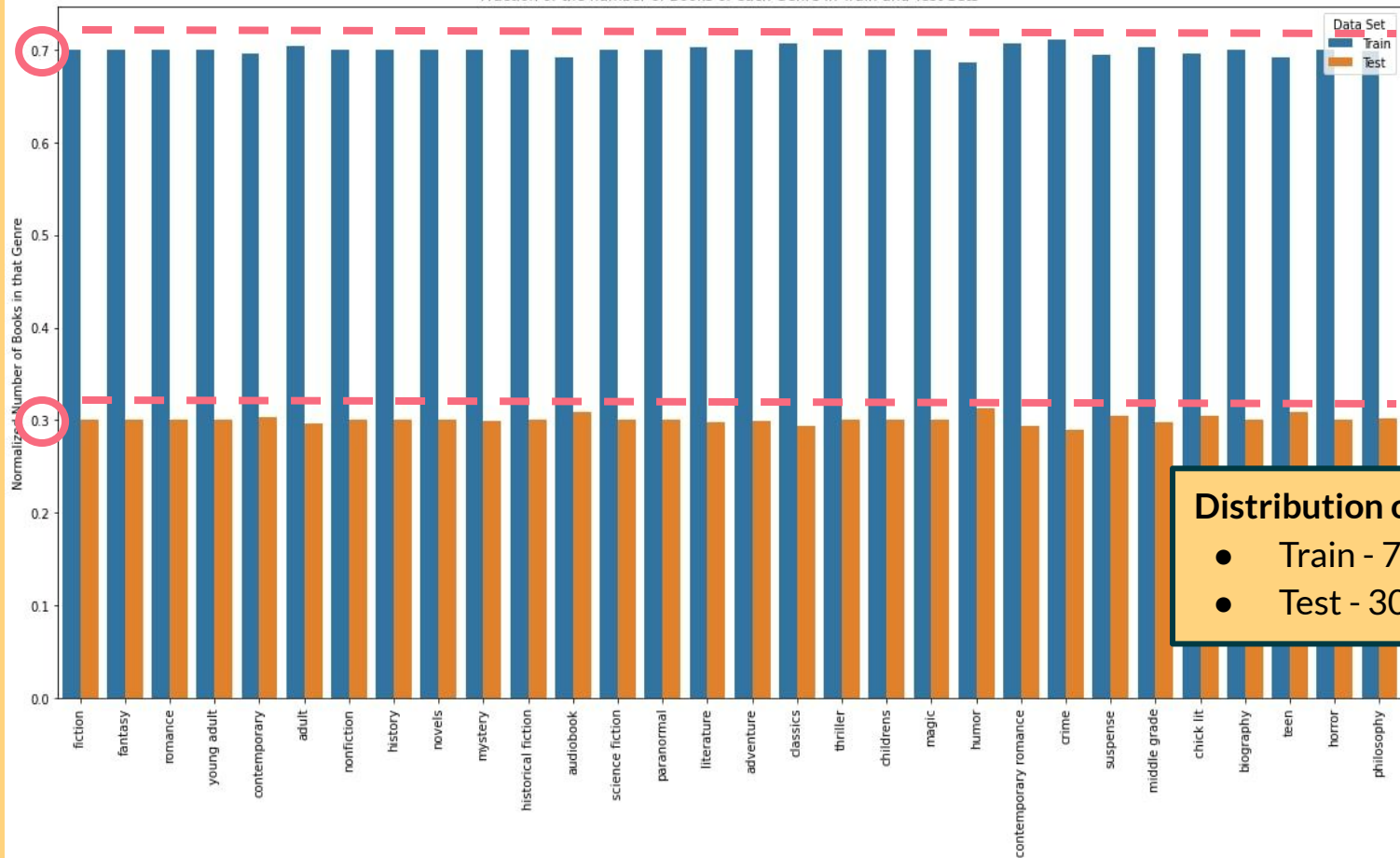
Instance	λ_1	λ_2	λ_3
i_2	0	0	1
i_9	0	0	1
sum	-	-	2

Secondly
Distribute the positive
examples of λ_1

1 st Fold			
i_3	0	1	0
i_1	1	0	1
i_8	1	0	1
desired	-0.3	0	0
2 nd Fold			
i_6	1	1	0
i_7	1	0	1
desired	-0.3	0	1
3 rd Fold			
i_5	0	1	1
i_4	1	0	0
desired	0.7	0	1

λ_2 is distributed first

Fraction of the number of Books of each Genre in Train and Test Sets



Distribution of Train/Test Split

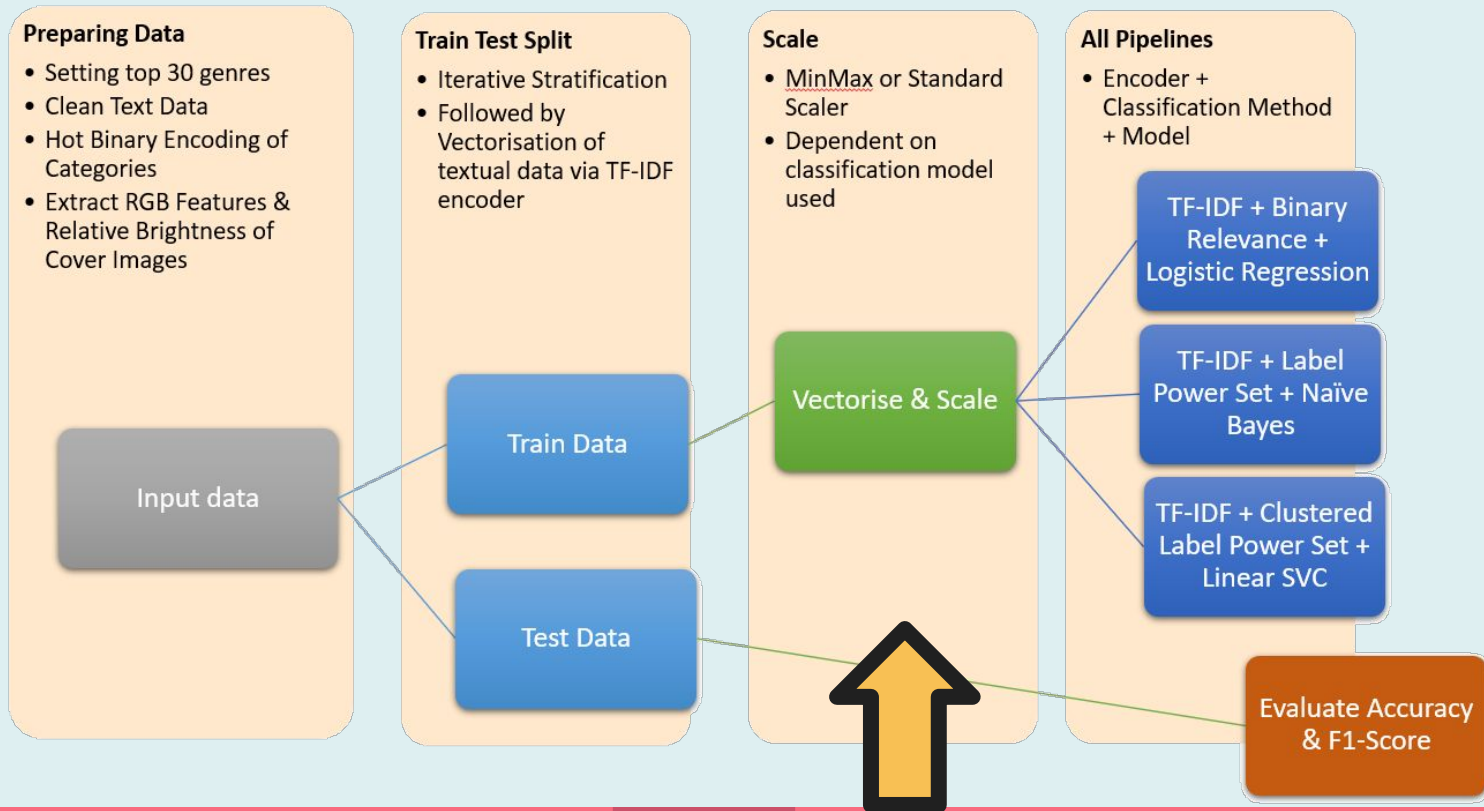
- Train - 70%
- Test - 30%

TRAIN/TEST SPLIT

Y	
Top 30 Genre	Standardised 30 Genres we decided on previously

X	
Brightness	Use of R,G,B values to calculate perceived luminance
Number of Ratings	Number of ratings the book has on GoodReads
Description	The book description given by the author

PROJECT PROCESS OVERVIEW



VECTORIZATION

Vectorization

- Similar concept to one hot encoding
- Convert text to numerical representation

TF-IDF

- Term frequency-inverse document frequency
- Used to quantify the importance or relevance of string representation (words, phrases, etc)

$$IDF_i = \log \left(1 + \frac{N_D}{f_i} \right)$$

Inverse Document Frequency for the search term i within the corpus of documents

The number of documents in the corpus of documents that contain the term D

The number of documents that contain the search term

MACHINE LEARNING

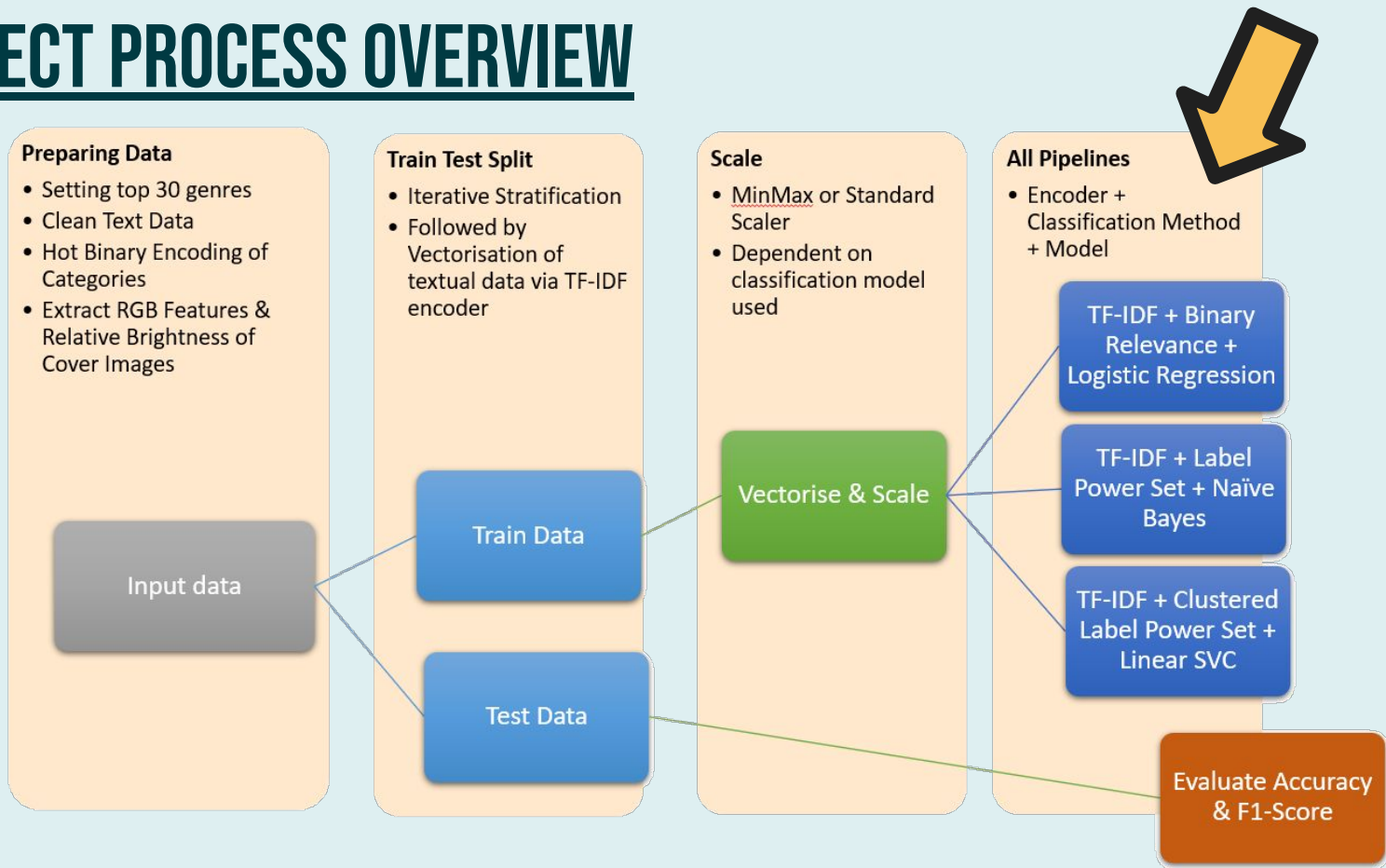
Scaling

- Our group decided to scale our data because ML algorithms are sensitive to data scales
- If the data is not scaled, the features with a higher value range starts dominating when calculating distances
- Our team choose **MinMax Scalar** because it scales from [0 to 1] which ensures that no negative values are returned when passing it into the different models

Machine Learning Algorithms

- Logistic Regression
- Naive Bayes
- Linear Support Vector Machine

PROJECT PROCESS OVERVIEW



MULTI-LABEL CLASSIFICATION ALGORITHMS

Binary Relevance

- Treat each label as a separate class classification
- 30 genres => 30 binary classifiers

X	Y ₁	Y ₂	Y ₃	Y ₄
x ⁽¹⁾	0	1	1	0
x ⁽²⁾	1	0	0	0
x ⁽³⁾	0	1	0	0
x ⁽⁴⁾	1	0	0	1
x ⁽⁵⁾	0	0	0	1



X	Y ₁	X	Y ₂	X	Y ₃	X	Y ₄
x ⁽¹⁾	0	x ⁽¹⁾	1	x ⁽¹⁾	1	x ⁽¹⁾	0
x ⁽²⁾	1	x ⁽²⁾	0	x ⁽²⁾	0	x ⁽²⁾	0
x ⁽³⁾	0	x ⁽³⁾	1	x ⁽³⁾	0	x ⁽³⁾	0
x ⁽⁴⁾	1	x ⁽⁴⁾	0	x ⁽⁴⁾	0	x ⁽⁴⁾	1
x ⁽⁵⁾	0	x ⁽⁵⁾	0	x ⁽⁵⁾	0	x ⁽⁵⁾	1

Label Powerset

- Treat each unique genre combination as a class
- 6436 unique combinations => 6436 classes

X	y1	y2	y3	y4
x1	0	1	1	0
x2	1	0	0	0
x3	0	1	0	0
x4	0	1	1	0
x5	1	1	1	1
x6	0	1	0	0

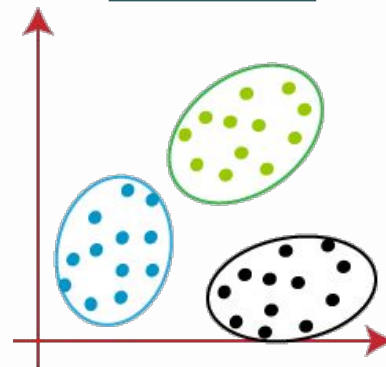


X	y1
x1	1
x2	2
x3	3
x4	1
x5	4
x6	3

Label Powerset with Clustering

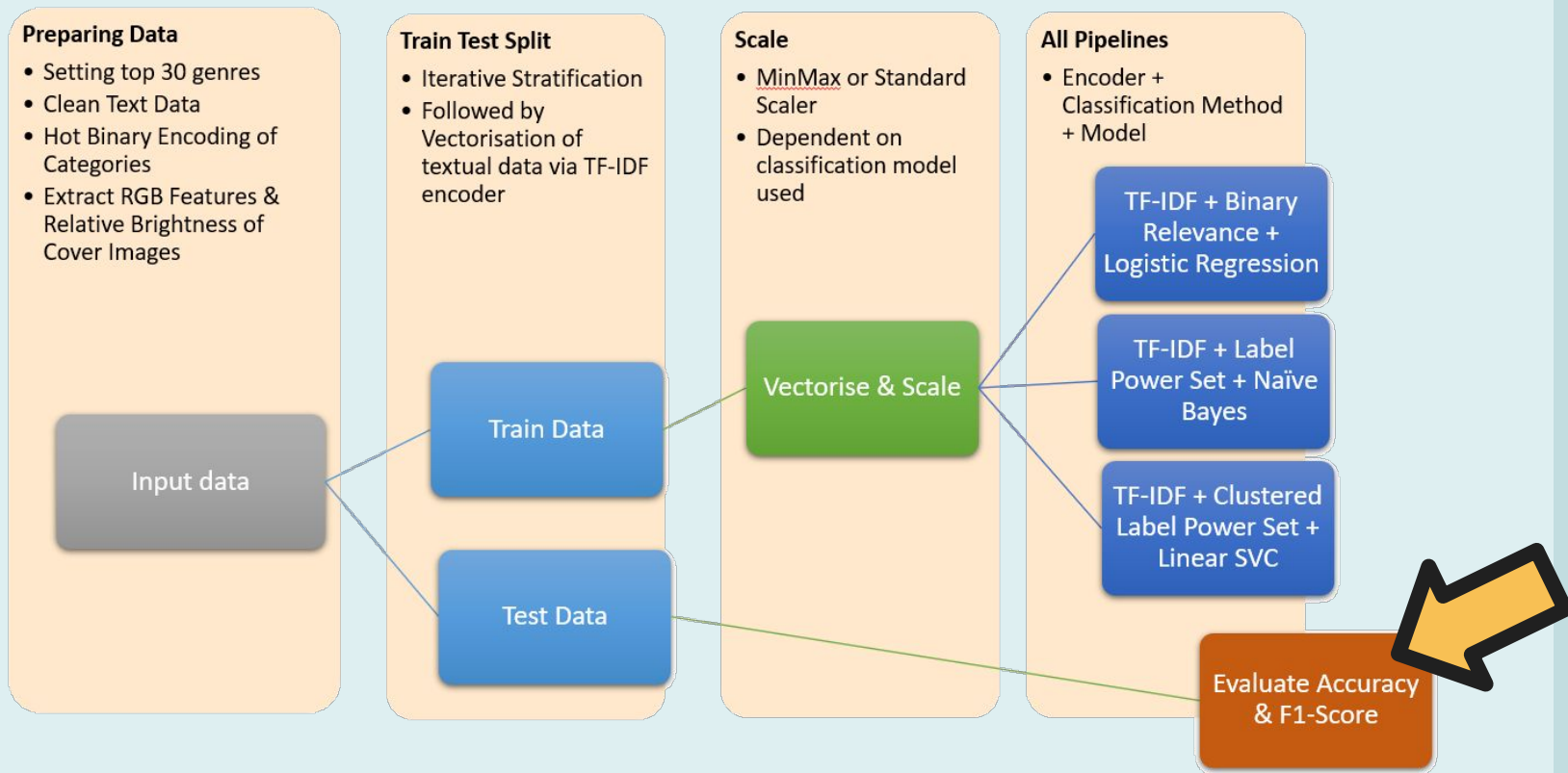
- Reduce the number of genre combinations by clustering from 6436 to 100
- k=100 (number of clusters) gave us the highest average F1-score

After K-Means

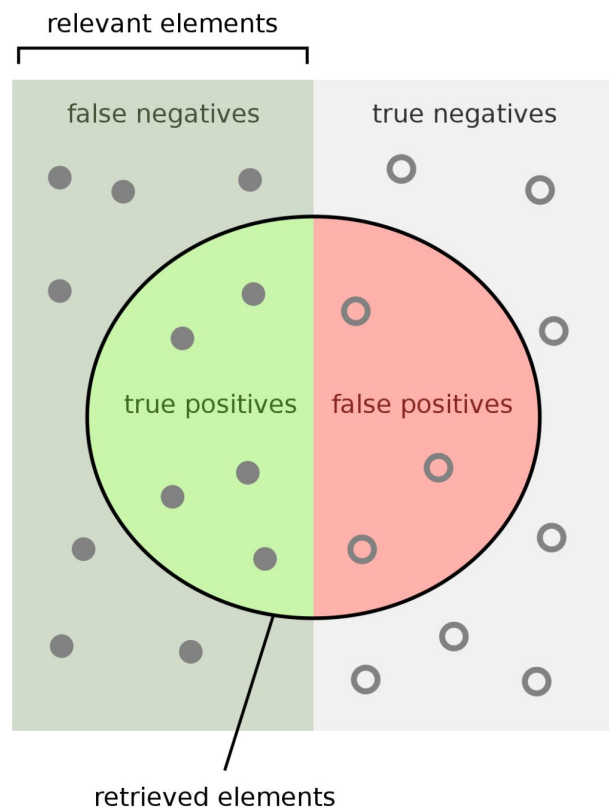


PIPELINE ANALYSIS

PROJECT PROCESS OVERVIEW



EVALUATION OF MODELS



How many retrieved items are relevant?

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

How many relevant items are retrieved?

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$F1\text{-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

EVALUATION OF MODELS

	Precision	Recall	F1-Score	Support
fiction	0.82	0.62	0.71	6477.0
fantasy	0.83	0.42	0.55	3208.0
romance	0.72	0.55	0.62	3298.0
young adult	0.70	0.43	0.53	2774.0
contemporary	0.67	0.27	0.39	2157.0
adult	0.32	0.47	0.38	2039.0
nonfiction	0.47	0.79	0.59	2176.0
biography	0.20	0.59	0.30	681.0
teen	0.56	0.11	0.18	1034.0
horror	0.44	0.08	0.14	575.0
philosophy	0.64	0.31	0.42	475.0
Avg/Total	0.58	0.42	0.43	48132.0

EVALUATION OF MODELS

Machine Learning Model	Precision	Recall	F1-Score
Binary Relevance + Logistic Regression	0.65	0.26	0.29
Label Powerset + Naive Bayes	0.58	0.42	0.43
Label Powerset Clustering + Linear Support Vector Machine	0.29	0.26	0.27



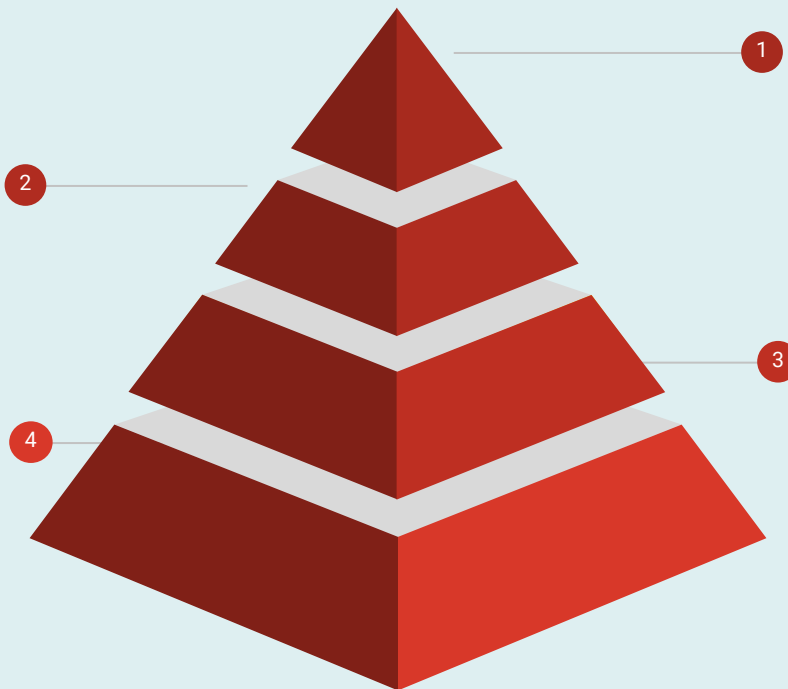
FUTURE WORK

Utilise different Text Encoder

Google Universal
Sentence Encoder

Utilising other Machine Learning Models

- Neural Network
- Cosine Similarity
- Adjusting hyperparameters

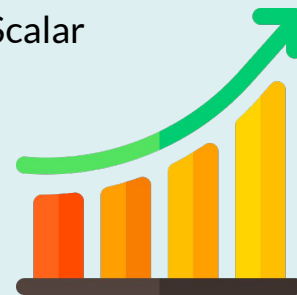


Expand the list of genres

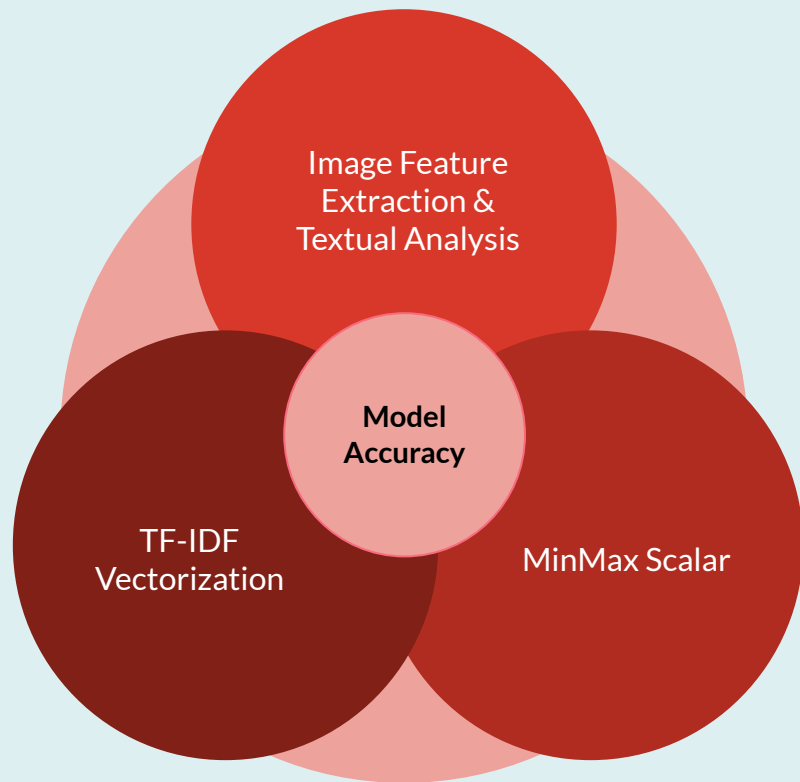
Increase from size 30 to cover a wider range of genres or make it more in depth

Utilising different Scaling techniques

Robust Scalar



NEW TECHNOLOGIES USED



Train Test Split

Iterative Stratification

Binary
Relevance

Logistic Regression

Label
Powerset

Naive Bayes

Label Powerset
with clustering

Linear SVM

THANK YOU!



<https://github.com/natisaver/GoodReads-Multilabel-Genre-Prediction>

References

- <https://scikit-learn.org/stable/modules/multiclass.html>
- <https://realpython.com/image-processing-with-the-python-pillow-library/>

