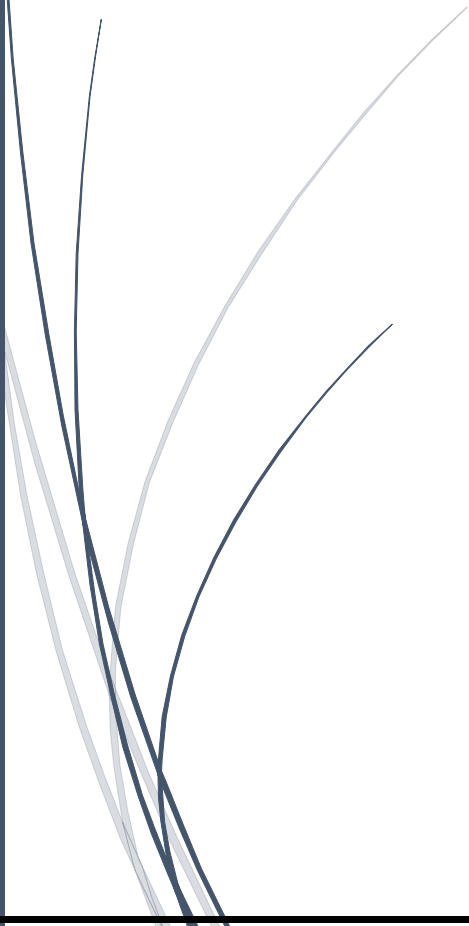


CSE4022 NATURAL LANGUAGE PROCESSING

J-COMPONENT
FINAL PROJECT REPORT

BOOK GENRE PREDICTION





VIT[®]

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

CSE4022 – NATURAL LANGUAGE PROCESSING

DR. G. BHARADWAJA KUMAR

J-COMPONENT

FINAL PROJECT REPORT

BOOK GENRE PREDICTION

Team members:

- 1. Abhinav Vijayakumar – 19BCE1311**
- 2. Amar Dixit – 19BCE1875**

ACKNOWLEDGMENT

We wish to express our sincere thanks and deep sense of gratitude to our project guide, *Dr. G. Bharadwaja Kumar*, Professor, SCSE, for his consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work.

We are extremely grateful to the Dean of the SCOPE, VIT Chennai, for extending the facilities of the School towards our project and for the unstinting support. We also take this opportunity to thank all the faculty of the School for their support and their wisdom imparted to us throughout the course.

TABLE OF CONTENTS

	Page no.
1. Abstract.....	4
2. Introduction.....	5
3. Keywords.....	8
4. Datasets.....	12
5. Methodology.....	14
6. Implementation.....	15
6.1 Data Distribution.....	15
6.2 Data Abstraction and Cleaning.....	16
6.3 Data Augmentation.....	17
6.4 Removing Stop Words and Finding Frequent Words.....	19
6.5 Modifying Dataset For Multi-Label.....	21
6.6 Predicting the Genres.....	22
6.7 Printing the Predicted Genre Classes.....	23
6.8 Predicted outcomes of some books based on our model.....	24
7. Conclusion.....	25
8. References.....	25

ABSTRACT

Writing genres (more commonly known as literary genres) are categories that distinguish literature (including works of prose, poetry, drama, hybrid forms, etc.) based on some set of stylistic criteria. Sharing literary conventions, they typically consist of similarities in theme/topic, style, tropes, and storytelling devices; common settings and character types; and/or formulaic patterns of character interactions and events, and an overall predictable form.

A literary genre may fall under either one of two categories: (a) a work of fiction, involving non-factual descriptions and events invented by the author; or (b) a work of nonfiction, in which descriptions and events are understood to be factual. In literature, a work of fiction can refer to a short story, novella, and novel, the latter being the longest form of literary prose. Every work of fiction falls into a literary subgenre, each with its own style, tone, and storytelling devices.

Moreover, these genres are formed by shared literary conventions that change over time as new genres emerge while others fade. Accordingly, they are often defined by the cultural expectations and needs of a particular historical and, cultural moment or place.

According to Alastair Fowler, the following elements can be used to define genres: organizational features (chapters, acts, scenes, stanzas); length; mood; style; the reader's role (e.g., in mystery works, readers are expected to interpret evidence); and the author's reason for writing.

This project examines automated genre classification in literature. The approach described uses text based comparison of book summaries as the primary feature to predict the genres of the book.

Genre means a type of art, literature, or music characterized by a specific form, content, and style. For example, literature has four main genres: poetry, drama, fiction, and non-fiction. All of these genres have particular features and functions that distinguish them from one another. Hence, it is necessary on the part of readers to know which category of genre they are reading in order to understand the message it conveys, as they may have certain expectations prior to the reading concerned.

This makes automatically generating genre labels a potentially useful tool in sorting unmarked text collections or searching the web. Our project deals with this problem by applying different text classification techniques and models to find the best solution for the same.

INTRODUCTION

Books, important cultural products, play a big role in our daily lives—they both educate and entertain. And it is big business: the publishing industry revenue is projected to be more than 43 billion dollars. The success of a book depends on the genre, author, title etc. So the prediction of the genre of a book is really important for both the publisher and the readers.

Some websites allow writers and publishers to publish their books in a simple way. Each person has a preference for literary genres when choosing their next reading, so choosing genres well when publishing a book can make your book reach the right audience, thereby increasing your sales or advertising. The focus of this project will be to create a method for recommending literary genre tags for writers and publishers to publish their books. Thus, when filling in filling out the book's information on a platform, it indicates some genre tags that best fit the description of the work.

Fiction refers to a story that comes from a writer's imagination, as opposed to one based strictly on fact or a true story. In the literary world, a work of fiction can refer to a short story, novella, and novel, which is the longest form of literary prose. Every work of fiction falls into a sub-genre, each with its own style, tone, elements, and storytelling devices.



The 14 Main Literary Genres

1. Literary Fiction. Literary fiction novels are considered works with artistic value and literary merit. They often include political criticism, social commentary, and reflections on humanity. Literary fiction novels are typically character-driven, as opposed to being plot-driven, and follow a character's inner story. Learn more about writing fiction in James Patterson's MasterClass.

2. Mystery. Mystery novels, also called detective fiction, follow a detective solving a case from start to finish. They drop clues and slowly reveal information, turning the reader into a detective trying to solve the case, too. Mystery novels start with an exciting hook, keep readers interested with suspenseful pacing, and end with a satisfying conclusion that answers all of the reader's outstanding questions.

3. Thriller. Thriller novels are dark, mysterious, and suspenseful plot-driven stories. They very seldom include comedic elements, but what they lack in humor, they make up for in suspense. Thrillers keep readers on their toes and use plot twists, red herrings, and cliffhangers to keep them guessing until the end. Learn how to write your own thriller in Dan Brown's MasterClass.

4. Horror. Horror novels are meant to scare, startle, shock, and even repulse readers. Generally focusing on themes of death, demons, evil spirits, and the afterlife, they prey on fears with scary beings like ghosts, vampires, werewolves, witches, and monsters. In horror fiction, plot and characters are tools used to elicit a terrifying sense of dread. R.L. Stine's MasterClass teaches tips and tricks for horror writing.

5. Historical. Historical fiction novels take place in the past. Written with a careful balance of research and creativity, they transport readers to another time and place—which can be real, imagined, or a combination of both. Many historical novels tell stories that involve actual historical figures or historical events within historical settings.

6. Romance. Romantic fiction centers around love stories between two people. They're lighthearted, optimistic, and have an emotionally satisfying ending. Romance novels do contain conflict, but it doesn't overshadow the romantic relationship, which always prevails in the end.

7. Western. Western novels tell the stories of cowboys, settlers, and outlaws exploring the western frontier and taming the American Old West. They're shaped specifically by their genre-specific elements and rely on them in ways that novels in other fiction genres don't. Westerns aren't as popular as they once were; the golden age of the genre coincided with the popularity of western films in the 1940s, '50s, and '60s.

8. Bildungsroman. Bildungsroman is a literary genre of stories about a character growing psychologically and morally from their youth into adulthood. Generally, they experience a profound emotional loss, set out on a journey, encounter conflict, and grow into a mature

person by the end of the story. Literally translated, a bildungsroman is “a novel of education” or “a novel of formation.”

9. Speculative Fiction. Speculative fiction is a supergenre that encompasses a number of different types of fiction, from science fiction to fantasy to dystopian. The stories take place in a world different from our own. Speculative fiction knows no boundaries; there are no limits to what exists beyond the real world. Learn more about speculative fiction in Margaret Atwood’s MasterClass.

10. Science Fiction. Sci-fi novels are speculative stories with imagined elements that don’t exist in the real world. Some are inspired by “hard” natural sciences like physics, chemistry, and astronomy; others are inspired by “soft” social sciences like psychology, anthropology, and sociology. Common elements of sci-fi novels include time travel, space exploration, and futuristic societies.

11. Fantasy: Fantasy novels are speculative fiction stories with imaginary characters set in imaginary universes. They’re inspired by mythology and folklore and often include elements of magic. The genre attracts both children and adults; well-known titles include *Alice’s Adventures in Wonderland* by Lewis Carroll and the *Harry Potter* series by J.K. Rowling. Learn more about character and worldbuilding in Neil Gaiman’s MasterClass.

12. Dystopian: Dystopian novels are a genre of science fiction. They’re set in societies viewed as worse than the one in which we live. Dystopian fiction exists in contrast to utopian fiction, which is set in societies viewed as better than the one in which we live. Margaret Atwood’s MasterClass teaches elements of dystopian fiction.

13. Magical Realism: Magical realism novels depict the world truthfully, plus add magical elements. The fantastical elements aren’t viewed as odd or unique; they’re considered normal in the world in which the story takes place. The genre was born out of the realist art movement and is closely associated with Latin American authors.

14. Realist Literature: Realist fiction novels are set in a time and place that could actually happen in the real world. They depict real people, places, and stories in order to be as truthful as possible. Realist works of fiction remain true to everyday life and abide by the laws of nature as we currently understand them.

KEYWORDS

- **NLTK:**

A suite of libraries and programs for symbolic and statistical natural language processing for English.

- **SEABORN**

Seaborn is an open-source Python library built on top of matplotlib. It is used for data visualization and exploratory data analysis. Seaborn works easily with dataframes and the Pandas library. The graphs created can also be customized easily.

Seaborn and Matplotlib are two of Python's most powerful visualization libraries. Seaborn uses fewer syntax and has stunning default themes and Matplotlib is more easily customizable through accessing the classes.

- **Wordnet**

WordNet is a large lexical database corpus in NLTK. WordNet maintains cognitive synonyms (commonly called synsets) of words correlated by nouns, verbs, adjectives, adverbs, synonyms, antonyms, and more. WordNet is a very useful tool for text analysis.

- **Tqdm**

tqdm is a library in Python which is used for creating Progress Meters or Progress Bars. tqdm got its name from the Arabic name taqaddum which means 'progress'.

tqdm is a Python library that allows you to output a smart progress bar by wrapping around any iterable. A tqdm progress bar not only shows you how much time has elapsed, but also shows the estimated time remaining for the iterable.

- **Sklearn**

Convert a collection of raw documents to a matrix of TF-IDF features.

- **TF-IDF transformer**

Transform a count matrix to a normalized TF or TF-IDF representation.

TF means term-frequency while TF-IDF means term-frequency times inverse document-frequency. This is a common term weighting scheme in information retrieval, that has also found good use in document classification.

The goal of using TF-IDF instead of the raw frequencies of occurrence of a token in a given document is to scale down the impact of tokens that occur very frequently in a given corpus and that are hence empirically less informative than features that occur in a small fraction of the training corpus.

- **Count Vectorizer**

Convert a collection of text documents to a matrix of token counts.

- **Xml.etree**

The `xml.etree.ElementTree` module implements a simple and efficient API for parsing and creating XML data.

- **Pickle**

The pickle module implements binary protocols for serializing and de-serializing a Python object structure.

“*Pickling*” is the process whereby a Python object hierarchy is converted into a byte stream, and “*unpickling*” is the inverse operation, whereby a byte stream (from a binary file or bytes-like object) is converted back into an object hierarchy.

Pickling (and unpickling) is alternatively known as “serialization”, “marshalling,” or “flattening”; however, to avoid confusion, the terms used here are “pickling” and “unpickling”.

- **matplotlib inline**

This function sets up the matplotlib to work interactively. It lets you activate the matplotlib interactive support anywhere in an IPython session.

The magic function `%matplotlib inline` to enable the inline plotting, where the plots/graphs will be displayed just below the cell where your plotting commands are written.

It provides interactivity with the backend in the frontends like the jupyter notebook. It also provides the feature where, the plotting commands below the output cell of the previous plot,

will not affect the previous plot, which means it separates different plots. For example, changing the color palette by colormap in the cell below the previous plot output cell will not change the colormap of that plot.

- **Sklearn MultiLabelBinarizer**

This transformer converts between this intuitive format and the supported multilabel format: a (samples x classes) binary matrix indicating the presence of a class label.

Combining Many genres into single Multi Label Matrix

- **Train_test_split()**

Split arrays or matrices into random train and test subsets

- **MODEL**

Logistic Regression

In statistics, the logistic model (or logit model) is used to model the probability of a certain class or event existing such as pass/fail, win/lose, alive/dead or healthy/sick. This can be extended to model several classes of events such as determining whether an image contains a cat, dog, lion, etc. Each object being detected in the image would be assigned a probability between 0 and 1, with a sum of one.

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist.

- **ALGORITHM**

One VS Rest scheme

Also known as one-vs-all, this strategy consists in fitting one classifier per class. For each classifier, the class is fitted against all the other classes.

In the multiclass case, the training algorithm uses the one-vs-rest (OvR) scheme if the 'multi_class' option is set to 'ovr', and uses the cross-entropy loss if the 'multi_class' option is set to 'multinomial'.

- **F1 score**

Also known as balanced F-score or F-measure.

The F1 score can be interpreted as a **harmonic mean** of the precision and recall, where an F1 score reaches its best value at 1 and worst score at 0. The relative contribution of precision and recall to the F1 score are equal.

$$F1 = 2 * (\text{precision} * \text{recall}) / (\text{precision} + \text{recall})$$

- **Pickle**

The pickle module implements binary protocols for serializing and de-serializing a Python object structure.

DATASETS

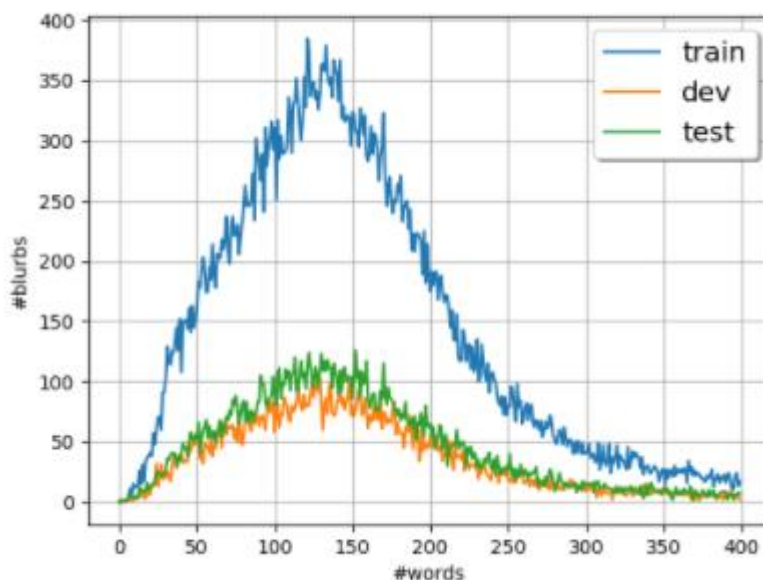
We have used BlurbGenreCollection-EN dataset with information of published books such as author, title, ISBN, publish date, book summary description and so on.

The BlurbGenreCollection-EN is a dataset consisting of advertising descriptions of books - so called blurbs - for the English language. Each blurb is categorized into one or multiple categories. The categories are structured hierarchically.

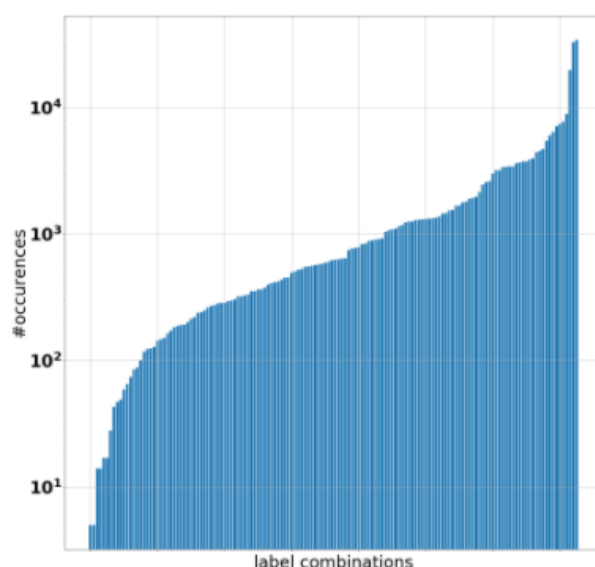
Dataset Link: <https://www.inf.uni-hamburg.de/en/inst/ab/lt/resources/data/blurb-genre-collection.html>

Quantitative characteristics

Number of samples	91,892
Average length of blurb	157.51
Total number of classes	146
Average number of genres per book	3.01
Classes on each level of the hierarchy	L1: 7; L2: 46; L3: 77; L4: 7



The hierarchy of the dataset consists of four levels and is organized as a forest. It is important to note that the most specific genre of a book does not have to be a leaf. For instance, the most specific class of a book could be Children's Books, although Children's Book has further children genres, such as Middle Grade books. However, a great number of books are simply not classified into more special classes on the website. Analysing the occurrence of each genre combination on a log scale shows that the English dataset has a distribution in which some labels have disproportionately few or many examples.



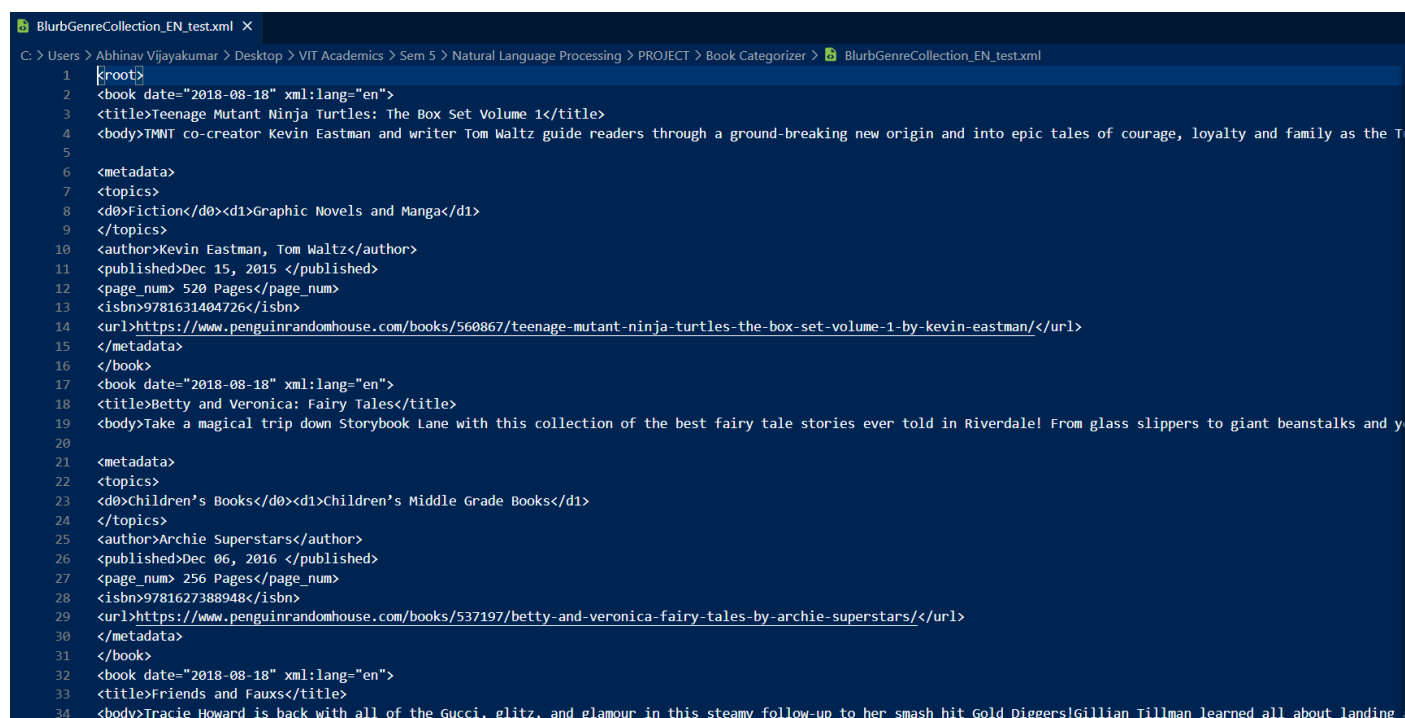
BlurbGenreCollection_EN_train.xml

```

BlurbGenreCollection_EN_train.xml X
C: > Users > Abhinav Vijayakumar > Desktop > VIT Academics > Sem 5 > Natural Language Processing > PROJECT > Book Categorizer > BlurbGenreCollection_EN_train.xml
1  <root>
2  <book date="2018-08-18" xml:lang="en">
3  <title>The New York Times Daily Crossword Puzzles: Thursday, Volume 1</title>
4  <body>Monday's Crosswords Do with EaseTuesday's Crosswords Not a BreezeWednesday's Crosswords Harder StillThursday's Crosswords Take Real SkillFriday's Crosswords – Yo
5  <metadata>
6  <topics>
7  <d0>Nonfiction</d0><d1>Games</d1>
8  </topics>
9  <author>New York Times</author>
10 <published>Dec 28, 1996 </published>
11 <page_num> 224 Pages</page_num>
12 <isbn>9780804115827</isbn>
13 <url>https://www.penguinrandomhouse.com/books/123094/the-new-york-times-daily-crossword-puzzles-thursday-volume-1-by-eugene-maleska/</url>
14 </metadata>
15 </book>
16 <book date="2018-08-18" xml:lang="en">
17 <title>Creatures of the Night (Second Edition)</title>
18 <body>Two of literary comics modern masters present a pair of magical and disturbing stories of strange creatures who are not quite what they seem! In The Price, a mys
19
20 <metadata>
21 <topics>
22 <d0>Fiction</d0>
23 <d1>Graphic Novels and Manga</d1>
24 </topics>
25 <author>Neil Gaiman</author>
26 <published>Nov 29, 2016 </published>
27 <page_num> 48 Pages</page_num>
28 <isbn>9781506700250</isbn>
29 <url>https://www.penguinrandomhouse.com/books/539586/creatures-of-the-night-second-edition-by-written-and-created-by-neil-gaiman-illustrated-by-michael-zulli/</url>
30 </metadata>
31 </book>
32 <book date="2018-08-18" xml:lang="en">
33 <title>Cornelia and the Audacious Escapades of the Somerset Sisters</title>
34 <body>Eleven-year-old Cornelia is the daughter of two world-famous pianists—a legacy that should feel fabulous, but instead feels just plain lonely. She surrounds hers
35
36 <metadata>

```

BlurbGenreCollection_EN_test.xml



```

1 <root>
2 <book date="2018-08-18" xml:lang="en">
3 <title>Teenage Mutant Ninja Turtles: The Box Set Volume 1</title>
4 <body>TMNT co-creator Kevin Eastman and writer Tom Waltz guide readers through a ground-breaking new origin and into epic tales of courage, loyalty and family as the T
5
6 <metadata>
7 <topics>
8 <d0>Fiction</d0><d1>Graphic Novels and Manga</d1>
9 </topics>
10 <author>Kevin Eastman, Tom Waltz</author>
11 <published>Dec 15, 2015 </published>
12 <page_num> 520 Pages</page_num>
13 <isbn>9781631404726</isbn>
14 <url>https://www.penguinrandomhouse.com/books/560867/teenage-mutant-ninja-turtles-the-box-set-volume-1-by-kevin-eastman/</url>
15 </metadata>
16 </book>
17 <book date="2018-08-18" xml:lang="en">
18 <title>Betty and Veronica: Fairy Tales</title>
19 <body>Take a magical trip down Storybook Lane with this collection of the best fairy tale stories ever told in Riverdale! From glass slippers to giant beanstalks and y
20
21 <metadata>
22 <topics>
23 <d0>Children's Books</d0><d1>Children's Middle Grade Books</d1>
24 </topics>
25 <author>Archie Superstars</author>
26 <published>Dec 06, 2016 </published>
27 <page_num> 256 Pages</page_num>
28 <isbn>9781627388948</isbn>
29 <url>https://www.penguinrandomhouse.com/books/537197/betty-and-veronica-fairy-tales-by-archie-superstars/</url>
30 </metadata>
31 </book>
32 <book date="2018-08-18" xml:lang="en">
33 <title>Friends and Fauxs</title>
34 <body>Tracie Howard is back with all of the Gucci, glitz, and glamour in this steamy follow-up to her smash hit Gold Diggers! Gillian Tillman learned all about landing

```

METHODOLOGY

First, we found it necessary to understand Natural Language Processing and the process of creating classification models. Once we did that we broke down our approach to the problem in the following subtopics:

Dataset identification and balancing.

Dataset cleaning.

Creating vectors of the book summaries.

Training a model to use these vectors to predict genre of a new book.

We established that we will be working with summaries of book in place of the entire text owing to the increase in complexity of large texts and the resources required for the same.

We defined the features of a genre as words or phrases that are specific to that genre. For example, “travelling through time” may be used in science fiction and is specific to that genre. So in case a new book has this phrase, the chances of it belonging to science fiction will be higher.

It is also possible that the same feature may belong to multiple genres, for example “time stopped” could belong to romance as well as science fiction. This meant that features could also be overlapping.

Hence, it is important to maintain the context of words and not solely depend on individual word or phrase as features. Also the frequency of a word did not provide a suitable metric.

We decided to use a TF-IDF technique to quantify the words in the summary. N-grams were also an important concept for preserving the context of words.

For training the model, we would approach various classifiers available through libraries like scikit-learn and how the accuracy was affected by changing various parameters.

IMPLEMENTATION

Data Distribution

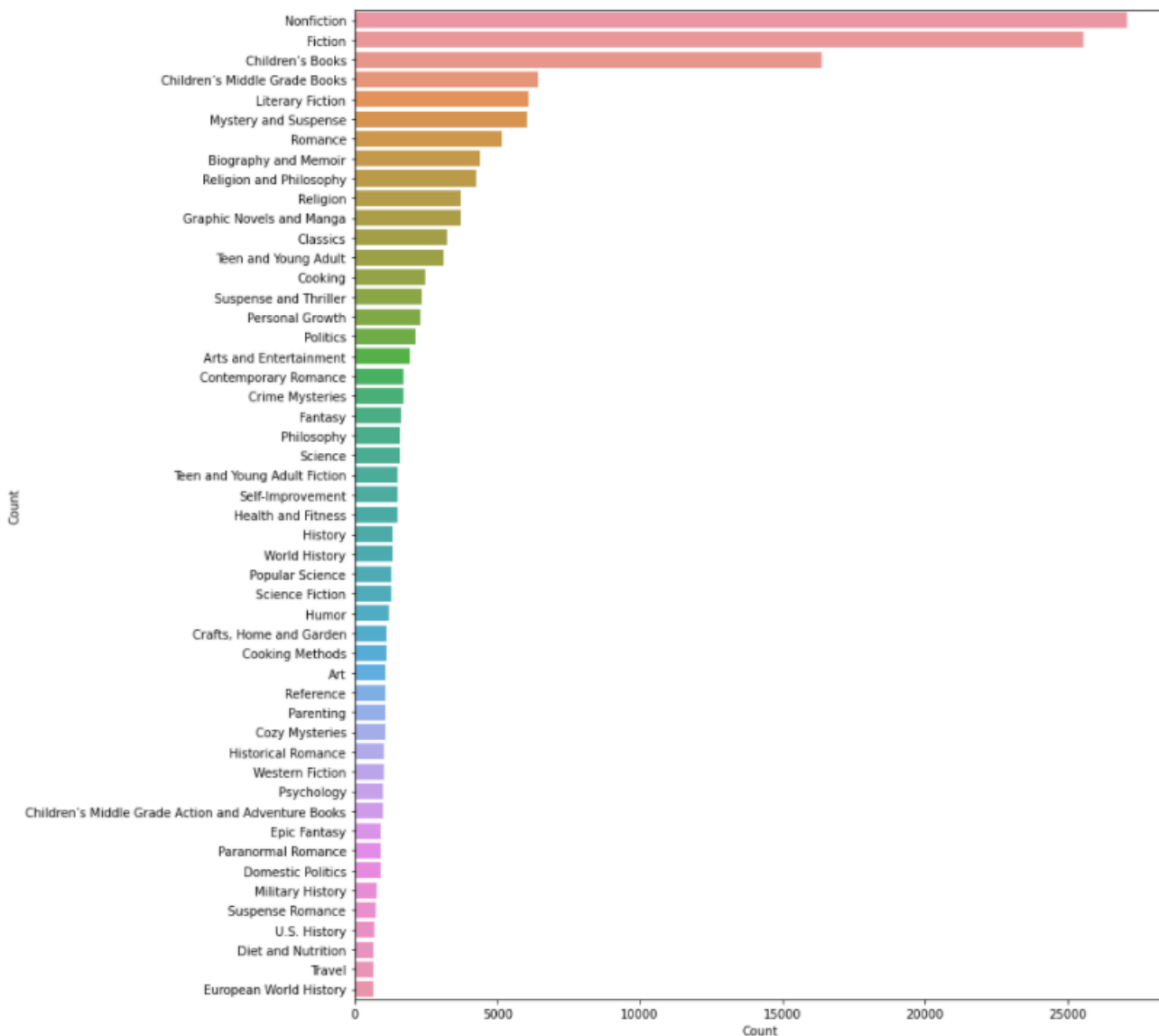


Figure 1: Distribution of Blurb Genre Collection

Data Extraction and Cleaning

The data is available in XML format so the first task was to extract the metadata from the file to create a data frame. Each blurb is categorized into one or multiple categories. The categories are structured hierarchically. The **minimum code policy** requires the assignment of at least one category to each document of the collection. The **hierarchy policy** ensures that every ancestor of a document's label is assigned as well. To make the access to data easier we created lists of genres to which each blurb belongs.

```
def getDataFrame(filename):
    dataset=[]
    tree = ElementTree.parse(filename)
    root = tree.getroot()

    for book in root.findall('book'):
        for metadata in book.find('metadata'):
            x=metadata
            break
        a=[]
        if(x.find('d3')!=None):
            a.append(x.find('d3').text)
        if(x.find('d2')!=None):
            a.append(x.find('d2').text)
        if(x.find('d1')!=None):
            a.append(x.find('d1').text)
        if(x.find('d0')!=None):
            a.append(x.find('d0').text)

        if len(a) != 0:
            dataset.append([book.find('title').text,a,book.find('body').text])
    return dataset

dataset = getDataFrame('/content/drive/MyDrive/Colab Notebooks/Books Categorizer/BlurbGenreCollection_EN_train.xml')
dataset2 = getDataFrame('/content/drive/MyDrive/Colab Notebooks/Books Categorizer/BlurbGenreCollection_EN_test.xml')
dataset = dataset + dataset2
```

```
df=pd.DataFrame(dataset,columns=['Name','Genres','Summary'])
```

```
[ ] df
```

	Name	Genres	Summary
0	The New York Times Daily Crossword Puzzles: Thursday, Volume 1	[Games, Nonfiction]	Monday's Crosswords Do with EaseTuesday's Crosswords Not a BreezeWednesday's Crosswords Harder StillThursday's Crosswords Take Real SkillFriday's Crosswords — You've Come This Far...Saturday's Crosswords — You're a Star!For millions of people, the New York Times crossword puzzles are as essential ...
1	Creatures of the Night (Second Edition)	[Graphic Novels and Manga, Fiction]	Two of literary comics modern masters present a pair of magical and disturbing stories of strange creatures who are not quite what they seem! In The Price, a mysterious feline engages in a nightly conflict with an unseen, vicious foe. The Daughter of Owls recounts an eerie tale of a beautiful or...
2	Cornelia and the Audacious Escapades of the Somerset Sisters	[Children's Middle Grade Books, Children's Books]	Eleven-year-old Cornelia is the daughter of two world-famous pianists—a legacy that should feel fabulous, but instead feels just plain lonely. She surrounds herself with dictionaries and other books to isolate herself from the outside world. But when a glamorous neighbor named Virginia Somerset ...
3	The Alchemist's Daughter	[Historical Fiction, Fiction]	During the English Age of Reason, a woman cloistered since birth learns that knowledge is no substitute for experience. Raised by her father in near isolation in the English countryside, Emilie Selden is trained as a brilliant natural philosopher and alchemist. In the spring of 1725, father and d...
4	Dangerous Boy	[Teen and Young Adult Mystery and Suspense, Teen and Young Adult]	A modern-day retelling of The Strange Case of Dr. Jekyll and Mr. Hyde with a chilling twist Harper has never been worried about falling in love, something she is skeptical even exists. But everything changes when Logan moves to town, and to Harper's shock, the two tumble into an intense romance....
...
77104	Cathedral of the Sea	[Romance, Fiction]	An unforgettable fresco of a golden age in fourteenth-century Barcelona, Cathedral of the Sea is a thrilling historical novel of friendship and revenge, plague and hope, love and war. Arnau Estanyol arrives in Barcelona to find a city dominated by the construction of the city's great pride—the...
77105	Understanding the Messages of Your Body	[Alternative Therapies, Health and Fitness, Nonfiction]	Fears, anxieties, traumas, and physical and emotional shocks imprint on the body and remain dormant in its vast memory store until they are roused by an event or encounter. They may manifest in a different form or place—a fearful incident may transform itself into a stomachache or a headache, or...
77106	Knockout	[Alternative Therapies, Health and Fitness, Nonfiction]	In Knockout, Suzanne Somers interviews doctors who are successfully using the most innovative cancer treatments—treatments that build up the body rather than tear it down. Somers herself has stared cancer in the face, and a decade later she has conquered her fear and has emerged confident with t...
77107	The End of the Suburbs	[Domestic Politics, Politics, Nonfiction]	"The government in the past created one American Dream at the expense of almost all others: the dream of a house, a lawn, a picket fence, two children, and a car. But there is no single American Dream anymore."For nearly 70 years, the suburbs were as American as apple pie. As the middle class ba...
77108	First Aid Fast for Babies and Children	[Parenting, Nonfiction]	An indispensable guide for all parents and caretakers that covers a wide range of childhood emergencies.From anaphylaxis to burns to severe bleeding and bruising, First Aid Fast for Babies and Children offers clear advice, and step-by-step photographs show you what to do.This revised and updated...

77109 rows × 3 columns

Data Augmentation

Data augmentation is commonly used in computer vision. In vision, you can almost certainly flip, rotate, or mirror an image without risk of changing the original label.

But there is some difference when we are working with text, especially summaries. We use one simple operation for data augmentation, such that it does not change the genre of the book.

Synonym Replacement: Randomly choose n words from the sentence that are not stop words. Replace each of these words with one of its synonyms chosen at random.

Functions for Data Augmentation:

```
def synonym_replacement(words, n):
    new_words = words.copy()
    random_word_list = list(set([word for word in words if word not in stop_words]))
    random.shuffle(random_word_list)
    num_replaced = 0
    for random_word in random_word_list:
        synonyms = get_synonyms(random_word)
        if len(synonyms) >= 1:
            synonym = random.choice(list(synonyms))
            new_words = [synonym if word == random_word else word for word in new_words]
            num_replaced += 1
        if num_replaced >= n:
            break

    sentence = ' '.join(new_words)
    new_words = sentence.split(' ')

    return new_words
```

```
[ ] def get_synonyms(word):
    synonyms = set()
    for syn in wordnet.synsets(word):
        for l in syn.lemmas():
            synonym = l.name().replace("_", " ").replace("-", " ").lower()
            synonym = "".join([char for char in synonym if char in 'qwertyuiopasdfghjklzxcvbnm'])
            synonyms.add(synonym)
    if word in synonyms:
        synonyms.remove(word)
    return list(synonyms)
```

```
[ ] def applysynonym(row):
    words = nltk.word_tokenize(row)
    n = len(words)
    res = synonym_replacement(words, n//10)
    return ' '.join(res)
```

```
def augmentdata(row, n):
    global df
    t1 = row
    for i in range(n):
        t1.Summary = applysynonym(row.Summary)
        df = df.append(t1, ignore_index=True)
    return
```

```

def do_augmentation():
    global df
    global genre_count
    len_df = len(df)
    for index, row in df.iterrows():
        genres = row.Genres
        append_data = True
        for g in genres:
            if genre_count[g] >= 20000:
                append_data = False
                break
        if append_data:
            augmentdata(row, 10)
            for g2 in genres:
                genre_count[g2] += 10
    printProgressBar(index + 1, len_df, prefix = 'Augmenting data:', suffix = 'Complete', length = 50)

```

```

for i in range(5):
    do_augmentation()

```

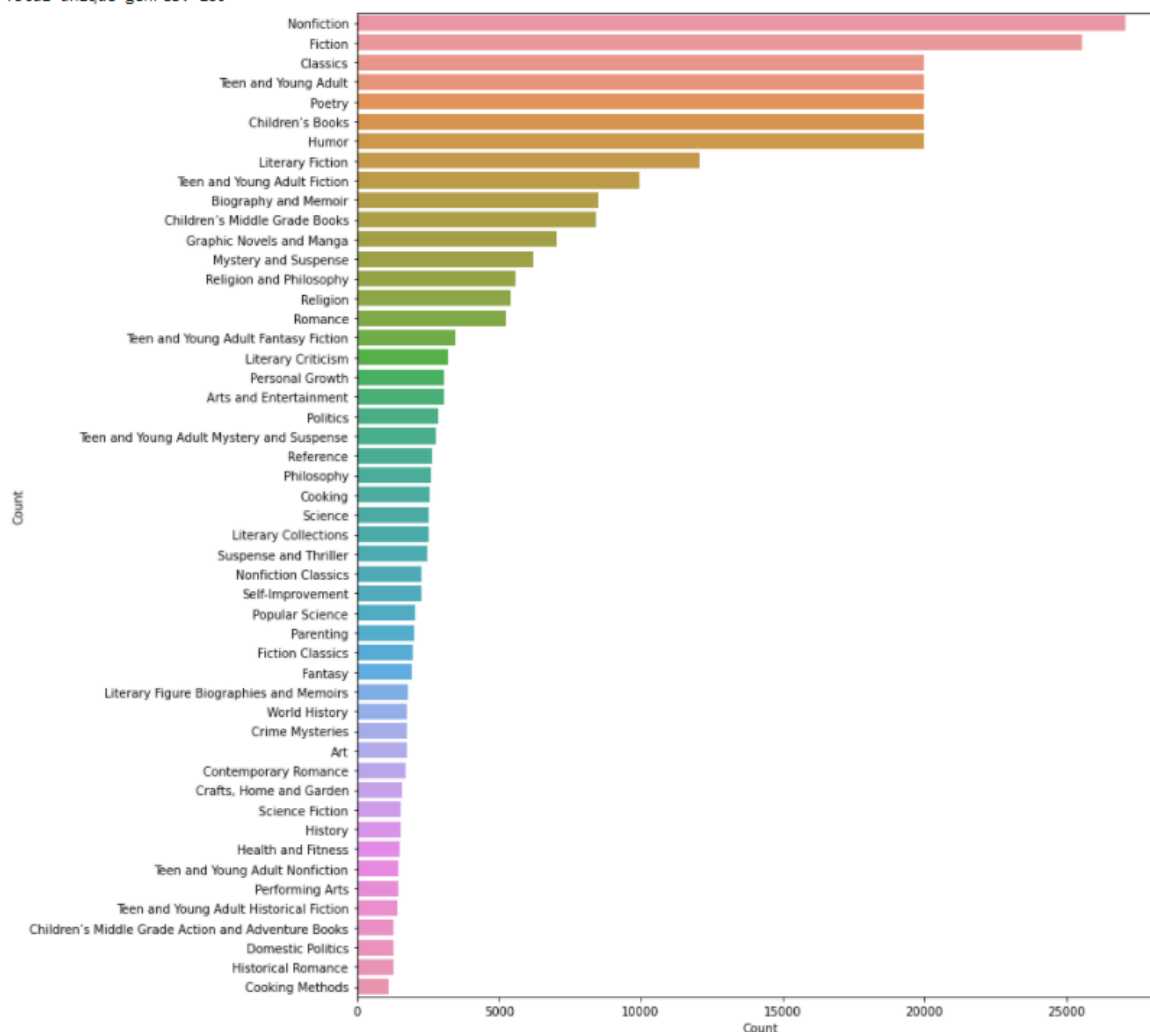
```

Augmenting data: | 100.0% Complete
Augmenting data: | 100.0% Complete
Augmenting data: | 100.0% Complete
Augmenting data: | 100.0% Complete
Augmenting data: | 100.0% Complete

```

```
[ ] draw_genre_plot(df)
```

Total unique genres: 139



Removing stop words and finding frequent words

```
[ ] def clean_text(text):
    text = re.sub("\'", "", text)
    text = re.sub("[^a-zA-Z]", " ", text)
    text = ' '.join(text.split())
    text = text.lower()
    return text

def remove_stopwords(text):
    no_stopword_text = [w for w in text.split() if not w in stop_words]
    return ' '.join(no_stopword_text)
```

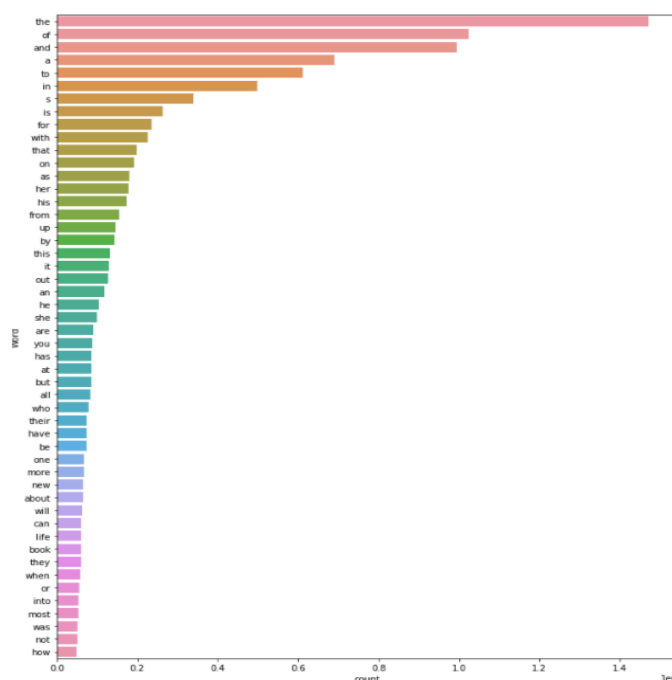
```
def freq_words(x, terms = 30):
    word_count = {}
    for index, row in df.iterrows():
        text = row["Summary"]
        for w in nltk.word_tokenize(text):
            try:
                word_count[w] += 1
            except:
                word_count[w] = 1

    fdist = nltk.FreqDist(word_count)
    words_df = pd.DataFrame({'word': list(word_count.keys()), 'count': list(word_count.values())})
    d = words_df.nlargest(columns="count", n = terms)
    plt.figure(figsize=(12,15))
    ax = sns.barplot(data=d, x= "count", y = "word")
    ax.set(ylabel = 'Word')
    plt.show()
```

```
[ ] df['Summary'] = df['Summary'].apply(lambda x: clean_text(x))
```

The most frequent words out of all the words in the data:

```
freq_words(df['Summary'], 50)
```

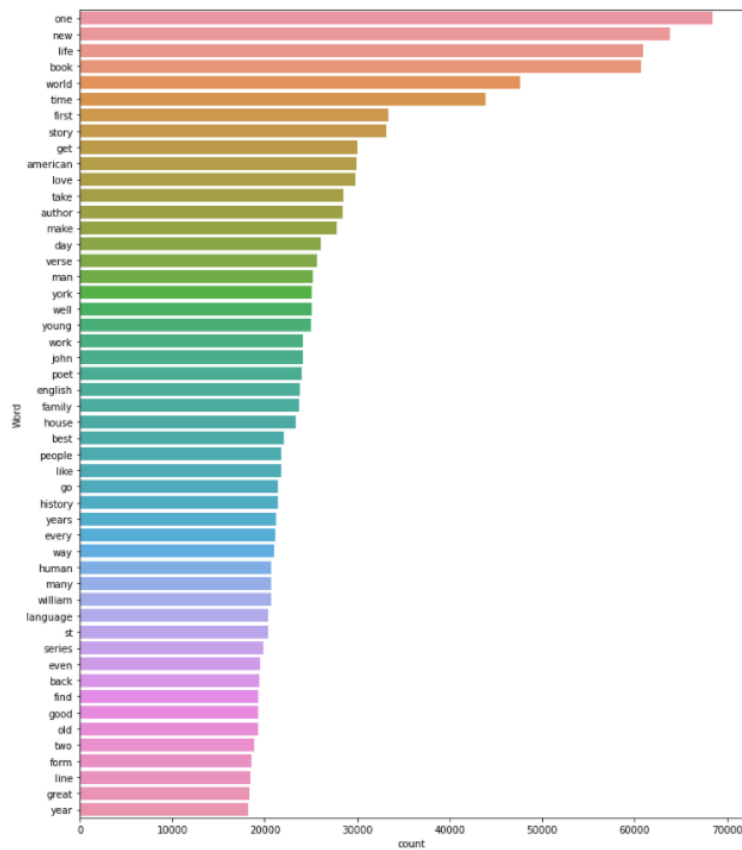


Thus, we remove all stopwords for relevant data.

```
[ ] df['Summary'] = df['Summary'].apply(lambda x: remove_stopwords(x))
```

The most frequent words after removing stopwords is as follows:

```
freq_words(df['Summary'], 50)
```



```
[ ] df.head()
```

	Name	Genres	Summary
0	The New York Times Daily Crossword Puzzles: Thursday, Volume 1	[Games, Nonfiction]	monday crosswords easetuesday crossword puzzles breezewednesday crosswords harder stillthursday crosswords take real skillfriday crosswords come far saturday crosswords star millions people new york times crossword puzzles essential day first cup coffee morning first time ever premier puzzles available...
1	Creatures of the Night (Second Edition)	[Graphic Novels and Manga, Fiction]	two literary comics modern masters present pair magical disturbing stories strange creatures quite seem price mysterious feline engages nightly conflict unseen vicious foe daughter owls recounts eerie tale beautiful orphan girl found clutching owl pellet would wrong would face bizarre unforeseen...
2	Cornelia and the Audacious Escapades of the Somerset Sisters	[Children's Middle Grade Books, Children's Books]	eleven year old cornelia daughter two world famous pianists legacy feel mythical instead feels plain lonely surrounds lexicon books isolate outside world glamorous neighbor key old dominion state somerset moves next door servant patel mischievous gallic bulldog key mister kinyatta cornelia disco...
3	The Alchemist's Daughter	[Historical Fiction, Fiction]	english age reason woman cloistered since birth learns knowledge substitute experience raised father near isolation english countryside emilie selden trained brilliant natural philosopher alchemist spring father daughter embark upon daring alchemical experiment date attempting breathe life dead ...
4	Dangerous Boy	[Teen and Young Adult Mystery and Suspense, Teen and Young Adult]	modern day retelling strange case dr jekyll mr hyde chilling twist harper never worried falling love something skeptical even exists everything changes logan moves town harper shock two tumble intense romance everything never thought wanted meets logan twin brother caleb expelled last school tru...

Multi-label classification is a generalization of multi-class classification which is the single-label problem of categorizing instances into precisely one of more than two classes, in the multi-label problem there is no constraint on how many of the classes the instance can be assigned to i.e. there could be one, two or many labels in the output data used for training.

Metric used:

F1 Score: F1 score is calculated using the harmonic mean of precision and recall.

$$\text{F1 Score} = 2 * (\text{precision} * \text{recall}) / (\text{precision} + \text{recall})$$

This F1 score is micro averaged to use it as a metric for multi-class classification. It is calculated by counting the value of true positives, false positives, true negatives, and false negatives. All the predicted outputs, in this case, are column indices and are used in sorted order by default.

Modifying the dataset for Multi label:

Although a list of sets or tuples is a very intuitive format for multilabel data, it is difficult to process. We modify the dataset to create a binary matrix, such that each genre is a separate column. There are 139 unique genres. If a blurb belongs to a genre the value for that column will be 1 otherwise 0. Another approach uses the `multilabel_binarizer` provided by scikit-learn. This transformer converts between the intuitive format and the supported multilabel format: a (samples x classes) binary matrix indicating the presence of a class label.

We have followed both approaches to get similar outputs.

Now we use a TF-IDF vectorizer from the scikit-learn library to vectorise our summaries. We can also specify the ngram length. A `TfidfVectorizer` converts a collection of raw documents to a matrix of TF-IDF features which can be fed to classifiers.

```
from sklearn.preprocessing import MultiLabelBinarizer
from sklearn.pipeline import Pipeline
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer, TfidfTransformer
#from sklearn import preprocessing

multilabel_binarizer = MultiLabelBinarizer()
multilabel_binarizer.fit(df['Genres'])

# transform target variable
y = multilabel_binarizer.transform(df['Genres'])
```

```
xtrain, xval, ytrain, yval = train_test_split(df['Summary'], y, test_size=0.2, random_state=9)
```

After vectorising, we will use a One vs. Rest classifier for multi label prediction. Also known as one-vs-all, this strategy consists in fitting one classifier per class. For each classifier, the

class is fitted against all the other classes. In addition to its computational efficiency (only n -classes classifiers are needed), one advantage of this approach is its interpretability. Since each class is represented by one and one classifier only, it is possible to gain knowledge about the class by inspecting its corresponding classifier. This is the most commonly used strategy for multiclass classification and is a fair default choice.

This strategy can also be used for multilabel learning, where a classifier is used to predict multiple labels for instance, by fitting on a 2-d matrix in which cell $[i, j]$ is 1 if sample i has label j and 0 otherwise.

In the multilabel learning literature, OvR is also known as the binary relevance method.

We need to provide an estimator as a parameter to the OneVsRest Classifier, an estimator is an object implementing fit or predict_proba.

```
[ ] xtrain, xval, ytrain, yval = train_test_split(df['Summary'], y, test_size=0.2, random_state=9)

[ ] from sklearn.linear_model import LogisticRegression

    # Binary Relevance
    from sklearn.multiclass import OneVsRestClassifier

    # Performance metric
    from sklearn.metrics import f1_score

[ ] import math
    lr = LogisticRegression()

    clf = Pipeline([('vect', CountVectorizer(ngram_range=(1,2), max_df=0.8, max_features=10000)),
                    ('tfidf', TfidfTransformer()),
                    ('ovr', OneVsRestClassifier(lr))
                    ])

[ ] clf.fit(xtrain, ytrain)
```

Predicting the genres

Using predict_proba: Probability estimates.

```
▶ y_pred_prob = clf.predict_proba(xval)
  t = 0.2505
  y_pred_new = (y_pred_prob >= t).astype(int)
```

```
[ ] f1_score(yval, y_pred_new, average="micro")
```

```
0.7712376455523989
```

We achieved an F1 score of 0.771 using this model.

The returned estimates for all classes are ordered by label of classes.

Note that in the multilabel case, each sample can have any number of labels. This returns the marginal probability that the given sample has the label in question. For example, it is entirely consistent that two labels both have a 90% probability of applying to a given sample.

Printing the predicted genre classes:

We use a threshold value to determine whether or not a book belongs to that genre. Applying different thresholds to the same prediction helps us identify the optimal value which is 0.25 in our case. A very large or a very small value of threshold gives a lower value of F1 metric score because when tags are chosen based on a lower threshold value, too many tags get chosen which reduce the F1 metric score, while when the threshold value gets very large, almost no tags get chosen and thus reducing the performance metric.

We then use `inverse_transform` to get string values of the classes from the binarizer.

```
[ ] def infer_tags(q):
    q = clean_text(q)
    q = remove_stopwords(q)
    q_pred_prob = clf.predict_proba([q])
    t = 0.2505
    q_pred_new = (q_pred_prob >= t).astype(int)

    return multilabel_binarizer.inverse_transform(q_pred_new)
```

```
[ ] for i in range(5):
    k = xval.sample(1).index[0]
    print("Book: ", df['Name'][k], "\nPredicted genre: ", infer_tags(xval[k])), print("Actual genre: ", df['Genres'][k], "\n")
```


The actual label and predicted labels:

Book: To the Edge of the World
 Predicted genre: [('Teen and Young Adult',)]
 Actual genre: ['Teen and Young Adult Historical Fiction', 'Teen and Young Adult']

Book: Thieves Get Rich, Saints Get Shot
 Predicted genre: [('Fiction', 'Mystery and Suspense')]
 Actual genre: ['Suspense and Thriller', 'Mystery and Suspense', 'Fiction']

Book: Peace, Love, and Mad Libs
 Predicted genre: [('Humor',)]
 Actual genre: ['Children's Middle Grade Books', 'Children's Books']

Book: American Heart Association Eat Less Salt
 Predicted genre: [('Cooking', 'Diet and Nutrition', 'Nonfiction')]
 Actual genre: ['Diet and Nutrition', 'Cooking', 'Nonfiction']

Book: Plenty
 Predicted genre: [('Poetry',)]
 Actual genre: ['Poetry']

Predicted outcomes of some books based on our model

```
[ ] # The Adventures of Tom Sawyer
infer_tags("The Adventures of Tom Sawyer is an 1876 novel by Mark Twain about a boy growing up along the Mississippi River. It is set in the 1840s in the town of St. Petersburg, which is based on Hannibal,
[('Children's Books', 'Children's Middle Grade Books', 'Literary Fiction')]
```

```
[ ] # Pride and Prejudice
infer_tags("Pride and Prejudice is set in rural England in the early 19th century, and it follows the Bennet family, which includes five very different sisters. Mrs. Bennet is anxious to see all her daught
[('Fiction', 'Historical Romance', 'Literary Fiction', 'Romance')]
```

```
[ ] # Harry Potter
infer_tags("Adaptation of the first of J.K. Rowling's popular children's novels about Harry Potter, a boy who learns on his eleventh birthday that he is the orphaned son of two powerful wizards and possess
[('Children's Books', 'Fiction', 'Teen and Young Adult')]
```

```
[ ] # The Theory of Everything
infer_tags("Based on a series of lectures given at Cambridge University, Professor Hawking's work introduced the history of ideas about the universe as well as today's most important scientific theories ab
[('Nonfiction', 'Popular Science', 'Science')]
```

```
[ ] # The Merchant of Venice
infer_tags("The Merchant of Venice is a 16th-century play written by William Shakespeare in which a merchant in Venice named Antonio defaults on a large loan provided by a Jewish moneylender, Shylock. It i
[('Classics',)]
```

```
[ ] # Around the World in 80 days
infer_tags("Around the World in Eighty Days is an adventure novel by the French writer Jules Verne, first published in French in 1872. In the story, Phileas Fogg of London and his newly employed French val
[('Classics', 'Fiction', 'Literary Fiction')]
```

```
• # My Experiments with Truth -by Mahatma Gandhi
infer_tags("The Story of My Experiments with Truth is the autobiography of Mahatma Gandhi, covering his life from early childhood through to 1921. It was written in weekly installments and published in his
[('Biography and Memoir', 'Classics')]
```

+ Code + Text

```
[ ] # The Shining
infer_tags("The Story of My Experiments with Truth is the autobiography of Mahatma Gandhi, covering his life from early childhood through to 1921. It was written in weekly installments and published in his
[('Biography and Memoir', 'Classics')]
```








```
[ ] # A Brief History of Time
infer_tags("A Brief History of Time: From the Big Bang to Black Holes is a book on theoretical cosmology by English physicist Stephen Hawking. It was first published in 1988. Hawking wrote the book for rea
[('Children's Books', 'Nonfiction')]
```

CONCLUSION

We were able to train an efficient model that had high accuracy. The model performed well on random inputs outside the dataset as well. We tried pursuing different approaches to solve the same problem in that process we were able to create this model which showed some very good results.

We have come to the understanding that it is important to maintain the context of the words in the summary as well as to ensure that we are accounting for overlaps between various genres. Classification of books into genres is a complex task, and our model works well.

REFERENCES

-  <https://towardsdatascience.com/tf-idf-for-document-ranking-from-scratch-in-python-on-real-world-dataset-796d339a4089>
-  <https://towardsdatascience.com/multi-class-text-classification-model-comparison-and-selection-5eb066197568>
-  <https://aclanthology.org/C18-1167.pdf>
-  <https://towardsdatascience.com/these-are-the-easiest-data-augmentation-techniques-in-natural-language-processing-you-can-think-of-88e393fd610>
-  <https://towardsdatascience.com/beyond-accuracy-precision-and-recall-3da06bea9f6c>
-  <https://scikit-learn.org/stable/tutorial/index.html>
-  <https://medium.com/coinmonks/multi-label-classification-blog-tags-prediction-using-nlp-b0b5ee6686fc>