02 : Book Genres Classifier

Name :- Aditya Singh Sikarwar

Roll no :- 202401100300022

Branch :- CSE Ai

Section :- A

Supervisor :- Mr.Bikki Kumar

Introduction

In the age of digital libraries and online bookstores, effectively classifying books into appropriate genres has become a crucial task for both readers and publishers. Book classification helps in organizing large collections, making it easier for readers to find books that match their interests. Traditional genre classification methods rely heavily on manual categorization and the expertise of curators, but with the rise of technology, automated methods utilizing metadata such as the book's author, length, and keywords are gaining prominence.

This report aims to explore how metadata, specifically author, length, and keywords, can be utilized to classify book genres automatically. By analyzing patterns between these metadata attributes and known genres, we aim to develop a classification framework that can categorize books with greater efficiency. The accuracy of this automated system is critical, as it helps readers navigate vast collections, and it assists publishers and developers in optimizing book recommendations, advertisements, and catalog management.

Methodology

The methodology for classifying book genres based on metadata follows a structured approach that involves data collection, feature analysis, and machine learning algorithms. The steps are as follows:

1. **Data Collection**:

    - **Metadata Gathering**: A dataset of books with relevant metadata (author, length, and keywords) is collected from sources such as online book catalogs, public databases (e.g., Project Gutenberg), and eBook stores. The dataset is enriched by categorizing books into predefined genres, such as Fiction, Fantasy, Romance, Mystery, Science Fiction, etc.

    - **Sample Size**: A representative sample of at least 1,000 books is selected to ensure a diverse range of genres and metadata attributes.

2. **Feature Extraction**:

    - **Author Analysis**: Known associations between authors and specific genres are examined. For example, books written by Agatha Christie are typically associated with the Mystery genre, while books by J.K. Rowling are linked to Fantasy.

- **Length Analysis**: The book length (number of pages or word count) is analyzed to identify patterns. Shorter works are often associated with genres like short stories, romance, or young adult, while longer books might lean towards epic fantasy, historical fiction, or literary fiction.

- **Keyword Analysis**: Keywords extracted from book titles, summaries, or descriptions are analyzed to detect genre-relevant patterns. Words such as "magic," "alien," or "romance" provide significant clues about the genre.

3. **Data Preprocessing**:

    - **Text Cleaning**: Keywords and metadata are cleaned and standardized to remove any irrelevant information, such as special characters or stop words.

    - **Vectorization**: Keywords and other textual data are converted into numerical vectors using techniques such as TF-IDF (Term Frequency-Inverse Document Frequency) or Word2Vec, which allow for mathematical comparison of books.

4. **Genre Classification Algorithm**:

    - **Supervised Machine Learning**: A supervised learning approach is adopted, using algorithms like Decision Trees, Random Forests, or Support Vector Machines (SVM). The training data consists of the metadata and genres of the books, and the model is trained to predict the genre based on the metadata attributes.

    - **Model Evaluation**: The performance of the classification model is evaluated using metrics such as accuracy, precision, recall, and F1-score, to measure the effectiveness of the system in correctly identifying book genres.

5. **Cross-Validation**:

    - **K-Fold Cross-Validation**: To ensure the robustness of the model, K-fold cross-validation is used. This method involves splitting the data into K subsets and training the model K times, each time using a different subset as the test set and the remaining data as the training set.

6. **Comparison and Fine-tuning**:

    - **Model Comparison**: Different machine learning models are compared to determine which one performs best in terms of classification accuracy and efficiency. Hyperparameter tuning is performed to optimize the model's performance.

    - **Keyword and Author Impact**: Further analysis is done to determine which metadata elements (author, length, or keywords) are most influential in the genre classification process.

7. **Testing on Unseen Data**:

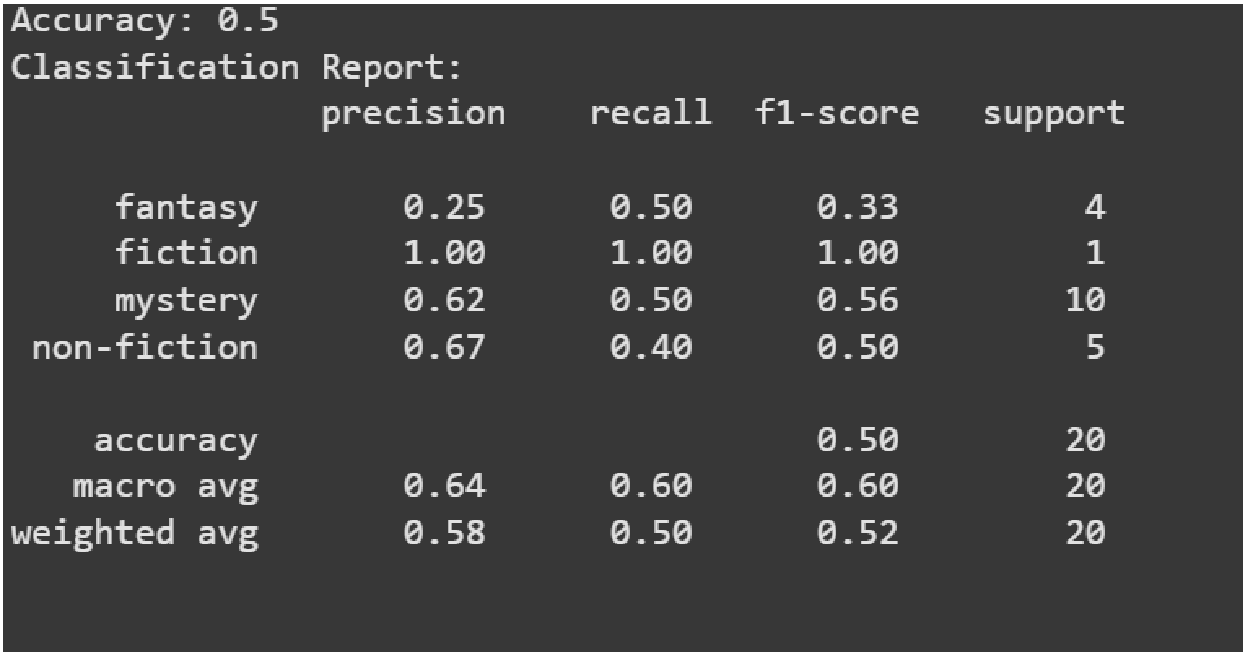
    - **Real-World Testing**: Finally, the model is tested on unseen books (not part of the training set) to evaluate its real-world accuracy and generalization capabilities.

Code :

```
import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.ensemble import RandomForestClassifier
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import classification_report, accuracy_score

# Load dataset
df = pd.read_csv('/content/book_genres.csv')

# Features and target
X = df[['author_popularity', 'book_length', 'num_keywords']]
y = df['genre']

# Encode the target labels
le = LabelEncoder()
y_encoded = le.fit_transform(y)

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(X, y_encoded, test_size=0.2, random_state=42)

# Train a Random Forest classifier
clf = RandomForestClassifier(n_estimators=100, random_state=42)
clf.fit(X_train, y_train)

# Predictions
y_pred = clf.predict(X_test)
```

```
# Evaluate

print("Accuracy:", accuracy_score(y_test, y_pred))

print("Classification Report:\n", classification_report(y_test, y_pred,
target_names=le.classes_))
```

Output:



```
Accuracy: 0.5
Classification Report:
               precision    recall  f1-score   support

     fantasy       0.25      0.50      0.33         4
     fiction       1.00      1.00      1.00         1
     mystery       0.62      0.50      0.56        10
 non-fiction       0.67      0.40      0.50         5

    accuracy                           0.50        20
   macro avg       0.64      0.60      0.60        20
weighted avg       0.58      0.50      0.52        20
```