Final Project
Applied Machine Learning
Aryaman Himatsingka

The data used in this project can be found here. The GitHub repository can be found here.

Introduction and Problem Statement

Genres are widely used for organizing books, such as in bookstores, libraries, online forums, and online bookstores; yet genre assignment is often manual and subjective. This project explores whether book genres - assigned by human curators in a large literary dataset - correspond to real, learnable patterns in plot summaries. We evaluate whether machine learning can reproduce or improve genre classification using only plot summaries.

Our primary audience is content managers, librarians, and literary theorists at institutions like libraries, online bookstores (Amazon, Goodreads, etc), and archival services who face challenges in classifying large volumes of content at scale.

If machine learning can identify clearer or more accurate genre boundaries, sub-genres and trends, they can provide intrinsic value to the above audience in the form of improved automated tagging systems, reduced metadata inconsistencies, capturing trends in subgenres (especially for online forums and bookstores), and reduced time in cataloguing and curation.

In this project, we utilize data from the CMU Book Summary database, and through embedding techniques such as TF-IDF, Bag of Words, Word2Vec, etc., and algorithms such as Logistic Regression, SVM and Random Forests, we can explore the best combination of embedding and algorithm for our classification task. Then, utilizing our embedding, we perform clustering to determine the clusters of genres that are prevalent. The number of clusters and their distance from each other help us delineate the existence of genres, sub-genres and mixed labels. We can then compare it to the classification from the database to verify the accuracy of our work.

Highlighted Result

In our project, we see that supervised learning, specifically a combination of TF-IDF embedding and Support Vector Machines (Linear Kernel) perform well in the multilabel classification task of genre classification. This combination displays a high F-1 score, with a low hamming loss, and out of the combinations of the different embeddings and algorithms, this combination performed the best.

We also utilize unsupervised learning, in the form of k-means clustering on TF-IDF embeddings, to serve as a baseline to improve upon with our supervised learning. It also serves the larger purpose of delineating genres and sub-genres, and seeing the boundaries between them. The optimal number of clusters, which is 6, is lower than the number of genres we use (7 genres), and thus, we are able to see the underlying structure of the books summaries and genre classification and analyze them. We see the existence of certain well-defined sub-genres (Sci-Fi + Fantasy, Mystery + Suspense), as well as certain clusters showing not as well-defined sub-genres.

Dataset Description

The database used in this project is the CMU Book Summary Database. This database consists of over 16,000 books, and over 227 genres. This database is managed and curated by researchers, and contains the following fields stores as JSON data: plot summary, author, publication date and genres. This data is introduced in the code as the 'booksummaries.txt' file.

This dataset was selected and can be justified as large due to several reasons. It is large in dimensions, with over 16,000 rows, and also contains long-form unstructured text data (plot summaries. This translates to a high-dimensional sparse feature space after vectorization, which we will see in the embeddings. There is also a high label cardinality due to the large number of possible genre labels.

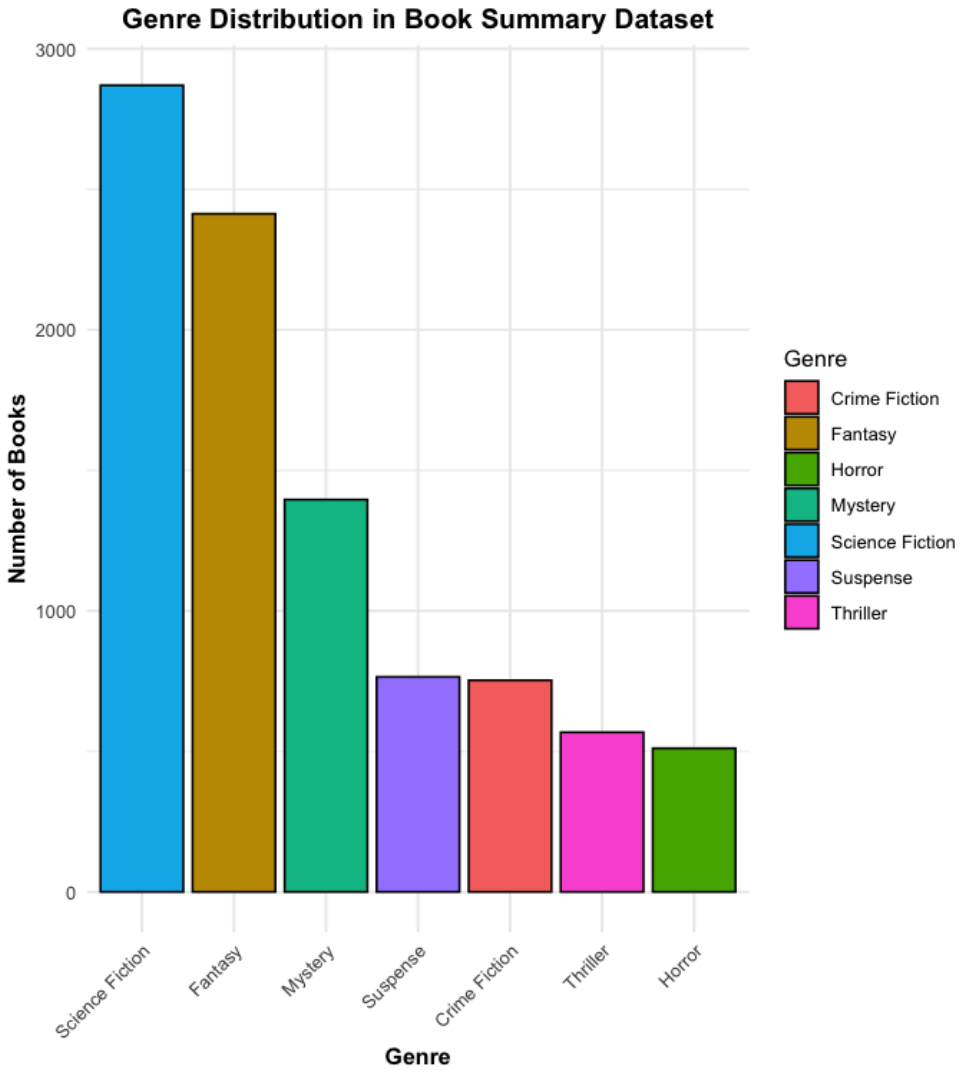
The structure of the dataset can be seen in the image below, which shows a sample book. The dataset consists of 6 columns - Wikipedia ID, Freebase ID, Book title, Book author, Publication date and Genres. The dataset was created as a part of the paper, 'New Alignment Methods for Discriminative Book Summarization', by David Bamman and Noah A Smith of Carnegie Mellon University. They are the managers of this dataset as it is still hosted by the CMU Computer Science Department, and responsible for its quality.

Wikipedia ID	1166383
Freebase ID	/m/04cvx9
Book title	White Noise
Book author	Don DeLillo
Publication date	1985-01-21
Genres	Novel, Postmodernism, Speculative fiction, Fiction

Exploratory Data Analysis

We performed several steps to prepare the data and perform quality checks. Initially, we parsed the JSON data into several readable labels, separating the different genres, author and plot summary.

We then looked at the distribution of the different genre labels, looking at the counts of the different genres across the database. We see that there occur only 7 genres with high frequency, and chose to focus on those genres for our multilabel prediction task. Since many books have multiple genres, this highlights a multilabel rather than multiclass setting. The genres we use in this project are - Science Fiction, Fantasy, Mystery, Suspense, Crime Fiction, Thriller and Horror.



The summary statistics of the dataset before and after cleaning are below:

	Stage	Total_Records	NonMissing_Plots	NonMissing_Genres	Unique_Authors	Pub_Years_Available	Median_Plot_Length
1	Raw	16559	16559	16559	4715	6799	1550.0
2	Cleaned	6830	6830	6830	1735	2822	1785.5

Our objective to prepare the data is to transform the plot summaries into a vector representation that can be used in the machine learning algorithms. This is two-fold: we pre-process the data in a way that can be parsed into the embedding schema, and then yield an embedding for each summary.

Steps of text pre-processing:

1. Lowercase all text
2. Tokenize text into individual words
3. Remove punctuation and special characters
4. Remove stop words
5. Lemmatize

The image below shows us the plot summaries, before and after being pre-processed.

Plot	Clean_Plot	Parsed_Genres
Alex, a teenager living in near-future England, leads ...	alex teenager live nearfuture england lead gang night...	Science Fiction, Novella, Speculative fiction, Utopian a...
The novel posits that space around the Milky Way is ...	novel posit space around milky way divide concentric ...	Hard science fiction, Science Fiction, Speculative fictio...
Ged is a young boy on Gont, one of the larger island...	ged young boy gont one large island north archipelag...	Children's literature, Fantasy, Speculative fiction, Bild...
Living on Mars, Deckard is acting as a consultant to a...	live mar deckard act consultant movie crow film story...	Science Fiction, Speculative fiction
Beginning several months after the events in Blade R...	begin several month event blade runner deckard retir...	Science Fiction, Speculative fiction
Nine years after Emperor Paul Muad'dib walked into t...	nine year emperor paul muaddib walk desert blind ec...	Science Fiction, Speculative fiction, Children's literatur...
The situation is desperate for the Bene Gesserit as th...	situation desperate bene gesserit find target honor m...	Science Fiction, Speculative fiction, Children's literatur...
The novel is told in epistolary format, as a series of l...	novel tell epistolary format series letter diary entry sh...	Science Fiction, Speculative fiction, Horror, Invasion li...

The next step after this, is the process of 1 hot encoding the genres. This essentially creates a column for each genre, and if the book in that row has been tagged in that genre, the column contains a 1, else a 0.

Parsed_Genres	Science Fiction	Fantasy	Mystery	Suspense	Crime Fiction	Thriller	Horror
Science Fiction, Novella, Speculative fiction, Utopian a...	1	0	0	0	0	0	0
Hard science fiction, Science Fiction, Speculative fictio...	1	1	0	0	0	0	0
Children's literature, Fantasy, Speculative fiction, Bild...	0	1	0	0	0	0	0
Science Fiction, Speculative fiction	1	0	0	0	0	0	0
Science Fiction, Speculative fiction	1	0	0	0	0	0	0
Science Fiction, Speculative fiction, Children's literatur...	1	0	0	0	0	0	0
Science Fiction, Speculative fiction, Children's literatur...	1	0	0	0	0	0	0
Science Fiction, Speculative fiction, Horror, Invasion li...	1	1	1	0	0	0	1
Science Fiction, Speculative fiction, Children's literatur...	1	0	0	0	0	0	0
Children's literature, Absurdist fiction, Novella, Specul...	0	1	0	0	0	0	0
Science Fiction, Children's literature, Speculative fictio...	1	0	0	0	0	0	0
Fantasy, Fiction	0	1	0	0	0	0	0
Mystery, Detective fiction, Novel, Fiction, Suspense	0	0	1	1	0	0	0
Science Fiction, Speculative fiction, Children's literatur...	1	0	0	0	0	0	0
Science Fiction, Speculative fiction, Fiction	1	0	0	0	0	0	0
Science Fiction, Speculative fiction	1	0	0	0	0	0	0
Cyberpunk, Science Fiction, Speculative fiction, Dysto...	1	0	0	0	0	0	0
Science Fiction, Speculative fiction, Fiction, Dystopia	1	0	0	0	0	0	0
Science Fiction, Speculative fiction, Fiction, Dystopia	1	0	0	0	0	0	0
Science Fiction, Psychological novel, Speculative fictio...	1	0	0	0	0	0	0
Science Fiction, Speculative fiction, Dystopia	1	0	0	0	0	0	0
Science Fiction, Speculative fiction, Fiction, Novel	1	0	0	0	0	0	0

Feature Engineering and Supervised Genre Classification

In our methodology, we use 4 different embedding methods. These are:

- Bag-of-Words (BoW): this utilizes a sparse word frequency matrix
- TF-IDF (Term Frequency - Inverse Document Frequency):
    - TF (Term Frequency) = ratio of the number of target terms in the document to the total number of terms in the document
    - IDF (Inverse Document Frequency) = log of the ratio of the total number of documents to the number of documents in which the target term occurs.
- Word2Vec: this is trained on the corpus of books using CBoW
- GloVe: 50-dimensional pretrained embeddings

Each of these different embedding techniques have their advantages and disadvantages. We evaluate the performance of each embedding technique across each different predictive model.

Supervised Genre Classification

Each embedding technique was then paired with the following supervised learning algorithms, and the pair was trained on an 80-20 split of the data, with 80% being used to train the data and 20% being used to test. The supervised learning algorithms used are:

- Logistic Regression

- Support Vector Machines with both Linear and RBF Kernels
- Decision Trees
- Random Forest

In order to evaluate the performance of these model and embedding pairs, we used the following metrics:

- Precision
- Recall
- Hamming Loss
- F1 score

We define these metrics as follows:

Precision: The proportion of predicted positive labels that are actually correct.

$$Precision = \frac{True\ Positives}{(True\ Positives + False\ Positives)}$$

Recall: The proportion of actual positive labels that were correctly identified.

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

Hamming Loss: The fraction of incorrect labels to the total number of labels, averaged over all instances. In multilabel classification, it accounts for both false positives and false negatives.

$$Hamming\ Loss = \frac{(Number\ of\ incorrect\ labels)}{(Number\ of\ total\ labels \times Number\ of\ instances)}$$

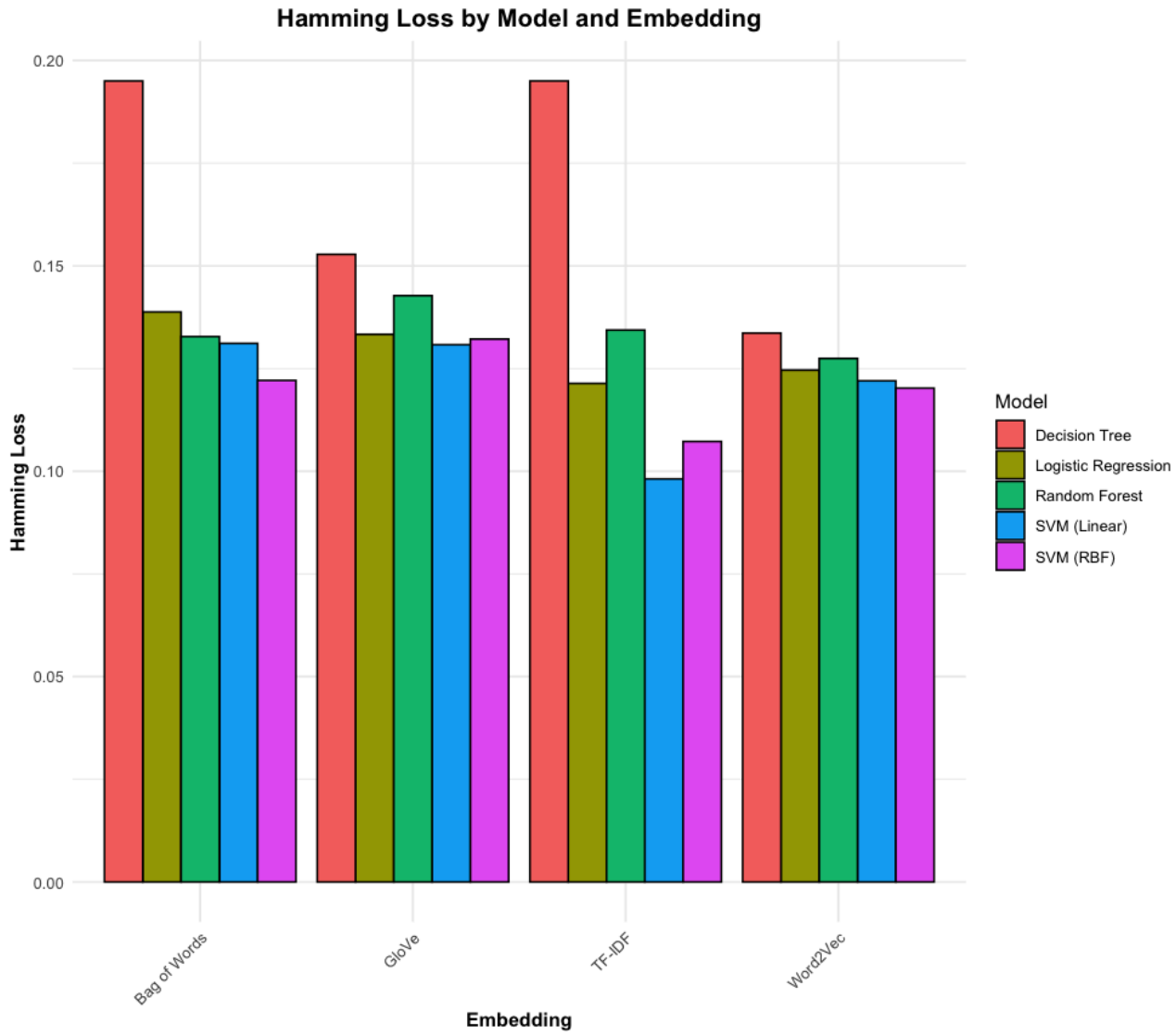
F1 Score: The harmonic mean of precision and recall. It balances the trade-off between the two metrics.

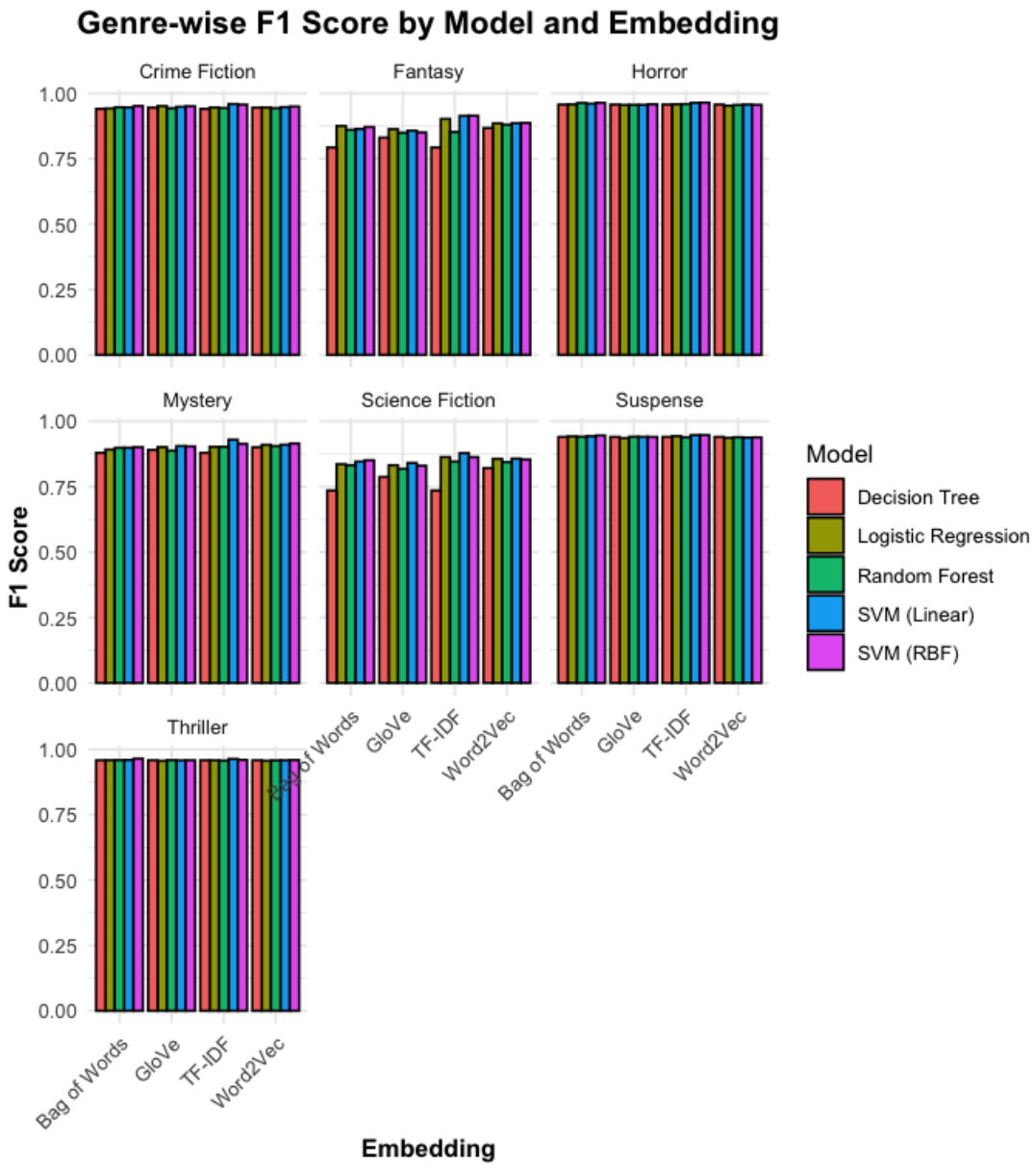
$$F1\ Score = \frac{2 \times (Precision \times Recall)}{(Precision + Recall)}$$

Key Observations

We can now analyze the performance of the different algorithm and embedding combinations. Our key take-aways are:

- We see through the graph below, that the combination of the TF-IDF embedding and the Support Vector Machine (Linear Kernel) algorithm had the lowest Hamming Loss.
- The worst performing model (highest Hamming Loss) across all embeddings was the Decision Tree. Decision Tree + TF-IDF was the worst combination.
- The best performing embedding was TF-IDF.

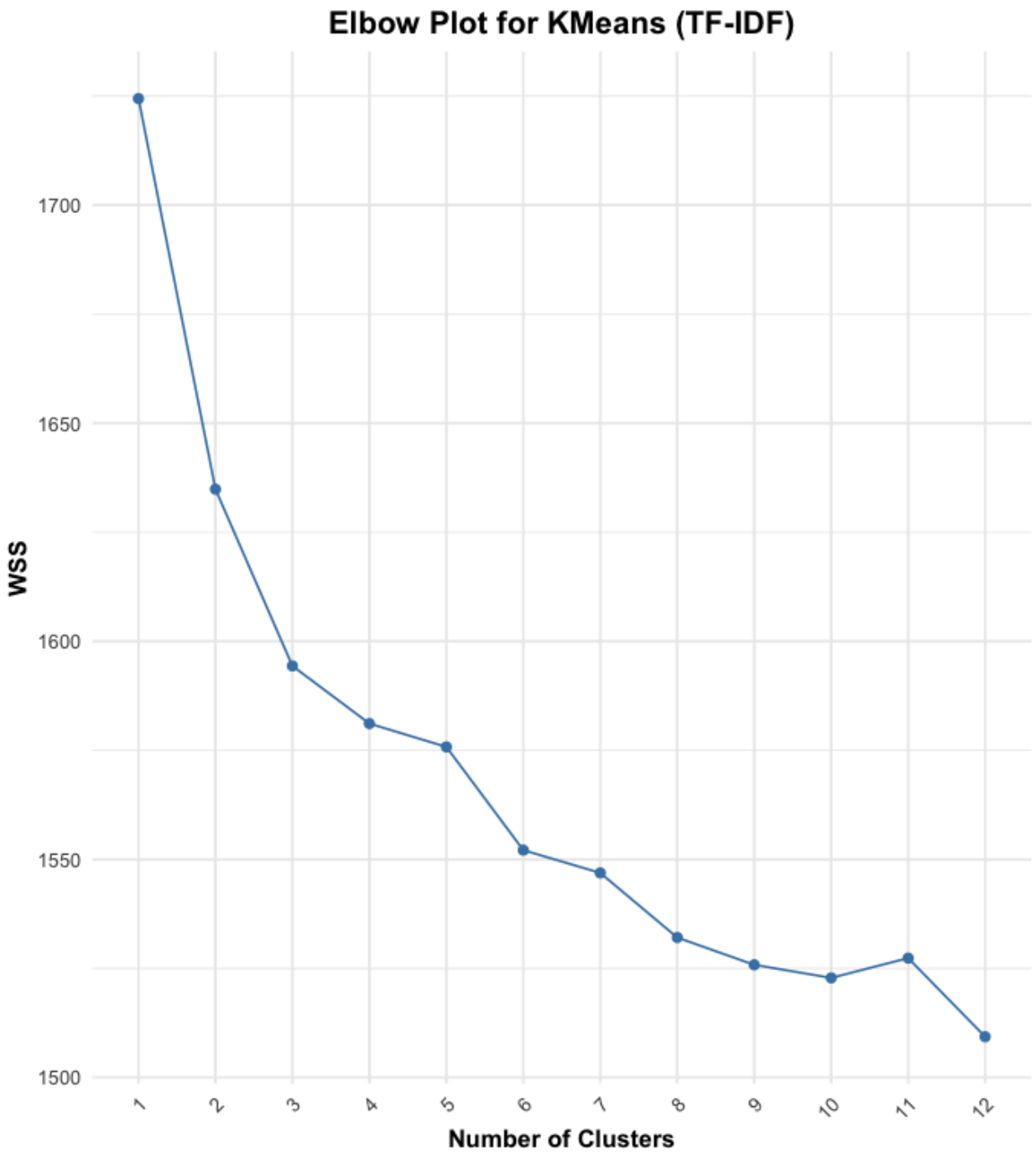

Hamming Loss by Model and Embedding
Hamming Loss
0.20
0.15
0.10
0.05
0.00
Bag of Words
GloVe
TF-IDF
Word2Vec
Embedding
Model
Decision Tree
Logistic Regression
Random Forest
SVM (Linear)
SVM (RBF)



Unsupervised Clustering and Genre Discovery

Moving on to the next step, we then used the TF-IDF embedding, and performed unsupervised k-means clustering on the embedding data.

In order to perform the clustering, we first iterated across several values for k (the number of clusters) and plotted the values of different k's versus the WSS (Within Sum of Squares). This helped us analyze the optimal value of k for the data. As we can see from the elbow plot below< the optimal number of clusters, k, could be interpreted as 6. Thus, we moved forward with our k-means clustering with k = 6.



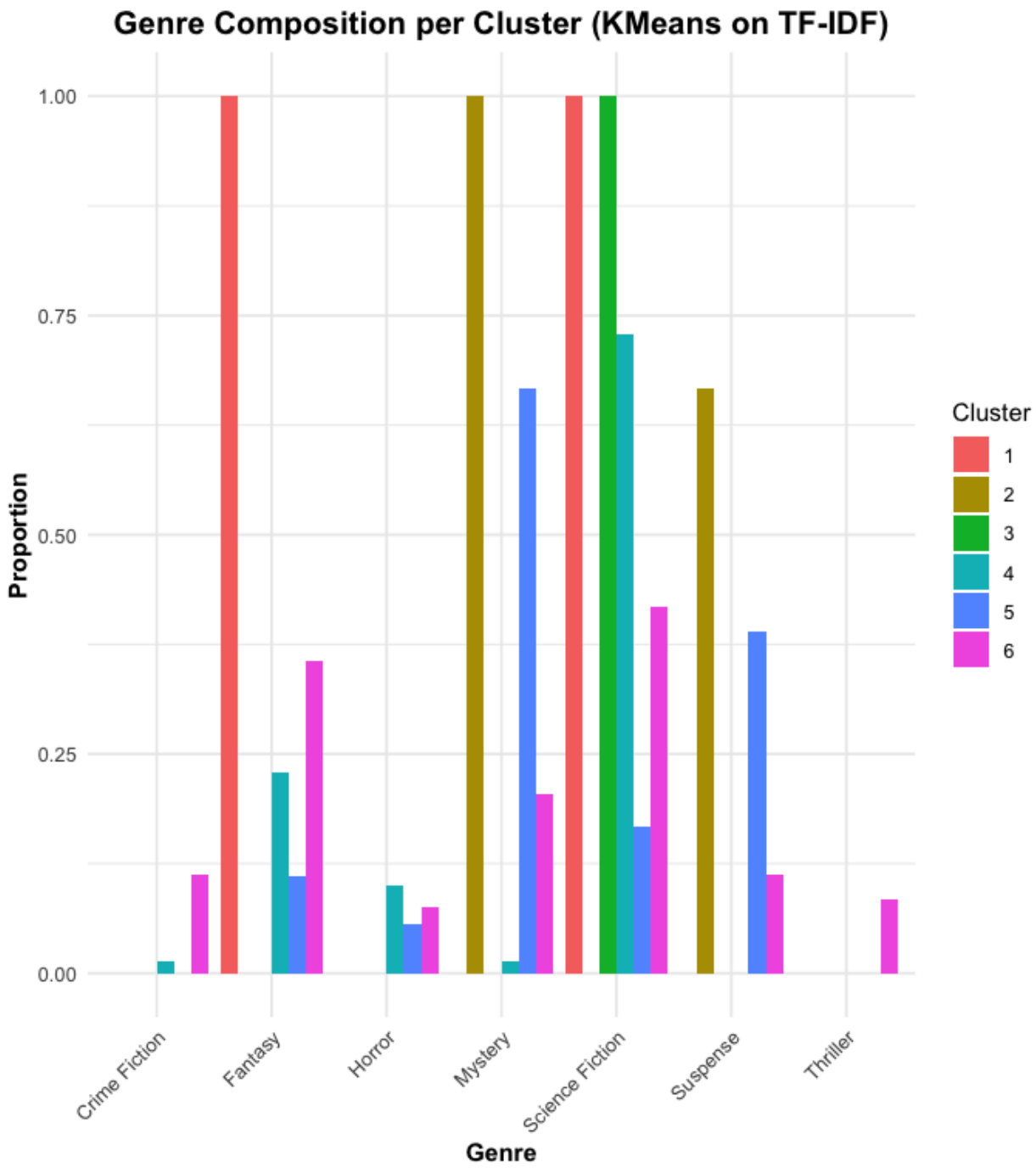
We then performed the clustering with 6 clusters, and our results for the clustering can be seen in the table below. Each column in this table represents one of the 7 genres used in our dataset, with each row representing one of the 6 clusters. Hence, each value represents the proportion of books tagged with that genre in that cluster, from a scale 0 to 1. If the value is 1, 100% of the books in that cluster are tagged with that genre, and if the value is 0, none of the books in that cluster are tagged with that genre.

Analyzing the clusters:

- We can see that cluster 1 consists of books tagged as Science Fiction and Fantasy exclusively. This indicates that the Fantasy and Science Fiction genres are heavily linked, and an entire cluster consists of just these 2 genres, and can be called a well-defined sub-genre. This combination is therefore 'pure', and the cluster seems reasonable given the fact that science fiction and fantasy are adjacent in the literary world.
- We see cluster 2 consists entirely of books tagged as Mystery, with 66.7% of them also being tagged as Suspense. This indicates that these 2 genres are linked as well, which seems reasonable. This sub-genre also seems to be well defined.
- Cluster 3 consists of only books tagged as Science Fiction, with no other genre present. Given that Science Fiction is the most prevalent genre in our dataset, this makes sense as it would also be the most well-defined.
- Cluster 4 appears to be a mix of largely Science Fiction and Mystery again, with small proportions of Mystery, Crime Fiction and Horror. This cluster appears blended, and the low proportions of other genres other than Science Fiction and Mystery seem to indicate that perhaps this cluster is somewhat redundant due to its similarity to Cluster 1.
- Cluster 5 is dominated by Mystery and Suspense, with small proportions of Science Fiction, Fantasy and Horror. This is another blended cluster, with similarity to Cluster 2.
- Cluster 6 contains books tagged with every genre, and not overwhelmingly dominated by any 1 genre. Due to the uniqueness of this fact, it makes sense that books containing all genres are in this cluster, as it would be rare to find books containing all 7 genres.

Through the clustering, it is important to note that the genres of Science Fiction, Fantasy, Mystery and Suspense are well classified, while the genres of Crime Fiction, Thriller and Horror are underclassified.

Cluster	Science Fiction	Fantasy	Mystery	Suspense	Crime Fiction	Thriller	Horror
1	1.000	1.000	0.000	0.000	0.000	0.000	0.000
2	0.000	0.000	1.000	0.667	0.000	0.000	0.000
3	1.000	0.000	0.000	0.000	0.000	0.000	0.000
4	0.729	0.229	0.014	0.000	0.014	0.000	0.100
5	0.167	0.111	0.667	0.389	0.000	0.000	0.056
6	0.418	0.355	0.205	0.112	0.112	0.084	0.075



Overall, the existence of certain blended clusters seems to indicate that some sub-genres or combinations of genres are not as well defined as they would seem. We can also analyze the performance of this unsupervised k-means clustering, through the Precision, Recall and F1 Score, which can be seen in the table below. It also contains the metrics for the SVM (Linear Kernel) + TF-IDF model, to highlight the performance differences between the 2. The stark

performance differences between the 2 techniques also further justifies our chosen path of supervised learning. SVM (linear kernel) + TF-IDF outperformed the clustering in every metric.

Model	Precision	Recall	F1 Score
KMeans (TF-IDF)	0.237	0.222	0.229
SVM Linear (TF-IDF)	0.692	0.741	0.716

We also attempted a case study of 10 books, randomly picked from the corpus and compared their true genre labels from the original dataset, to their genres assigned via clustering. We can see that the cluster genres seem to align well with the true genres, however, seem to miss out on the depth and variety captured by the true genres.

Book ID	Cluster	True Genres	Cluster Genres
1839	1	Science Fiction, Fantasy	Science Fiction, Fantasy
6062	2	Mystery	Mystery, Suspense
6063	2	Mystery, Suspense	Mystery, Suspense
5302	3	Science Fiction	Science Fiction, Fantasy
4361	4	Science Fiction	Science Fiction, Fantasy
6574	4	Science Fiction	Science Fiction, Fantasy
4550	5	Fantasy, Mystery, Suspense	Mystery, Suspense
714	5	Mystery, Suspense	Mystery, Suspense
163	6	Mystery	Science Fiction, Fantasy
1905	6	Science Fiction	Science Fiction, Fantasy

Analysis of Methodology and Robustness

We explored two different methodologies in our project: Supervised and Unsupervised learning. The supervised learning method was through a combination of different embeddings and

algorithms, while our unsupervised learning was through TF-IDF embedding and k-means clustering.

We see that the clustering did help us discover an underlying structure to the genres, through the existence of clusters indicating distinct sub-genres, such as cluster 1, which showed us a pure Science Fiction + Fantasy sub-genre. However, the overall performance of the prediction via clustering was poor, especially in comparison to the best supervised learning model of SVM (linear kernel) + TF-IDF. This can be explained in part, due to the fact that this is a multilabel, in which clustering may not perform as well. Thus, this helps to justify our use of supervised learning, due to the vast improvement in performance.

We also utilized cross-validation in our methodology, splitting the data randomly in an 80-20 proportion of training versus test. This enhanced the robustness of our methodology, verifying the legitimacy of our results, and also verifying that our methodology would work across a different dataset as well.

If we consider the use of unsupervised learning through k-means as our baseline, which is a reasonable approach in and of itself, we can justify our use of supervised learning through a drastic performance improvement. Compared to other works, such as - ‘Book Genre Classification Based on Titles with Comparative Machine Learning Algorithms - Ozsarfati, et al. (2019)’, we also see a performance improvement over existing machine learning techniques.

Limitations and Improvements

The project, while thorough, also had some limitations that can be improved upon in further work in this area. The nature of genre classification leads to the issue of class imbalance, i.e, TN + FN >> TP + TN. This means that due to the nature of far more negatives than positives, we can fix this issue by increasing the weight of a positive value.

We also use GloVe embeddings in our supervised learning methodology, which are pre-trained embeddings on Wiki data. Thus, as some of the data in the original dataset is also derived from Wiki data, this may lead to data leakage. However, the magnitude of the GloVe training data is too large for our test dataset to have any significant influence on the GloVe embedding. The solution to this problem can be to use a different GloVe embedding, which is not trained on Wiki data. However, with this, the issue of domain mismatch arises.

Genre may also depend on the tone, structure and audience, not just word choice. So further analysis could include further features other than just text embedding, and this may also help with under-classified genres of Horror, Thriller and Crime Fiction.

Conclusion

This project reconsiders the problem of genre classification through the lens of machine learning. We see through our two methodologies - supervised and unsupervised learning - that there is the possibility of using machine learning to tag books as one or more genres based on their summaries.

In supervised learning, through the use of several embedding and model combinations, we were able to see that the TF-IDF embedding combined with Support Vector Machines (Linear Kernel), performed well, with reasonably high F-1 score. This highlights the potential of using machine learning for genre classification by content managers and librarians working at University libraries, bookstores, online forums and online bookstores (Amazon, GoodReads). Utilizing supervised learning techniques can help save time, improve efficiency, as well as improve consistency in genre classification.

In unsupervised learning, although the k-means clustering algorithm performed poorer than the SVM (linear kernel) + TF-IDF, we saw that the clustering helped us prove the existence of underlying structure and patterns in the book summaries and genre classification. We were able to delineate the existence of well-defined sub-genres, as well as fuzzy sub-genres that are not as well-defined. This also served as a baseline for us to prove the better performance of the supervised learning techniques.