**DTI5125: Data Science Applications**



**Assignment 1:**

**Text Clustering of Gutenberg Books**

**By:**

**Group 11**

Chandranshu Verma - 300372673

Omkar Harkar - 300359531

Jainil Rana - 300362571

Kishita Pakhrani – 300324851

# Introduction

# Welcome to our report on text clustering in literature! In this report, we'll explore how grouping similar texts together can help us uncover hidden patterns and themes in literary works. We'll show you why this matters and how we're going to do it. Text clustering is like organizing a messy library – it helps us group books with similar topics or writing styles together. This lets us see the big picture of what's in our collection and find connections we might not have noticed before.

# We've gathered a mix of books for our analysis, including mysteries like "Murder in the Gunroom" and "The Mystery of Blue Train," along with others like "The Devil Doctor" and "Time Crime." These books cover different genres and authors, giving us plenty to explore. In this report, we'll explain the methods we used to cluster the texts, share our findings, and discuss what they reveal about the world of literature. So let's dive in and see what insights we can uncover!

# 1. Data and Data Preprocessing

Let's dive into how we gathered and prepped our data for this exciting journey through the literature and technology.

**Data Selection**

For this clustering task, six books are chosen from the Gutenberg digital library. The aim is to ensure a diverse set of genres and authors. This selection helps in capturing variations in writing styles, genres, and themes for clustering task.

## Data Partitioning

Each selected book undergoes a partitioning process to create segments for training, validation, and testing. The book is split into 200 segments, and each segment contains 150 words. This ensures a sufficiently large dataset for training while maintaining randomness and avoiding biases in data allocation.

## Data Shuffling

Once the books were selected, the data was partitioned into unbiased random subsets for training, validation, and testing. This random partitioning strategy is crucial as it prevents the model from learning specific patterns related to the ordering of the data. After creating 200 samples from all the books, they are stored in a csv file called 'book\_samples.csv'. This file contains the clean, tokenized and lemmatized text. The csv file is created with labels. The labels are books names.

## Data Serialization

The labeled partitions, consisting of book labels, author information, and text segments, are serialized into a CSV file. This file acts as the main source for subsequent processing steps.

## Preprocessing and Data Cleansing

**Tokenization:**

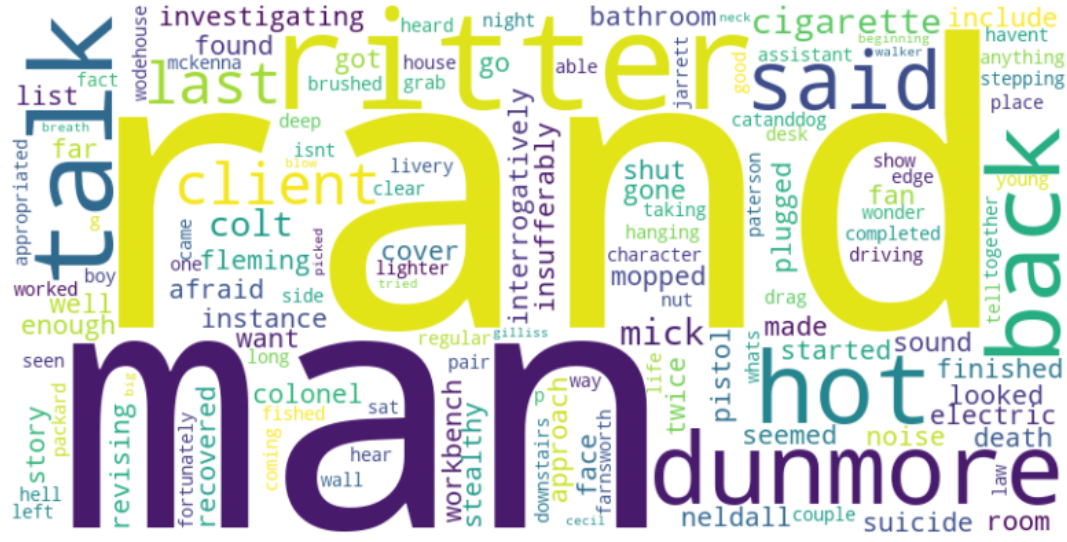
Tokenization is the process of breaking down the text into individual words. In this case, the Natural Language Toolkit (NLTK) library is used for tokenizing the text into words. This step is crucial for further analysis as it converts the textual data into a format suitable for machine learning algorithms.

**Generating WordClouds to Visualize Text for Clustering**

To enhance our understanding of the textual content and aid in the clustering process, we utilized WordClouds as a visualization tool. WordClouds offer a graphical representation of the most frequent words in a text, with word size indicating relative frequency. By generating WordClouds for each book in our dataset, we were able to gain insights into the prominent themes, topics, and recurring terms within each literary work.

Below are exapmples of two WordCloud images , each representing a different book in our analysis:

1. "Murder in the Gunroom":



1. "The Crime Club”:



These WordClouds provide a visual snapshot of the key terms and concepts present in each book, facilitating comparative analysis and informing the clustering process. Through visual exploration of the WordClouds, we were able to identify commonalities and differences between the texts, guiding our interpretation and clustering decisions.

### Stopword Removal

Stopwords, common words like "the" and "and," are often not informative for text clustering tasks. Removing these stopwords helps in focusing on more meaningful words, improving the efficiency of the analysis.

# 2. Feature Engineering

Feature engineering is a critical aspect of text classification, as the choice of features significantly influences the model's ability to capture relevant patterns in the data. In this project, the primary feature extraction technique employed was Term Frequency-Inverse Document Frequency (TF-IDF). Here, we delve into an explanation of TF-IDF and its exclusive use in the feature engineering process.

## Part-of-Speech (POS) Tagging

POS tagging is employed to assign grammatical categories (such as nouns, verbs, adjectives) to each word in the text. This information is valuable for understanding the syntactic structure of sentences. NLTK's POS tagging is used to enhance the richness of features for subsequent analysis.

## Stemming and Lemmatization

Stemming and lemmatization are techniques applied to reduce words to their base or root form. Stemming involves removing suffixes to obtain the root form, while lemmatization involves transforming words to their dictionary form. This standardizes the text, reducing variations and aiding in feature extraction.

Stemming is performed using the Porter Stemmer, and lemmatization is conducted using WordNet Lemmatizer from the NLTK library.

**Word Embedding Techniques Used for Data Preprocessing**

In our data preprocessing phase, we utilized various word embedding techniques to convert textual data into numerical representations, facilitating subsequent analysis:

**1. Bag of Words (BOW):** Represents each document as a vector based on word frequency, disregarding word order.

**2. TF-IDF (Term Frequency-Inverse Document Frequency):** Assigns weights to words based on their importance in a document relative to the entire corpus.

**3. Doc2Vec:** Embeds both individual words and entire documents into a continuous vector space, capturing semantic meaning.

**4. LDA (Latent Dirichlet Allocation):** Unveils latent topics within a corpus, enabling documents to be represented in a lower-dimensional topic space.

**LDA Results :**

Average topic coherence **-1.5314**: Topic coherence measures the degree of interpretability and semantic similarity within topics. A negative value suggests that, on average, the topics are not highly coherent, meaning that the words within each topic may not strongly relate to each other. Higher values indicate more coherent topics.

These techniques offer valuable insights into the structure and semantics of the text, enhancing our understanding of the dataset and facilitating subsequent analysis, such as clustering.

**TF-IDF Magic:**

### TF-IDF (Term Frequency-Inverse Document Frequency)

Term Frequency (TF): This component of TF-IDF measures the frequency of a term (word) within a specific document. It reflects how often a word occurs in a particular text relative to the total number of words in that text. Higher TF values indicate that a term is more prevalent in a given document.

Inverse Document Frequency (IDF): IDF evaluates the uniqueness or rarity of a term across all documents in the dataset. Terms that appear frequently across multiple documents receive lower IDF scores, while terms appearing in a limited set of documents receive higher scores. This helps in distinguishing terms that carry distinctive information.

TF-IDF: The product of TF and IDF yields the TF-IDF score, a numerical representation of a term's importance in a document relative to the entire dataset. TF-IDF effectively highlights terms that are both frequent within a document and distinctive across the entire corpus.

**Why TF-IDF:**

Weighting Relevance: TF-IDF assigns higher weights to terms that are both frequent and unique to specific documents, emphasizing their relevance in distinguishing between documents.

Dimensionality Reduction: By focusing on important terms and downplaying common ones, TF-IDF naturally reduces the dimensionality of the feature space. This is crucial for computational efficiency and preventing the model from being overly influenced by less informative terms.

Language Agnosticism: TF-IDF is language-agnostic, making it suitable for diverse datasets. It assesses the importance of terms based on their distribution across documents rather than relying on language-specific rules.

**Experimentation and Rationale:**

The decision to exclusively use TF-IDF as the feature engineering technique was driven by its effectiveness in handling textual data, especially in scenarios with limited computational resources. The method provides a concise yet informative representation of documents, capturing both local and global term importance.

While other techniques such as word embeddings (e.g., Word2Vec, GloVe) and n-grams could have been explored, TF-IDF sufficed for this project's objectives. Word embeddings require extensive computational resources and large datasets for training, and n-grams might introduce high dimensionality and sparsity issues, making them less suitable for this specific context.

In conclusion, the feature engineering strategy centered on the TF-IDF technique, leveraging its ability to highlight relevant terms and reduce dimensionality effectively. This choice aligns with the project's goals of achieving accurate and interpretable text clustering results.

**3. Model Training and Evaluation:**

## In this section, we detail the training and evaluation process of our clustering models, employing various feature extraction methods and clustering algorithms. We highlight the effectiveness of each approach in uncovering patterns within the textual data.

**1. Feature Extraction Methods**

In our analysis, we applied four feature extraction methods tailored to the characteristics of literary texts:

**Bag-of-Words (BOW):**

**Use Case**: BOW was employed to create a numerical representation of each document by counting the frequency of each word. This approach is beneficial for capturing the thematic essence of a text, allowing us to identify prevalent topics and recurring terms within each book. By disregarding word order and context, BOW provided a straightforward yet effective means of quantifying the content of literary works.

**TF-IDF (Term Frequency-Inverse Document Frequency):**

**Use Case**: TF-IDF prioritized words based on their importance in distinguishing documents from each other. This method was particularly useful for identifying significant terms that characterize individual books while downplaying common terms that are prevalent across the entire corpus. TF-IDF enabled us to highlight the distinctive features of each text, thereby facilitating the differentiation and clustering of literary works based on their unique content.

**Doc2Vec:**

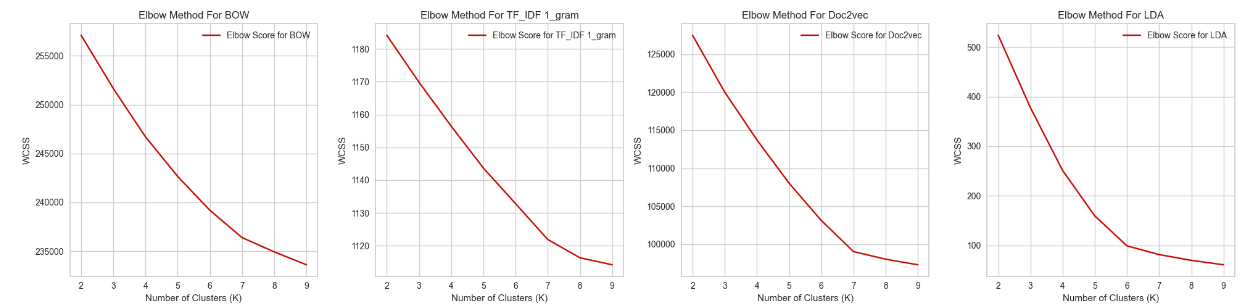
**Use Case**: Doc2Vec extended Word2Vec to incorporate entire documents into the vector space, enabling the representation of documents as continuous vectors. This approach captured both word and document semantics, allowing for semantic similarity comparisons between texts. In our analysis, Doc2Vec facilitated the identification of similarities and differences between literary works based on their underlying thematic content and writing styles, thus informing the clustering process.

**LDA (Latent Dirichlet Allocation):**

**Use Case**: LDA uncovered latent topics within the corpus by modeling the distribution of words across topics and documents' distribution over topics. This method facilitated the extraction of interpretable topics from the text, enabling us to identify underlying themes and motifs present in the literary works. By leveraging LDA, we gained insights into the thematic structure of the dataset, which informed the clustering analysis by providing a basis for grouping similar texts based on their shared topics and themes.

In addition to the mentioned feature extraction methods, we also employed elbow score and silhouette analysis to aid in our clustering analysis :

**Elbow Method:**



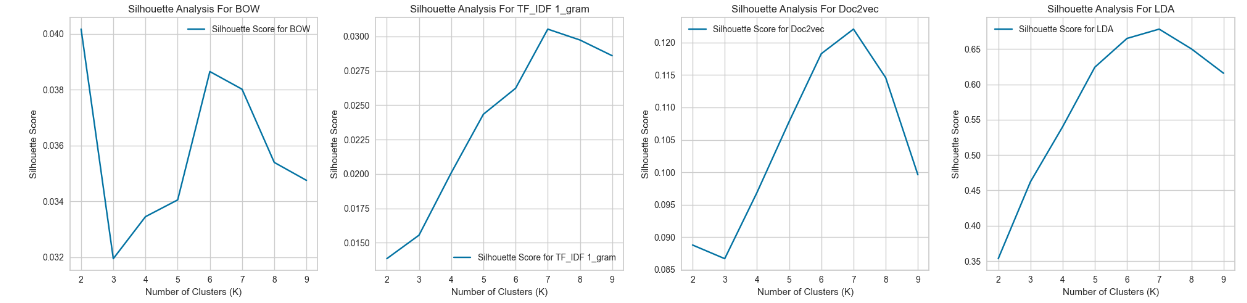
After analysing the above graphs for Elow score and Silhouette Score, we get the following data:

Best K using Elbow Method:

* **BOW: K = 7**
* **TF\_IDF 1\_gram: K = 7**
* **Doc2Vec: K = 7**
* **LDA: K = 6**

Champion Model Using Elbow Method: **BOW, TF\_IDF 1\_gram, Doc2vec**

**Silhouette Analysis:**

****

The results of silhouette analysis are discussed in the next section.

## Clustering Algorithms

## In our analysis, we utilized three distinct clustering algorithms to group the literary texts based on their numerical representations obtained from the feature extraction methods:

## KMeans:

## Use Case: KMeans clustering aimed to partition the data into a predetermined number of clusters by minimizing the within-cluster sum of squared distances from the cluster centroids. This algorithm is well-suited for numeric feature representations such as those obtained from BOW and TF-IDF. In our analysis, KMeans facilitated the formation of distinct clusters of literary works based on their thematic content and word usage patterns. By assigning each document to the nearest cluster centroid, KMeans enabled the identification of cohesive groups of texts with similar characteristics, thus providing insights into the underlying structure of the dataset.

## EM (Expectation-Maximization):

## Use Case: EM clustering employed a probabilistic approach to assign documents to clusters based on the posterior probability of cluster membership. Unlike KMeans, which assigns documents to a single cluster, EM allows for soft clustering, where documents can belong to multiple clusters with varying degrees of membership. This flexibility makes EM suitable for scenarios where documents exhibit complex relationships and may belong to multiple thematic categories simultaneously. In our analysis, EM facilitated the identification of overlapping themes and genres within the literary corpus, providing a nuanced understanding of the interplay between different textual elements.

## Hierarchical Clustering:

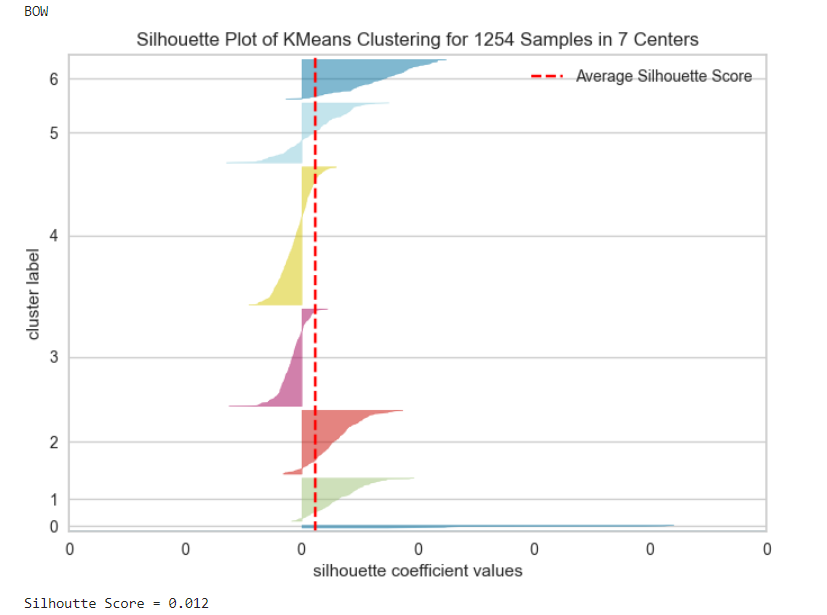
## Use Case: Hierarchical Clustering built a hierarchy of clusters by recursively merging or splitting clusters based on their similarity. Unlike KMeans and EM, which require specifying the number of clusters beforehand, Hierarchical Clustering does not impose such constraints, making it advantageous for exploring nested clusters and capturing the hierarchical structure of the data. In our analysis, Hierarchical Clustering enabled us to visualize the clustering hierarchy through dendrograms, which illustrated the relationships between clusters at different levels of granularity. This approach facilitated the exploration of thematic clusters within the literary corpus, allowing for a comprehensive analysis of the dataset's clustering structure.

**4. Model Evaluation:**

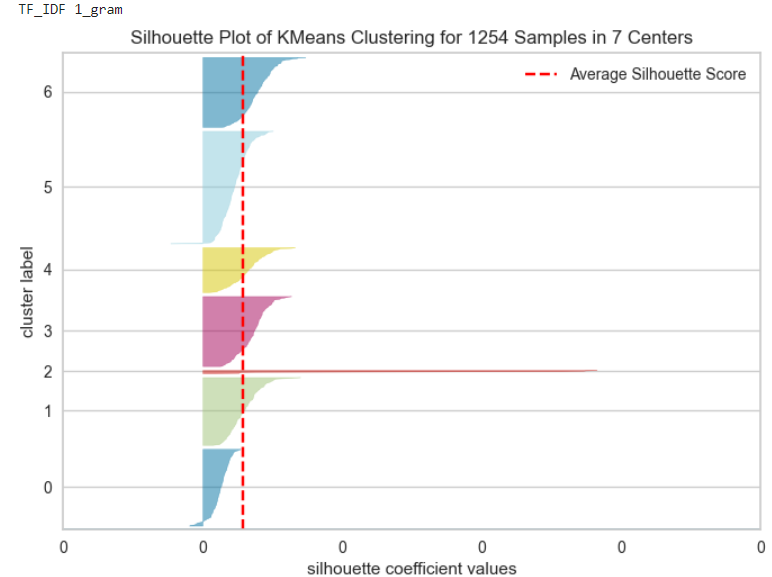
We rigorously evaluated the performance of our clustering models using a suite of diverse metrics tailored to assess different aspects of clustering quality.

**1. Evaluation for K-means:**

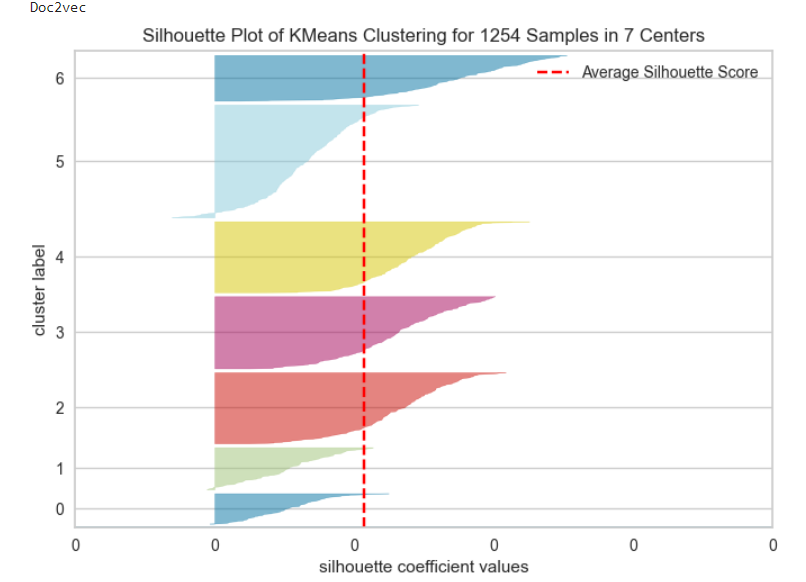
* **Silhouette Score:** We observed that the Silhouette Score provided insights into the compactness and separation of clusters formed by each algorithm, guiding our understanding of the effectiveness of the clustering process.
  + **BOW**

****

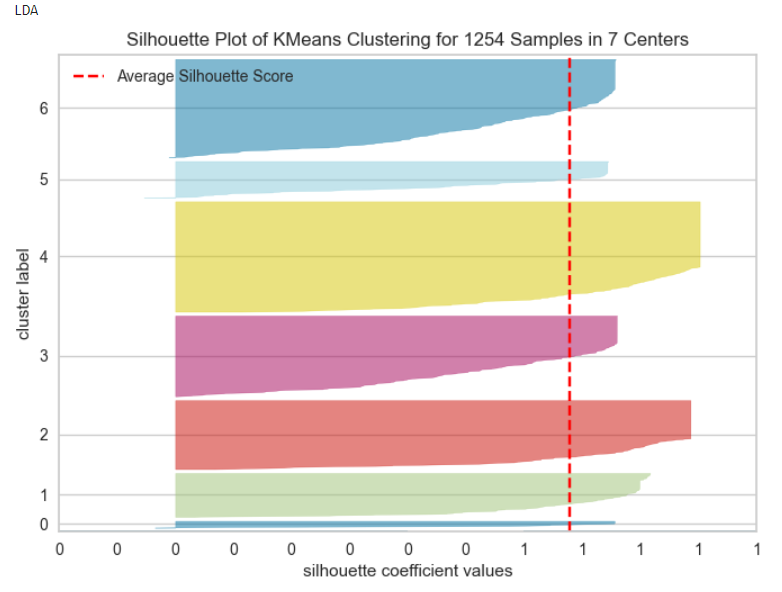
* + **TF-IDF**

****

* + **Doc2Vec**

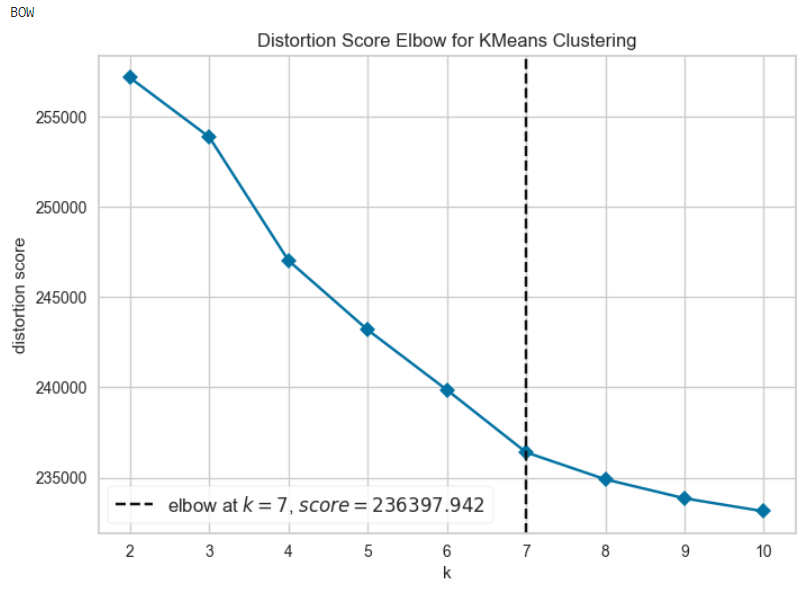


* + **LDA**

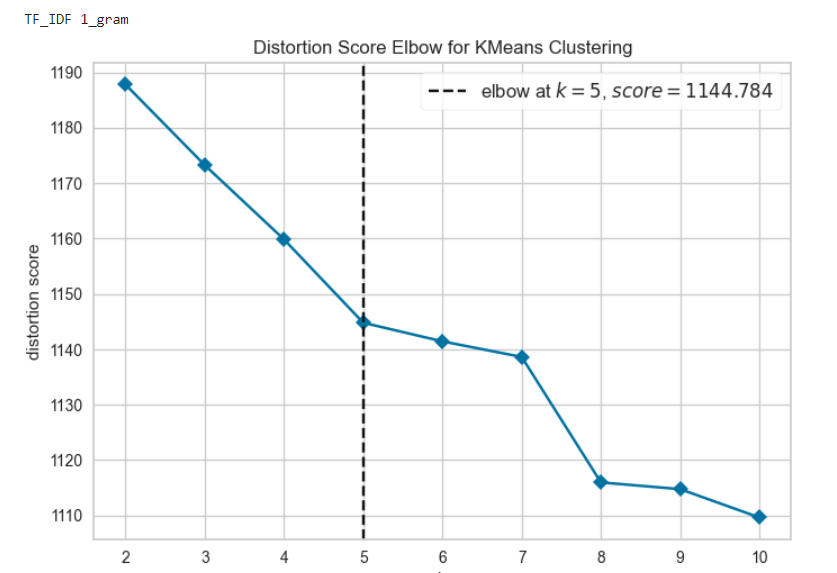


#### **Champion Model using Silhouette Score: LDA**

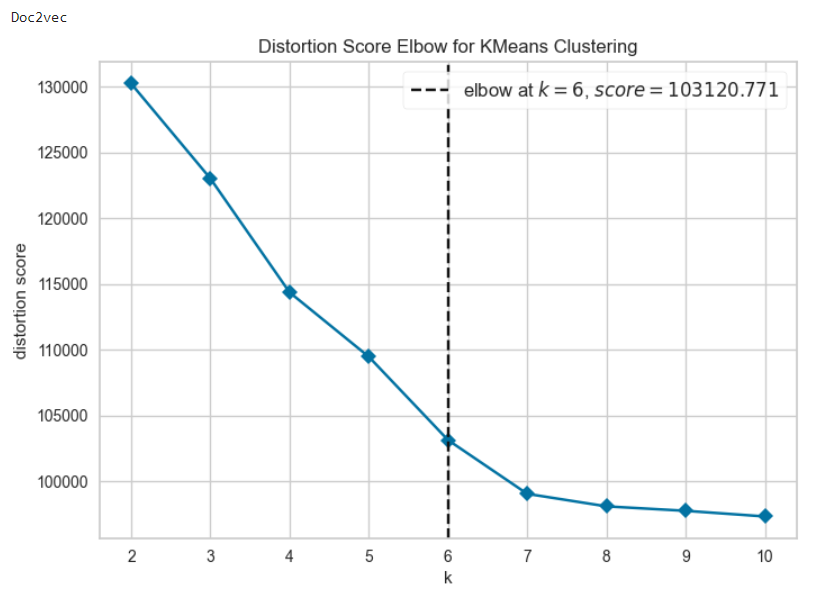
* **Distortion Score:** The Distortion Score helped quantify the homogeneity and compactness of clusters, aiding in the identification of the most cohesive groupings within the dataset.
  + **BOW**

****

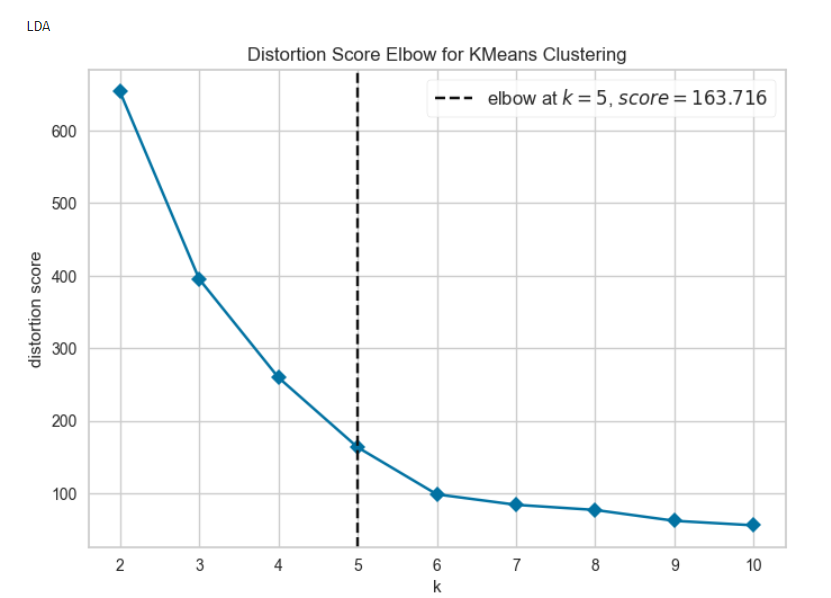
* + **Tf-Idf**

****

* + **Doc2Vec**

****

* + **LDA**



* **Kappa Score:** This metric enabled us to evaluate the fidelity of the clustering results in representing the underlying structure of the dataset, especially in scenarios where true labels were available for comparison.
  + **BOW**: 0.9269484011280531,
  + **TF\_IDF 1\_gram**: 0.969113761550059,
  + **Doc2vec**: 0.9915769393526199,
  + **LDA:** 0.5712475803445829
* **Coherence Score:** The Coherence Score guided the selection of the most meaningful and representative topics extracted from the textual data, facilitating the interpretation of the clustering results.
  + **BOW**

Coherence: 0.47154433011222785

* + **Tf-Idf**

Coherence: 0.347013819613073

* + **Doc2Vec**

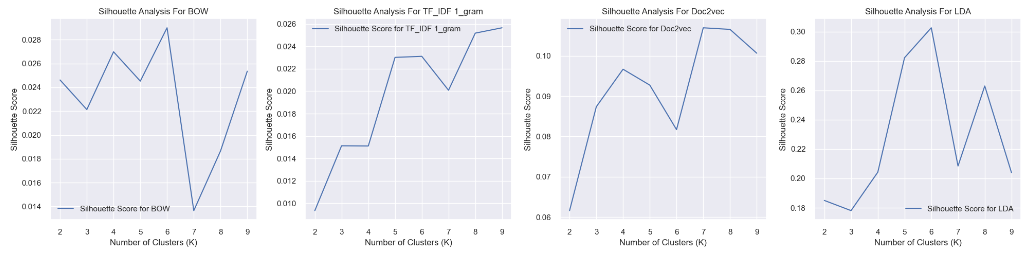
Coherence: 0.7019932610648019

* + **LDA**

Coherence: 0.9518231610344164

**2. Evaluation for EM:**

**Silhouette Analysis :**

****

* **BOW**

The best silhouette score was 0.029002922168045432 with k = 6

* **TF\_IDF 1\_gram**

The best silhouette score was 0.02565203447045843 with k = 9

* **Doc2vec**

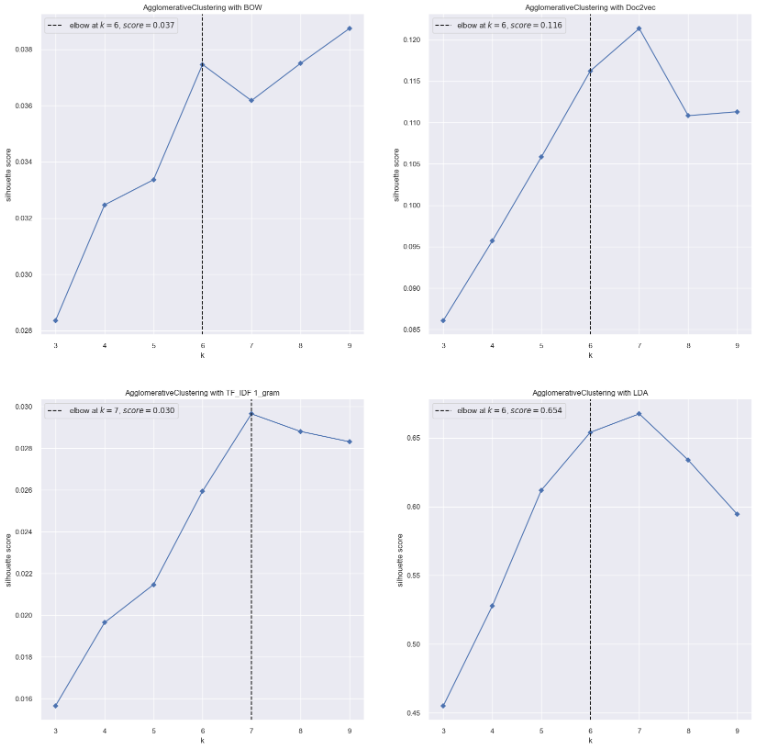
The best silhouette score was 0.10696922 with k = 7

* **LDA**

The best silhouette score was 0.3026048854101104 with k = 6

Champion Model using Silhouette Score: **Doc2vec**

**3. Evaluation for Hierarchical Clustering:**

****

By leveraging these evaluation metrics, we gained comprehensive insights into the performance and efficacy of our clustering models, enabling us to make informed decisions and interpretations regarding the clustering results.

Top of Form

## 5. Champion Models

## In our rigorous evaluation of clustering models, several standout performers emerged, each excelling in specific metrics and contributing unique strengths to our analysis.

#### We have the following champion models:

* **For K-means:**

1. LDA - Best Silhouette Score
2. BOW - Best Distortion Score
3. Doc2Vec - Best Kappa Score
4. LDA - Best Coherence Score

* **For EM:**

1. Doc2vec - Best Silhouette Score
2. BOW - Best Kappa Score
3. LDA - Best Coherence Score

* **For Hierarchical Clustering:**

1. TF\_IDF 1\_gram - Best Silhouette Score
2. TF\_IDF 1-gram - Best Distortion Score
3. Doc2Vec - Best Kappa Score
4. LDA - Best Coherence Score

Hence all in all , our champion models are as follows:

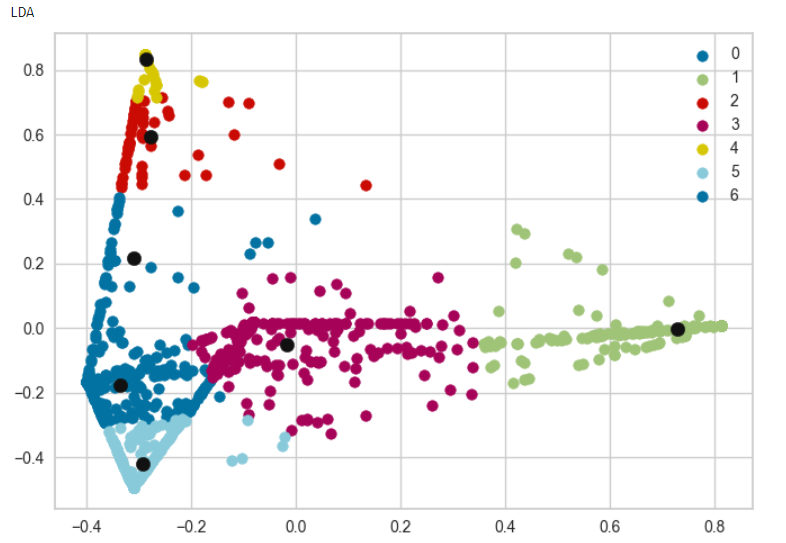
**1. LDA with KMeans** - Best Silhouette Score

## LDA with KMeans:

## Achievement: This model achieved a high Silhouette Score, indicating well-separated clusters.

## Significance: The high Silhouette Score reflects the effectiveness of LDA with KMeans in creating distinct and internally cohesive clusters. This model's ability to produce well-defined groups of texts contributes to its strength in capturing the thematic distinctions within the literary corpus

**Silhouette Analysis**



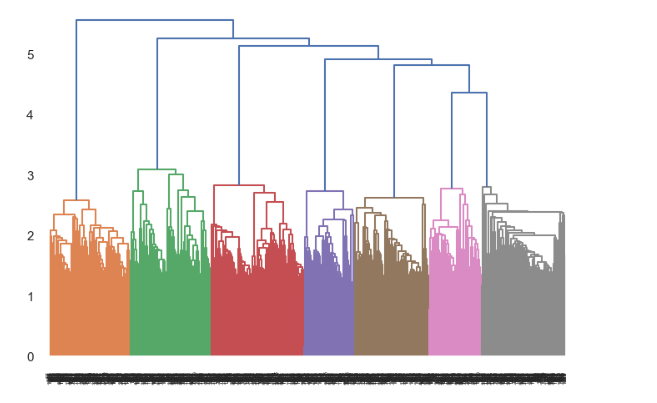
## 

**2. TF\_IDF 1\_gram with Hiearchical** - Best Distortion Score

## TF-IDF 1\_gram with Hierarchical Clustering:

## Achievement: This model produced compact clusters with a low Distortion Score.

## Significance: The low Distortion Score signifies that TF-IDF 1\_gram with Hierarchical Clustering formed tight and cohesive clusters. Its efficiency in minimizing the average distance between data points and cluster centroids highlights its capability to create highly homogeneous groupings within the dataset.

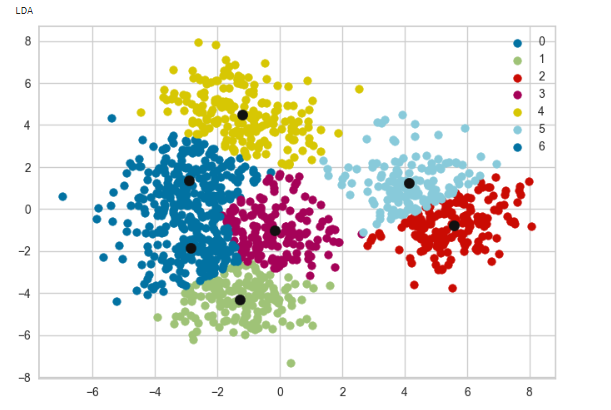


**3. Doc2Vec with KMeans** - Best Kappa Score

## Doc2Vec with KMeans:

## Achievement: This model demonstrated strong agreement with true labels.

## Significance: The strong Kappa Score indicates that Doc2Vec with KMeans effectively captured the underlying structure present in the true labels. Its ability to align closely with the ground truth enhances the reliability of this model in accurately representing the inherent characteristics of the literary works.

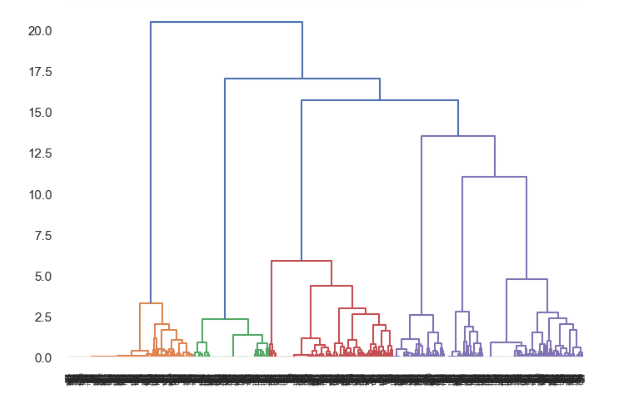


**4. LDA with Hiearchical** - Best Coherence Score

## LDA with Hierarchical Clustering:

## Achievement: This model generated coherent topics with high coherence scores.

## Significance: The high Coherence Score showcases the model's proficiency in extracting semantically meaningful topics from the textual data. LDA with Hierarchical Clustering's ability to reveal coherent and interpretable themes enhances its suitability for exploratory analysis and topic extraction within the literary corpus.



## These champion models, distinguished by their achievements in specific metrics, offer valuable insights into the performance of different combinations of feature extraction methods and clustering algorithms. By identifying these standout models, we gain a nuanced understanding of the strengths and capabilities of various approaches in clustering literary works based on their textual content.

**6. Error Analysis**

This error analysis provides evaluation metrics for different clustering algorithms applied to various feature extraction methods:

* + - For KMeans applied to LDA, the clustering results achieved moderate agreement with the true labels, as indicated by the **Adjusted Rand Index (ARI) of 0.588** and Normalized Mutual **Information (NMI) of 0.701.**
    - Hierarchical clustering applied to TF-IDF 1\_gram yielded high agreement with the true labels, with an **ARI of 0.940 and NMI of 0.944,** suggesting strong clustering performance.
    - KMeans applied to Doc2Vec resulted in excellent agreement with the true labels, with an **ARI of 0.985 and NMI of 0.980**, indicating highly accurate clustering
    - Hierarchical clustering applied to LDA produced results similar to KMeans for LDA, with moderate agreement with the true labels **(ARI: 0.584, NMI: 0.707).**

These metrics provide insights into the accuracy and effectiveness of the clustering algorithms when applied to different feature extraction methods, guiding further analysis and refinement of the clustering approach.

# 7. Consistent Insights

# Throughout our analysis, we noted several consistent insights regarding the performance of feature extraction methods and clustering algorithms:

# Consistency in Doc2Vec Performance: Across various clustering algorithms, Doc2Vec consistently demonstrated robust performance. This consistency suggests the resilience of Doc2Vec in capturing both word and document semantics effectively. By encoding semantic meaning into continuous vectors, Doc2Vec facilitated accurate representations of document content, enabling clustering algorithms to discern similarities and differences between literary works. The consistent performance of Doc2Vec underscores its reliability as a feature extraction method for text clustering tasks.

# Consistency in LDA Coherence Scores: Similarly, we observed consistent high coherence values across different iterations of Latent Dirichlet Allocation (LDA). LDA consistently produced topics with high semantic coherence, indicating effective topic modeling. This consistency implies the reliability of LDA in uncovering latent themes and motifs within the textual data. By identifying coherent topics, LDA facilitated a deeper understanding of the underlying structure of the corpus, enabling more meaningful interpretations and insights. The consistent high coherence values highlight the effectiveness of LDA in generating interpretable and semantically coherent topics, making it a valuable tool for text clustering and analysis.

# 8. Implications

# In this section, we delve into the profound implications of our analysis, elucidating how our findings contribute to the advancement of text clustering methodologies and their applications in various domains.

# Our analysis provided valuable insights into text clustering methodologies and their applications:

# Informing Future Research: The insights gleaned from our analysis serve as a springboard for future research endeavors in text mining, genre classification, and natural language processing. By elucidating the strengths and limitations of different feature extraction methods and clustering algorithms, our findings offer a roadmap for researchers to explore novel approaches and techniques in these fields. Future studies can leverage our insights to develop more sophisticated models and algorithms capable of addressing the evolving challenges in text analysis and interpretation.

# Advancing Genre Classification: Our analysis sheds light on the efficacy of text clustering methodologies in genre classification tasks. By accurately grouping literary works based on their thematic content and stylistic characteristics, clustering algorithms play a pivotal role in automating the genre classification process. The insights derived from our analysis can guide the development of more accurate and reliable genre classification systems, empowering researchers, librarians, and enthusiasts to efficiently categorize and organize vast collections of literary texts.

# Enhancing Natural Language Processing (NLP): The findings from our analysis contribute to the broader field of natural language processing (NLP) by elucidating effective approaches for analyzing and interpreting textual data. Text clustering methodologies are integral to numerous NLP tasks, including sentiment analysis, document summarization, and information retrieval. By elucidating the underlying mechanisms and challenges of text clustering, our analysis informs the development of more sophisticated NLP systems capable of extracting actionable insights from unstructured text data, thereby facilitating decision-making and knowledge discovery across various domains.

# Conclusion

# In conclusion, our model training and evaluation process have illuminated effective approaches for clustering textual data, offering profound insights into genre analysis and topic modeling. By rigorously assessing the performance of feature extraction methods and clustering algorithms, we have paved the way for advancements in text mining, genre classification, and natural language processing. As the field continues to evolve, further research can build upon our findings to propel the development of innovative methodologies and applications, ultimately enriching our understanding of language and literature in the digital age.