TEXT CLUSTERING

Report

Nada Abd-Elmageed , Nada Seddik, Hadeer Mohammed, Khaled El-sakka

Introduction

A text cluster is formed by grouping a number of unlabeled texts together in such a way that texts in one cluster are more likely to be more similar than those in other clusters. so, the assignment goes this way we take five different samples of Gutenberg digital books, which are all of five different genres and of five different authors, that are semantically different. Secondly, preprocess and clean the data and create random samples of 200 documents from each book. Thirdly, applying different transformers such as BOW and TF-IDF also uses other features LDA and Word-Embedding. Fourthly, use clustering algorithms such as K-means, EM, and Hierarchical then Calculate Kappa against true authors, Consistency, Coherence, and Silhouette for each model. Finally, Compare and decide which clustering result is the closest to the human labels

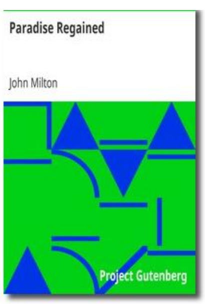
The Goal

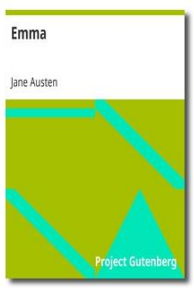
The overall aim is to produce similar clusters and compare them; analyze the pros and cons of algorithms, and Compare and decide which clustering result is the closest to the human labels

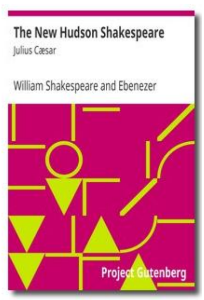
Methodology

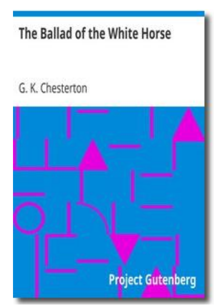
1. Selecting the Books

We picked five books with different authors and different genres from the Gutenberg Digital Library.









2. Preprocessing and Data Cleansing

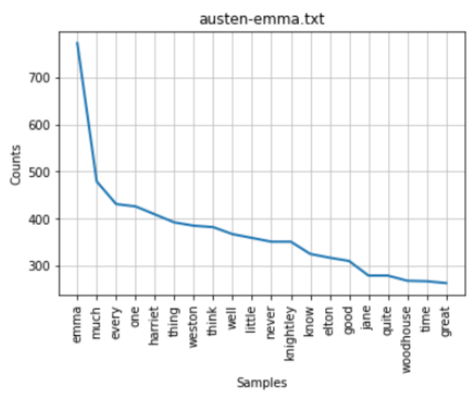
- **Import essential libraries**
  Before starting we imported libraries for many reasons that help us to read, clean, model, evaluate the data, and visualization. we use stopword, pandas, Word2Vec, Dictionary, plt, Svm, and WordNetLemmatizer, and many of them are in the code.

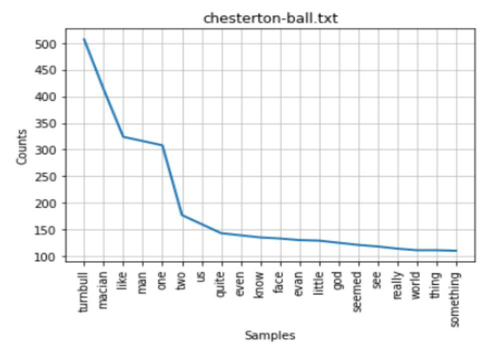
- **Read the data**
  In this stage we read the list of books from nltk.corpus.gerenberg.words, the function takes the list of books and returns a list of book words using nltk. Corpus.

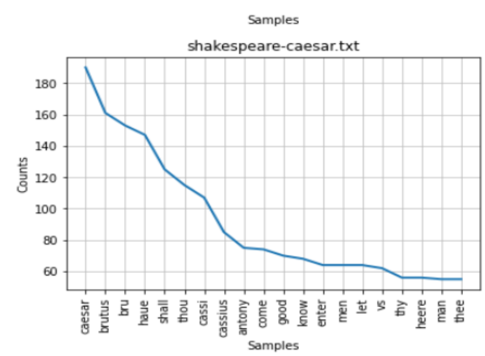
```
# list of books of the same geners (fiction book)
list_of_books=["austen-emma.txt",'milton-paradise.txt', 'bible-kjv.txt','chesterton-ball.txt','shakespeare-caesar.txt']

# list of authors
list_of_authors=["Jane Austen","JOHN MILTON","King James Version"," G.K. Chesterton ","Maria Edgeworth"]
```

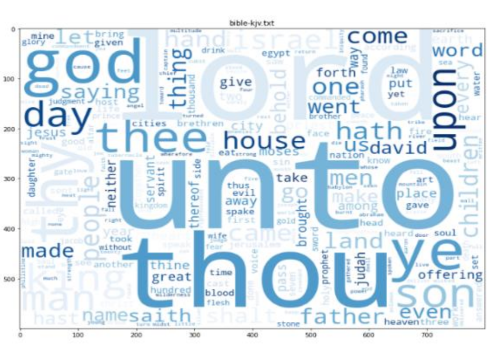
- **Clean the Data**
  It is a very important step to clean the raw text with different methods and make sure that is not contain anything that makes mis cluster or noise. We started by removing punctuations and any not needed numbers by using RegexpTokenizer from the nltk library and removing stop words, which are the noise in the text. After that, we create random samples of 200 partitions from each book. we prepared the records of 150 words records for each document. Then we performed labeling and indexing and created a Data frame. Finally, we normalize text to words by using Lemmatisation which reduces words to their word root.

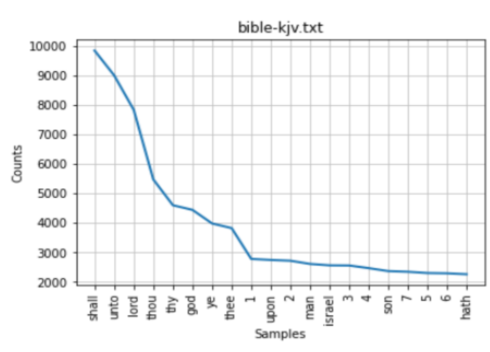
- **Exploratory Analysis (EDA)**

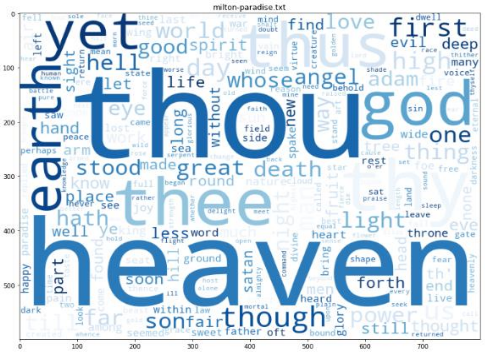


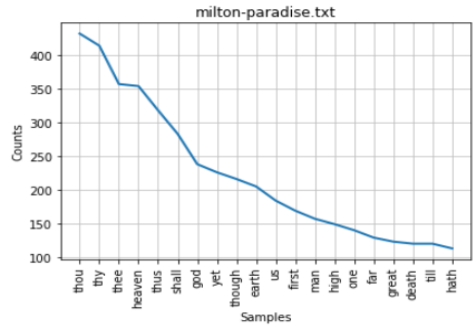












We clearly can see different word distribution in the 5 book plots...

3. Transformation

We have to convert that text into some numbers or number vectors. And we applied four types of transformations (BOW, TF-IDF, LDA, and Word-Embedding).

3.1.BOW

Bag of words is a Natural Language Processing technique of text modelling. In technical terms, we can say that it is a method of feature extraction with text data. This approach is a simple and flexible way of extracting features from documents. A bag of words is a representation of text that describes the occurrence of words within a document.

```
1 data_frame['BOW'] = list(BOWs)
2 data_frame
```

	sentences	Words	Authors_Names	BOW
0	enlighten jealousy frank churchill take countr...	[enlighten, jealousy, frank, churchill, take, ...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
1	find father quickly back weston met little per...	[find, father, quickly, back, weston, met, lit...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
2	eye emma complacently rude--but upon word wood...	[eye, emma, complacently, rude--but, upon, wor...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
3	daily visit donwell two eldest boy whose healt...	[daily, visit, donwell, two, eldest, boy, whos...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
4	friendly goodhumoured playfulness exactly suit...	[friendly, goodhumoured, playfulness, exactly,...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
...	...	...	...	...
995	bru talke stand publius good cheere harme inte...	[bru, talke, stand, publius, good, cheere, har...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
996	farewell cassius meete againe shall smile part...	[farewell, cassius, meete, againe, shall, smil...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
997	roome hoe tell antony brutus tane 1 sold ile t...	[roome, hoe, tell, antony, brutus, tane, 1, so...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
998	speak brutus groan vnderneath ages yoake haue ...	[speak, brutus, groan, vnderneath, ages, yoake...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
999	heare countrymen yet heare speake peace hoe he...	[heare, countrymen, yet, heare, speake, peace,...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...

1000 rows × 4 columns

3.2.TF-IDF

Term Frequency: In document d, the frequency represents the number of instances of a given word t. Therefore, we can see that it becomes more relevant when a word appears in the text, which is rational. Since the ordering of terms is not significant, we can use a vector to describe the text in the bag of term models. For each specific term in the paper, there is an entry with the value being the term frequency.

```
1 data_frame
```

	sentences	Words	Authors_Names	BOW	TF-IDF
0	bore people half much together jane fairfax on...	[bore, people, half, much, together, jane, fai...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
1	look join find much fault believe bad house ho...	[look, join, find, much, fault, believe, bad, ...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
2	talk farm draw music every body general inform...	[talk, farm, draw, music, every, body, general...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
3	opinion time time sure make one u think differ...	[opinion, time, time, sure, make, one, u, thin...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
4	england yes rest claim distinction many county...	[england, yes, rest, claim, distinction, many,...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
...	...	...	...	...	...
995	still full rest defence bru good reason force ...	[still, full, rest, defence, bru, good, reason...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
996	caesar seeme ambitious poore haue caesar hath ...	[caesar, seeme, ambitious, poore, haue, caesar...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
997	selfe bru cassius constant lena speakes purpos...	[selfe, bru, cassius, constant, lena, speakes,...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
998	rich noble blood world beseech yee beare hard ...	[rich, noble, blood, world, beseech, yee, bear...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...
999	night good brother tit messa good night lord b...	[night, good, brother, tit, messa, good, night...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...

1000 rows × 5 columns

3.3 Latent Dirichlet Allocation (LDA)

Latent Dirichlet allocation is one of the most popular methods for performing topic modeling. Each document consists of various words and each topic can be associated with some words. The aim behind the LDA is to find topics that the document belongs to, on the basis of words contains in it. It assumes that documents with similar topics will use a similar group of words. This enables the documents to map the probability distribution over latent topics and topics are probability distribution.

	sentences	Words	Authors_Names	BOW	TF-IDF	LDA
0	awe compassion--and rest lighten feeling Frank...	[awe, compassion--and, rest, lighten, feeling,...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.053535287305441666, 0.0016500545798690675, ...
1	Jane distress fill compassion though bad suffe...	[Jane, distress, fill, compassion, though, bad...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.12427884684350003, 0.001497382384539805, 0....
2	neighbourhood Elton first enter two year ago m...	[neighbourhood, Elton, first, enter, two, year...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.0016092065477370866, 0.0015996194420356232,...
3	Woodhouse look want aunt always sends shopping...	[Woodhouse, look, want, aunt, always, sends, s...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.0016748591521308297, 0.001665923691440583,...
4	pleasure often happiness destroyed preparation...	[pleasure, often, happiness, destroyed, prepar...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.13677150366659174, 0.0017302299054737628, 0...
...	...	...	...	...	...	...
995	receiuest Thy full Petition hand Brutus Enter ...	[receiuest, Thy, full, Petition, hand, Brutus,...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.27101906431645917, 0.001432489005080659, 0....
996	Pompeyes Statue ran blood great Caesar fell fa...	[Pompeyes, Statue, ran, blood, great, Caesar, ...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.0014496942111736353, 0.001451690524334299, ...
997	Ayre Horsses neigh die men grone Ghosts shriek...	[Ayre, Horsses, neigh, die, men, grone, Ghosts...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.0015128503815490862, 0.0015169730676517443,...
998	wrought dispos'd therefore meet Noble mindes k...	[wrought, dispos'd, therefore, meet, Noble, mi...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.3532521152371669, 0.0014602659475543098, 0....
999	Messala meditate dye haue patience endure Mess...	[Messala, meditate, dye, haue, patience, endur...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.0014999523392384094, 0.0015306802175092977,...

3.4 Doc2Vec

Doc2Vec is an unsupervised algorithm that learns embeddings from variable-length pieces of texts, such as sentences, paragraphs, and documents.

```
[52] 1 data_frame
```

	sentences	Words	Authors_Names	BOW	TF-IDF	LDA	Doc2Vec
0	awe compassion--and rest lighten feeling Frank...	[awe, compassion--and, rest, lighten, feeling,...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.053535287305441666, 0.0016500545798690675, ...	[0.078669, -0.26765475, 0.061254326, -0.064297...
1	Jane distress fill compassion though bad suffe...	[Jane, distress, fill, compassion, though, bad...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.12427884684350003, 0.001497382384539805, 0....	[0.19912769, -0.29548597, 0.11015332, -0.05147...
2	neighbourhood Elton first enter two year ago m...	[neighbourhood, Elton, first, enter, two, year...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.0016092065477370866, 0.0015996194420356232,...	[0.10718377, -0.31028664, 0.040655583, -0.0655...
3	Woodhouse look want aunt always sends shopping...	[Woodhouse, look, want, aunt, always, sends, s...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.0016748591521308297, 0.001665923691440583,...	[0.026540969, -0.15322354, 0.016255617, -0.091...
4	pleasure often happiness destroyed preparation...	[pleasure, often, happiness, destroyed, prepar...	2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.13677150366659174, 0.0017302299054737628, 0...	[0.054894533, -0.30789623, -0.028229777, 0.081...
...	...	...	...	...	...	...	...
995	receiuest Thy full Petition hand Brutus Enter ...	[receiuest, Thy, full, Petition, hand, Brutus,...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.27101906431645917, 0.001432489005080659, 0....	[0.18549646, 0.15637437, -0.14605208, 0.342126...
996	Pompeyes Statue ran blood great Caesar fell fa...	[Pompeyes, Statue, ran, blood, great, Caesar, ...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.0014496942111736353, 0.001451690524334299, ...	[0.055284508, 0.14853376, -0.09836552, -0.1557...
997	Ayre Horsses neigh die men grone Ghosts shriek...	[Ayre, Horsses, neigh, die, men, grone, Ghosts...	4	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.0015128503815490862, 0.0015169730676517443,...	[-0.056478955, -0.13509864, -0.05936547, 0.140...

4. Modeling (Clustering algorithms)

We applied 3 unsupervised clustering algorithms:

- K-Means
- EM
- Hierarchical

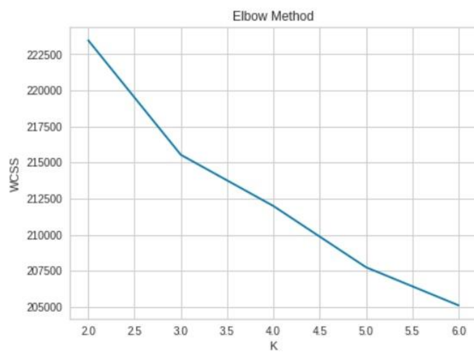
4.1 K-Mean

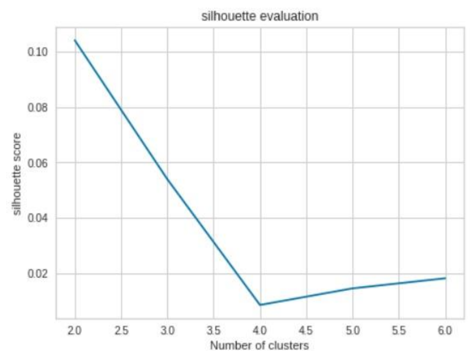
Finding groups in the data is the objective of the K-Means algorithm, with the variable K indicating how many groups there are in total. The algorithm assigns via an iterative process. Based on the above features, each point of data belongs to one of the K groups. Data Based on feature similarity, points are grouped. We employ the El-Bow Method to optimize the number of clusters while implementing the K-Means Model with one of each transformation approach, even though we only require 5 clusters

a) **K-Mean with BOW**

Here we applied the k-mean to the BOW transformer, from using the Elbow method the best number of clusters is 5

```
models_evaluation_dictionary["Kmeans_BOW"] = Kmeans_model(list(df['BOW']))
```



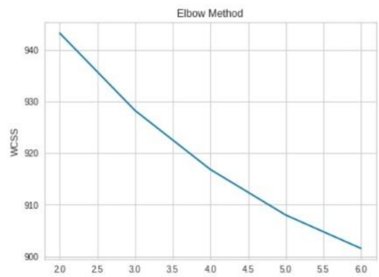


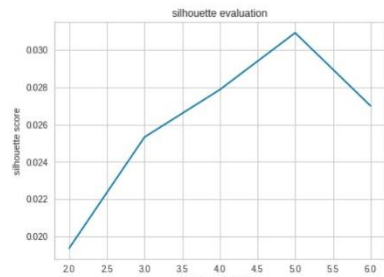

```
Kmeans
Best K value = 2, with silhouette_score = 0.10402185477631332
the cohen_kappa_score =  0.21750000000000003
the homogeneity_score =  0.19511050206105182
the completeness_score =  0.594539282275722
the v_measure_score =  0.2938033040995827
```

b) K-Mean with TF-IDF

We applied K-mean with TF-IDF, we get the best K is 5 clusters and the silhouette score are 0.03.

```
models_evaluation_dictionary["Kmeans_TF-IDF"] = Kmeans_model(list(df['TF-IDF']))
```

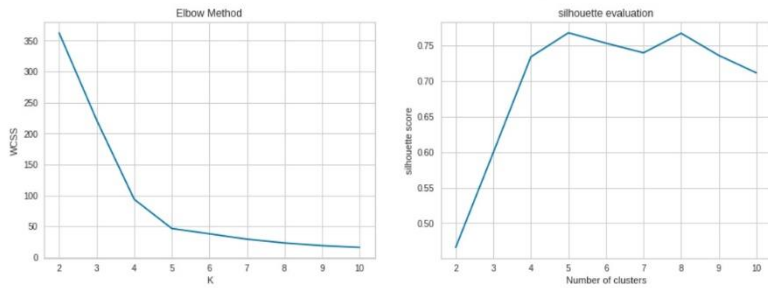





```
Kmeans
Best K value = 5, with silhouette_score = 0.03092437536935884
the cohen_kappa_score =  0.9775
the homogeneity_score =  0.951147364695347
the completeness_score =  0.9516047049378852
the v_measure_score =  0.9513759798540962
```

c) K-Mean with LDA
We applied K-mean with LDA, we get the best K is 5 clusters and the silhouette score are 0.76

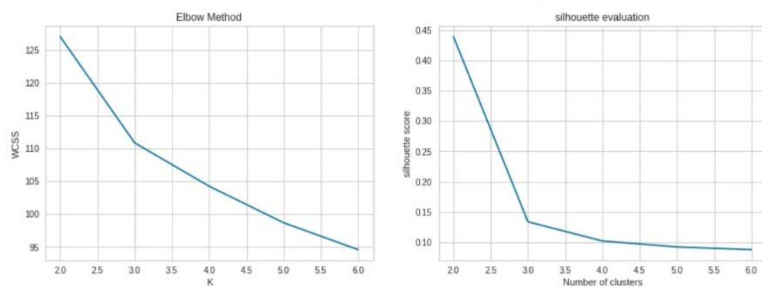
```
models_evaluation_dictionary["Kmeans_LDA"] = Kmeans_model(list(df['LDA']),n_components=11)
```




```
Kmeans
Best K value = 5, with silhouette_score = 0.7682198286513264
the cohen_kappa_score =  0.8
the homogeneity_score =  0.8007667527923448
the completeness_score =  0.8534623371464443
the v_measure_score =  0.8262752341909452
```

d) K-Mean with Doc2Vic
We applied K-mean with LDA, we get the best K is 2 clusters and the silhouette score are 0.43

```
models_evaluation_dictionary["Kmeans_Doc2Vec"] = Kmeans_model(list(df['Doc2Vec']))
```




```
Kmeans
Best K value = 2, with silhouette_score = 0.4389464259147644
the cohen_kappa_score =  0.042499999999999998
the homogeneity_score =  0.028335336538354055
the completeness_score =  0.18961429979882696
the v_measure_score =  0.049302995752395794
```

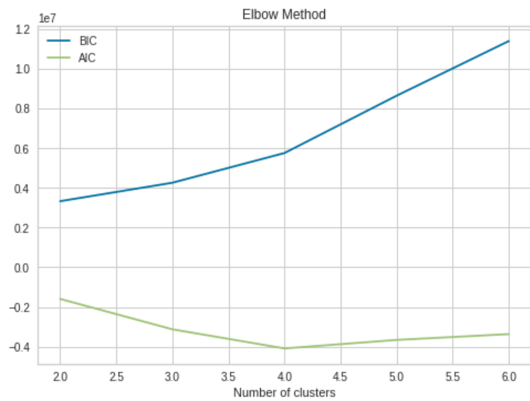
4.2 EM

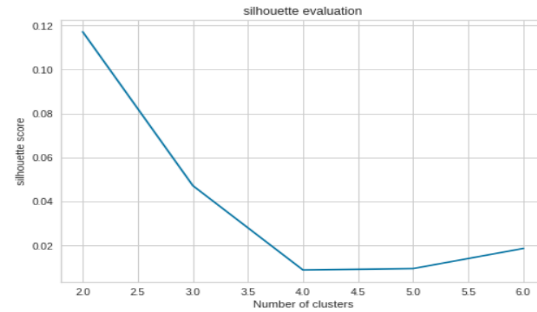
Similar to k-means but instead of using k points, we use k Gaussian mixture distributions. The model then tries to find the best parameters' values for the Gaussian that fit the data well.

<u>Elbow method measures</u>
AIC stands for Akaike information criterion for the current model on the input X.
BIC stands for Bayesian information criterion for the current model on the input X.

a) **EM with BOW**

The shape of the input data for the model is (1000, 12315) and it takes too much time and memory storage for training and evaluation so we used the PCA dimensionality reduction technique to reduce the number of features from 12315 to 1000.



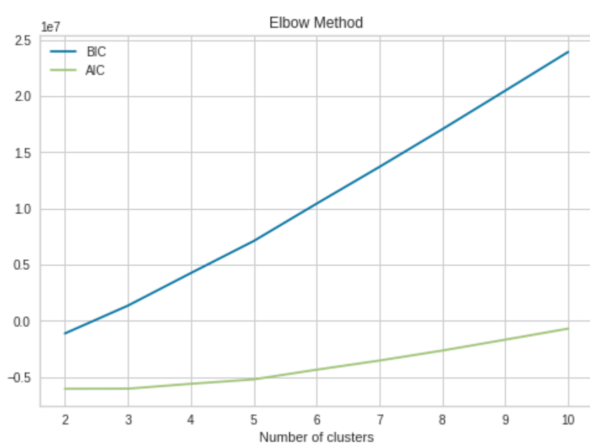


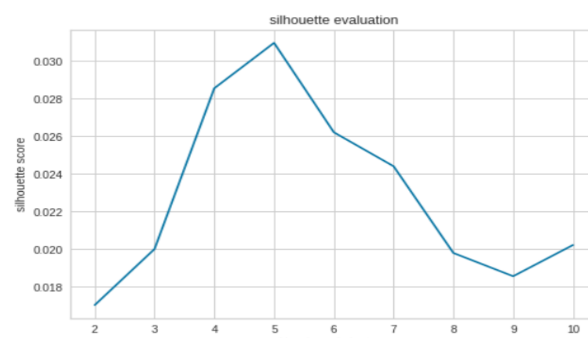

```
EM
Best K value = 2, with silhouette_score = 0.11712436637134505
the cohen_kappa_score =  0.20999999999999996
the homogeneity_score =  0.18488224529471656
the completeness_score =  0.5804522430497108
the v_measure_score =  0.2804402927497223
```

The model has the highest silhouette value at k = 2 which is far from reality. From the Elbow method the AIC state that the best value is 4 but on the other hand it is the worst value when it comes to the silhouette. Also, the Kappa the other measures scores are very small near 0.

b) **EM with TF-IDF**

We encountered the same problem as with BOW so we used the PCA dimensionality reduction technique to reduce the number of features from 12315 to 1000.
The Elbow method with BIC and AIC doesn't show anything the BIC is increasing all the time and there's no elbow here. There may be a slight elbow for AIC at a number of clusters = 3



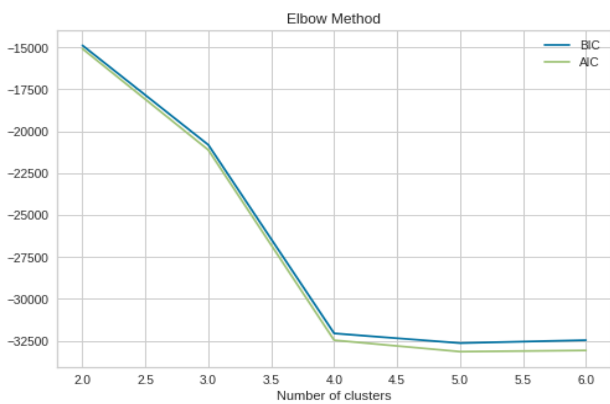


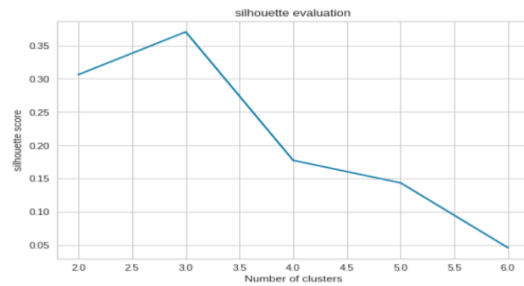

```
EM
Best K value = 5, with silhouette_score = 0.030972286377508047
the cohen_kappa_score =  0.9925
the homogeneity_score =  0.9831441396800142
the completeness_score =  0.9832541236056475
the v_measure_score =  0.983199128567039
```

It is totally clear from the silhouette evaluation that the best number of clusters is 5 and it gives very good values of the Kappa = 0.99 meaning that the predicted clustering labels are almost equal to the actual labels, homogeneity, completeness, and V-measure scores which are so close to 1 so the clusters are well separated.

c) **EM with LDA**

We hadn't any problem with the computational power here as the number of features is really small (5). From the Elbow method's perspective, the best value for the number of clusters = 4 using both AIC and BIC measures.



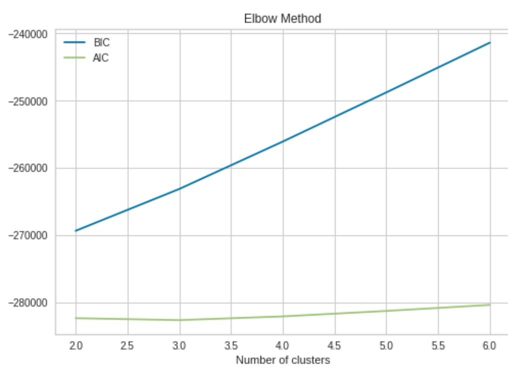


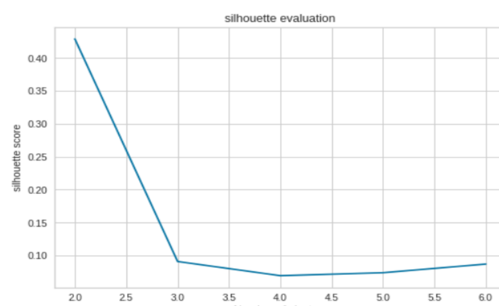

```
EM
Best K value = 3, with silhouette_score = 0.3704785722583183
the cohen_kappa_score =  0.44875
the homogeneity_score =  0.42720734780316205
the completeness_score =  0.6732911361038326
the v_measure_score =  0.5227356961604042
```

Here it is different, using the silhouette the best number of clusters is 3. In that case, the Kappa score and the other measures are on average near 0.5.

d) **EM with Doc2Vec**:

The Elbow method with BIC and AIC doesn't show anything the BIC is increasing all the time and there's no elbow here. There may be a slight elbow for AIC when the number of clusters = 3






```
EM
Best K value = 2, with silhouette_score = 0.42875853180885315
the cohen_kappa_score =  0.048749999999999996
the homogeneity_score =  0.03738763224308582
the completeness_score =  0.23969082165382008
the v_measure_score =  0.06468545039139305
```

The best number of clusters = 2 measured from the highest silhouette values. These are very bad scores the Kappa score = 0.04 which is a very small value meaning that the predicted labels are so far from the actual labels. Also, the other measures' values are very small meaning that the model doesn't separate the clusters well

4.3 **Hierarchical clustering:** is a general family of clustering algorithms that build nested clusters by merging or splitting them successively. This hierarchy of clusters is represented as a tree (or dendrogram). The root of the tree is the unique cluster that gathers all the samples, the leaves being the clusters with only one sample.

1. Hierarchical clustering with BOW transformation:

```
models_evaluation_dictionary["Hierarchical_BOW"] = Hierarchical(list(df["BOW"]))
```

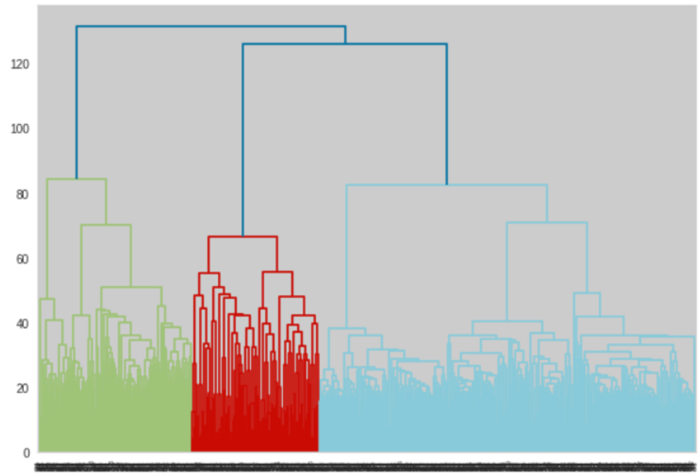
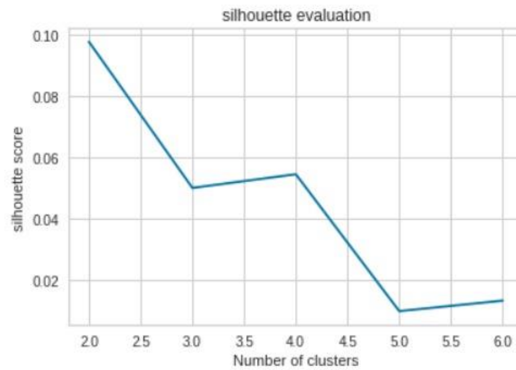


Figure 2 Hierarchical Cluster with BOW




```
Hierarchical
Best K value = 2, with silhouette_score = 0.09784655052994483
the cohen_kappa_score =  0.24
the homogeneity_score =  0.24313840612261273
the completeness_score =  0.7192457875096975
the v_measure_score =  0.36342299788918747
```

Figure 1 Hierarchical Cluster Evaluation with BOW

2. Hierarchical clustering with TFIDF transformation:

```
models_evaluation_dictionary["Hierarchical_TF-IDF"] = Hierarchical(list(df["TF-IDF"]))
```

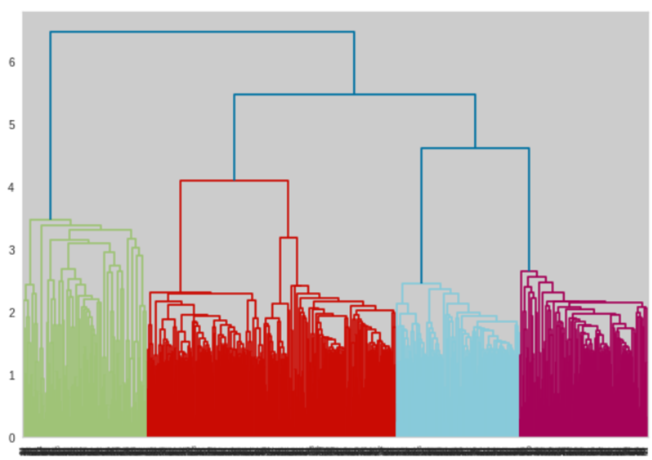
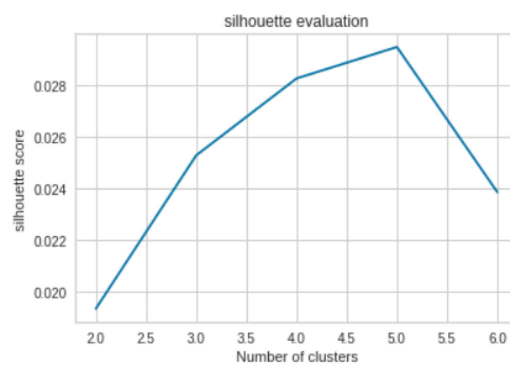


Figure 4:Hierarchical Clustering with TFIDF Transformer




```
Hierarchical
Best K value = 5, with silhouette_score = 0.029476961617526044
the cohen_kappa_score =  0.96125
the homogeneity_score =  0.9228519183948832
the completeness_score =  0.9233909187014164
the v_measure_score =  0.9231213398690822
```

Figure 3: Hierarchical Clustering Evaluation TFIDF Transformation

3. Hierarchical clustering with LDA transformation:

```
models_evaluation_dictionary["Hierarchical_LDA"] = Hierarchical(list(df["LDA"]),n_components = 9)
```

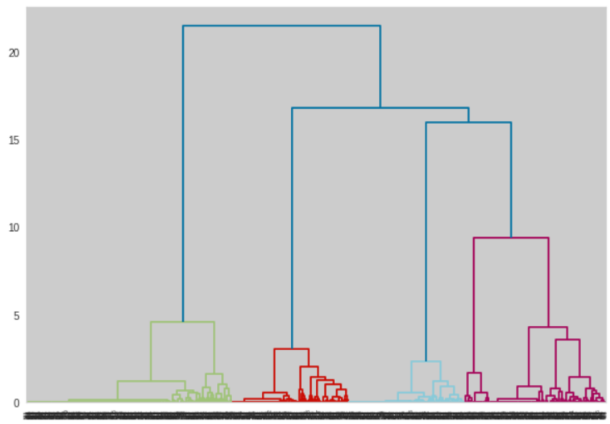
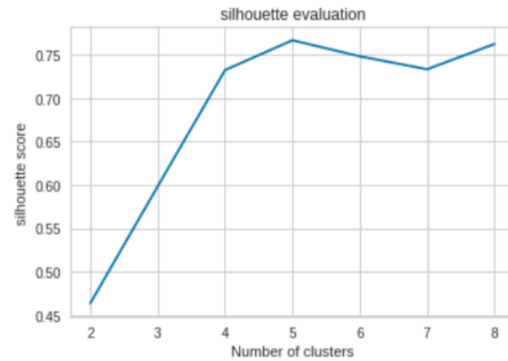


Figure 5: Hierarchical Clustering with LDA Transformation




```
Hierarchical
Best K value = 5, with silhouette_score = 0.7669098404862652
the cohen_kappa_score =  0.79875
the homogeneity_score =  0.8245917470247597
the completeness_score =  0.9035198851541059
the v_measure_score =  0.8622533714809469
```

Figure 6: Hierarchical Clustering Evaluation with LDA Transformation

4. Hierarchical clustering with Doc2Vec transformation:

```
models_evaluation_dictionary["Hierarchical_Doc2Vec"] = Hierarchical(list(df["Doc2Vec"]))
```

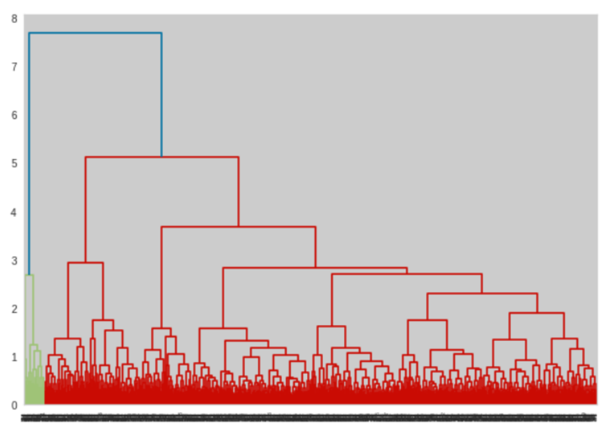
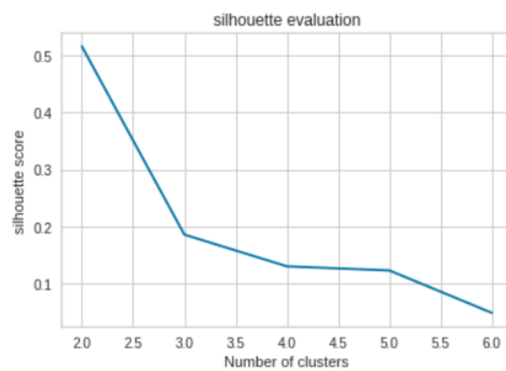


Figure 7: Hierarchical Clustering with Doc2Vec Transformation




```
Hierarchical
Best K value = 2, with silhouette_score = 0.5166573524475098
the cohen_kappa_score =  0.027499999999999997
the homogeneity_score =  0.01664093466641861
the completeness_score =  0.16919938937369638
the v_measure_score =  0.03030166890537523
```

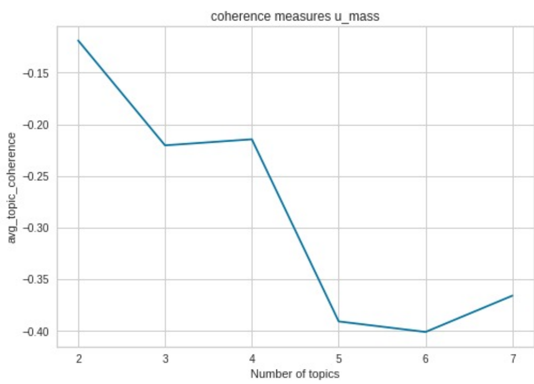
Figure 8: Hierarchical Clustering Evaluation with Doc2Vec

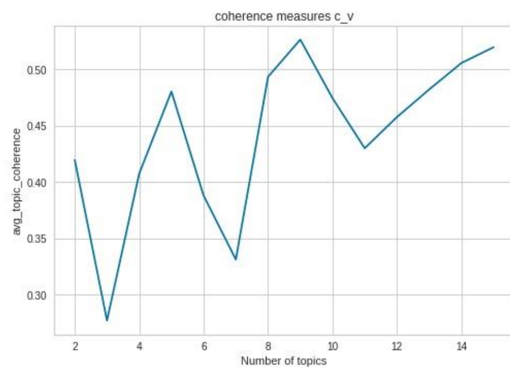
5. Evaluation

We try 6 different measurements from each model to each transformer:

- Silhouette
- Cohen's Kappa
- Coherence
- Homogeneity score
- Completeness score
- V measure score

- **Coherence**
  It measures how the words for each topic are related to that topic and related to each other.
  We used it to get insights and to help us select the best number of topics.





- **Silhouette score**
  Measures how each cluster's points are close to each other and separated from the other clusters. The best value is 1 and the worst value is -1. Values near 0 indicate overlapping clusters. Negative values generally indicate that a sample has been assigned to the wrong cluster, as a different cluster is more similar.
- **Cohen kappa score**
  Measures the similarity between the true labels and the predicted labels.
  A kappa statistic is a number between -1 and 1. The maximum value means complete agreement; zero or lower means chance agreement.
- **Homogeneity score:**
  A clustering result satisfies homogeneity if all of its clusters contain only data points that are members of a single class.
  The result is between 0 and 1. Score = 1 for perfectly homogeneous labeling.
- **Completeness score:**
  A clustering result satisfies completeness if all the data points that are members of a given class are elements of the same cluster.
  The result is between 0 and 1. Score = 1 for perfectly complete labeling.

- **V measure score**
  Measures the harmonic mean between homogeneity and completeness.
  The result is between 0 and 1. Score = 1 for perfectly complete labeling.

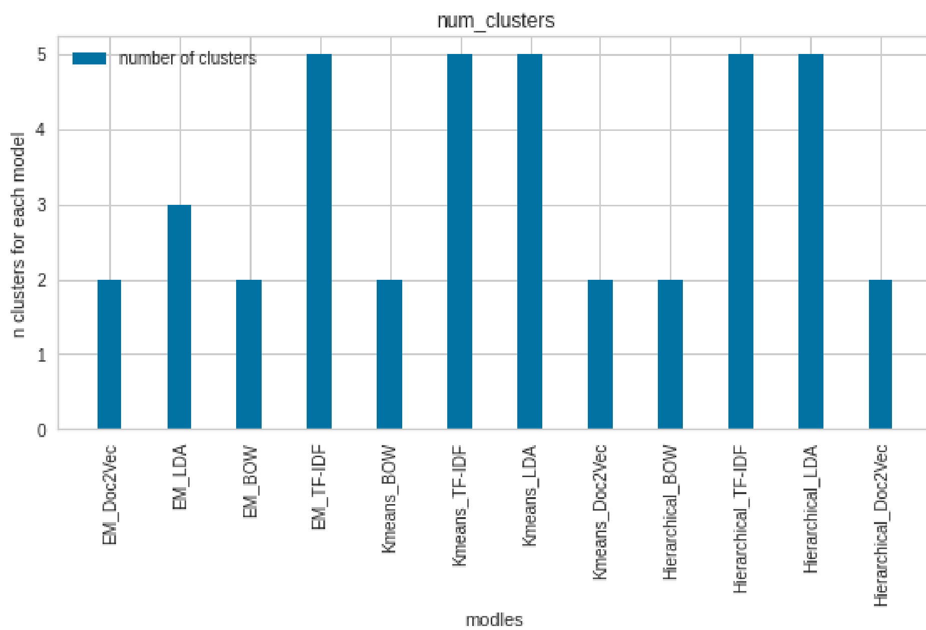
In general, we select the best number of clusters for each model based on the highest **silhouette score**

	Silhouette	**Cohen's Kappa**	**Homogeneity score**	**Completeness score**	**V measure score**
K-Mean with BOW	**0.10**	**0.21**	**0.19**	**0.59**	**0.29**
K-Mean with TF-IDF	**0.03**	**0.97**	**0.95**	**0.95**	**0.95**
K-Mean with LDA	**0.76**	**0.80**	**0.80**	**0.85**	**0.85**
K-Mean with Doc2Vec	**0.43**	**0.04**	**0.02**	**0.18**	**0.04**

	Silhouette	**Cohen's Kappa**	**Homogeneity score**	**Completeness score**	**V measure score**
EM with BOW	**0.11**	**0.20**	**0.18**	**0.58**	**0.28**
EM with TF-IDF	**0.03**	**0.99**	**0.98**	**0.98**	**0.98**
EM with LDA	**0.37**	**0.44**	**0.42**	**0.67**	**0.52**
EM with Doc2Vec	**0.42**	**0.04**	**0.03**	**0.23**	**0.06**

	Silhouette	Cohen's Kappa	Homogeneity score	Completeness score	V measure score
Hierarchical with BOW	**0.097**	**0.24**	**0.243**	**0.71**	**0.36**
Hierarchical with TF-IDF	**0.029**	**0.96**	**0.92**	**0.92**	**0.92**
Hierarchical with LDA	**0.76**	**0.79**	**0.82**	**0.90**	**0.86**
Hierarchical with Doc2Vec	**0.51**	**0.027**	**0.016**	**0.169**	**0.03**

6. Comparison and the champion Model



From the human perspective, we know we know that the best number of clusters is 5 which equals the number of books. We eliminated all the models other than that. Now we have 5 models to compare (EM-TFI-DF, Kmeans_TF-IDF, Kmeans_LDA, Hirarchical_TF-IDF, Hirarchical_LDA).

```
pca_tfidf = PCA_function(list(df["TF-IDF"]))
my_scater_plot(pca_tfidf,"Before Clustering with pca" )
```

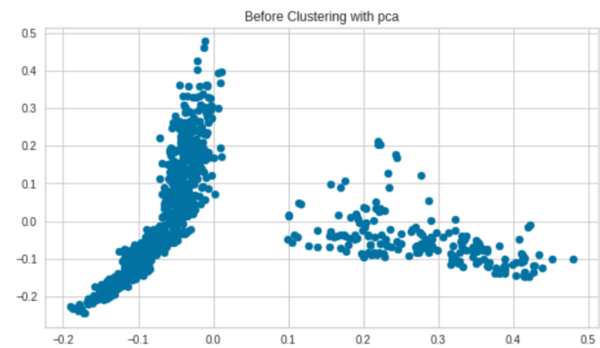


Figure 10 Before Clustering with PCA

```
model_pca = GaussianMixture(5 , covariance_type='full', random_state=0)

em_labels = model_pca.fit_predict(pca_tfidf)
my_scater_plot(pca_tfidf,"EM Clustering with pca" ,centers = model_pca.means_, labels =em_labels)
```

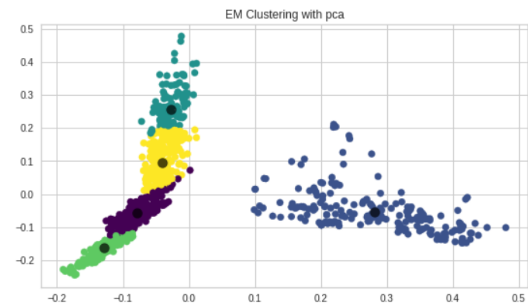
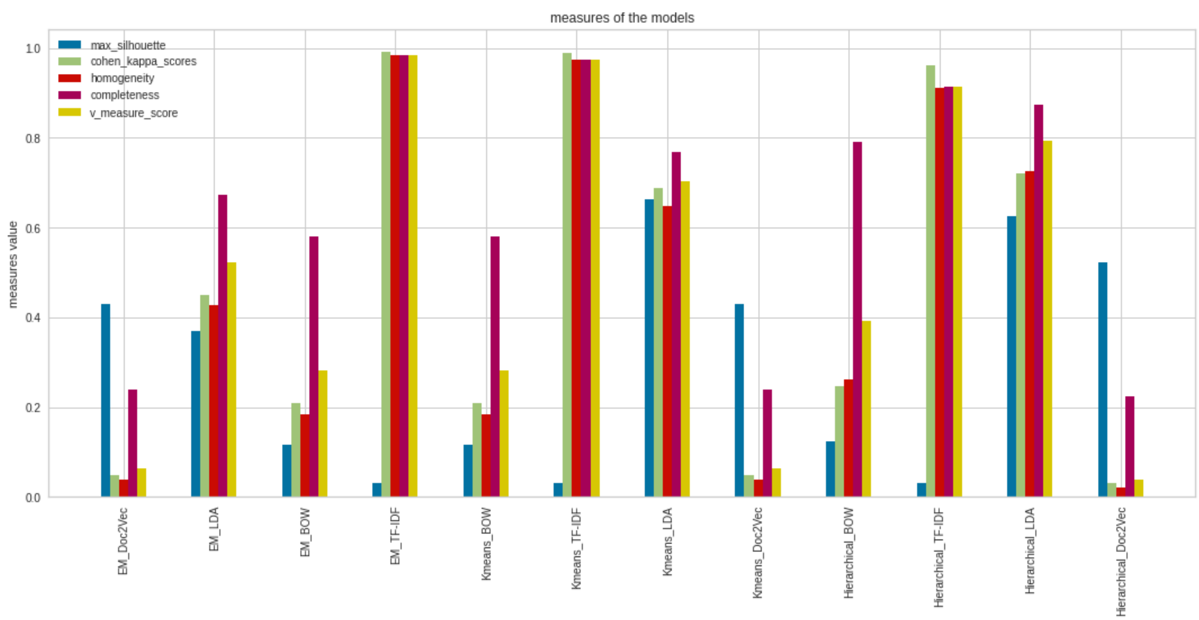


Figure 9 EM Clustering with PCA

Now we move to our measures:



Although we applied dimensionality reduction with PCA it has the highest Kappa score meaning that the predicted clustering labels are almost equal to the actual labels, also homogeneity, completeness, and V-measure scores are so close to 1 meaning that the clusters are well separated.

7. Error Analysis

For our champion, we tracked all the records that were predicted in a wrong cluster with respect to the actual true clustering labels. We found only 4 records then, and we tried to find the 10 most frequent words in those records. Those words are the main cause of the wrong labeling and if we removed them the model would perform better.