CLUSTERING PATENT DATA USING KMEANS APPROACH

Background:

Today patent database is growing in size and companies want to explore this dataset to have an edge for its competitor. Patent data have diversely technological information of any technology field. So, many companies have managed the patent data to build their R&D policy. Patent analysis is an approach to the patent management. In addition, patent analysis is an important tool for technology forecasting.

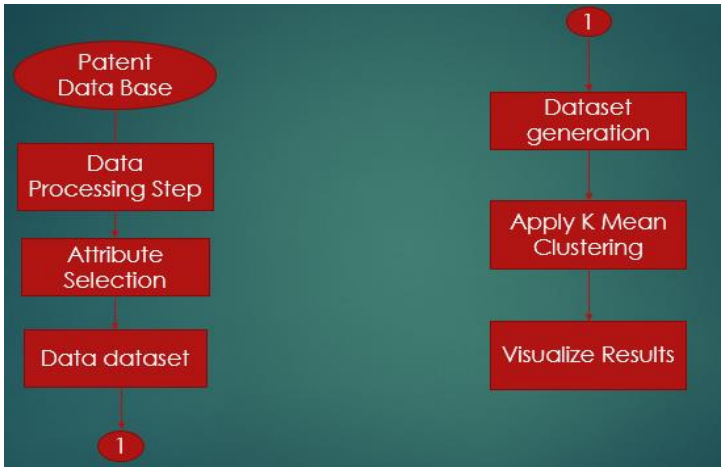
One of the most polarizing collection of tasks, associated with patent analytics, is the use of machine learning methods for organizing, and prioritizing documents. Patent clustering is one of the works for patent analysis. Clustering is the task of grouping datasets either physical or abstract objects into classes of similar objects.

K-means is a simple clustering technique, which groups the similar items in the same cluster and dissimilar items in different cluster. In this study, the metadata associated with database will be used as attribute for clustering. The performance will be validated via Davies–Bouldin index.

Approach used:

The basic approach include data mining, text mining and visualization techniques. Classification and clustering are popular methods of patent analysis. Unstructured data uses the text mining approach for datasets like images, tables, figures. Natural Language Processing (NLP) technique is widely used for patent documents which are highly unstructured in nature.

Clustering the unsupervised classification technique helps the patents to be divided into groups based on similarities of the internal features or attributes. Presently, the clustering algorithm used in text clustering is the K-means clustering algorithm.



Data Acquisition Process:

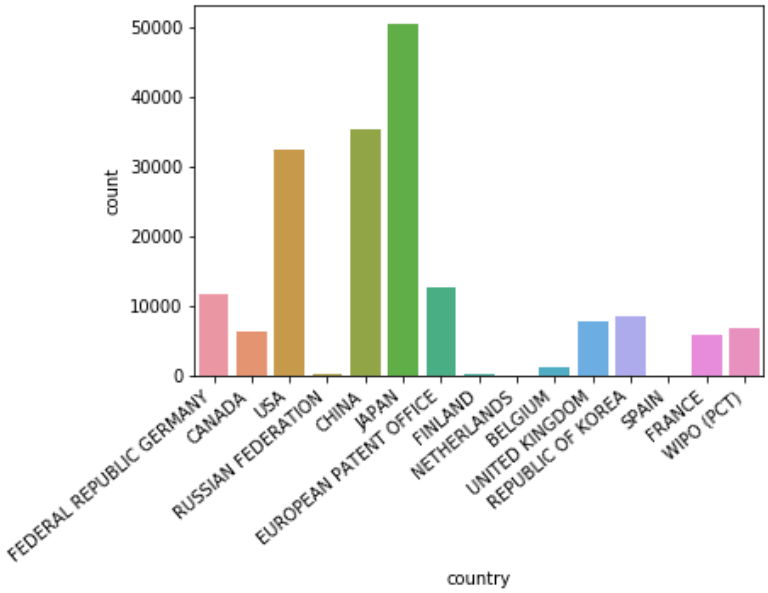
- **Data Source** : Google's BigQuery Data warehouse
- **Dataset name:** google_patents_research
- **Table name** : publications_201802
- **Link:**

https://bigquery.cloud.google.com/dataset/patents-public-data:google_patents_research

- **Data Attributes:** publication number, title, top terms, country, publication_description
- The data is selected from the table using standard SQL query.
- The size of the data the query processed is estimated to be 240 GB.

Few exploratory data analysis are made and the below conclusions are drawn:

- The types of patent documents in the publication description column ranges from 1 to 25468.
- Hence it was filtered to have counts above 1000 only. This limited the unnecessary and least significant category.
- The country which has largest number of patent release is Japan (50485) and the smallest number is contributed by Netherlands(6) during February 2018.



Tools used for the project:

- Jupyter Notebook in Google Colab (with GPU- 1 K80 core provided for google account users)
- Google Cloud datawarehouse
- BigQuery Client.

Text mining approach:

There are actually many methods to convert a corpus to a vector format. The simplest is the bag-of-words approach, where one number will represent each unique word in a text.

First, we write a function that will split a text into its individual words and return a list. We also remove very common words, ('the', 'a', etc.). To do this we will take advantage of the NLTK library. It is the standard library in Python for processing text and has a lot of useful features. We only use some of the basic ones here.

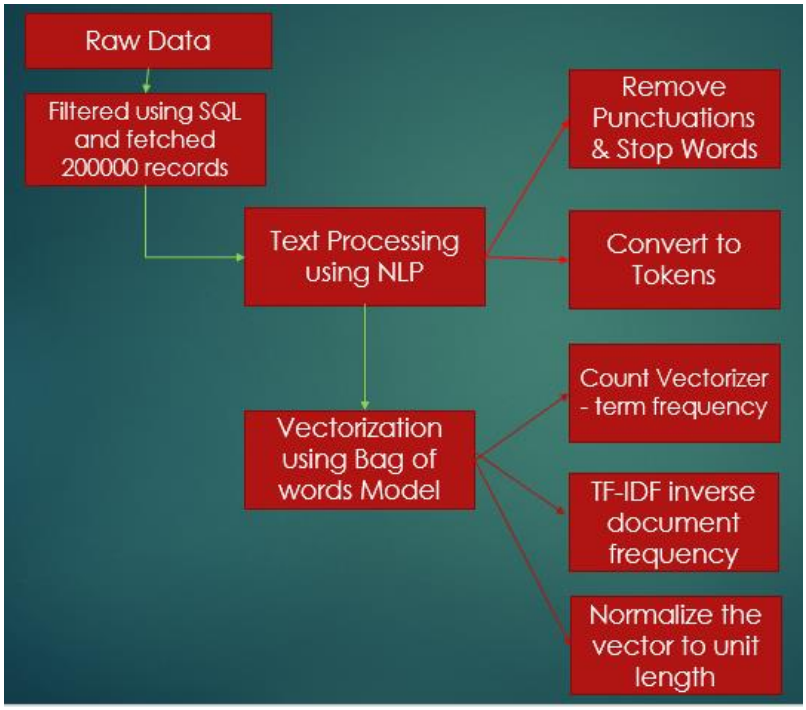
We create a function that will process the string in the publication_description column, then we can just use **apply()** in pandas to process all the text in the DataFrame.

For removing punctuation, we can just take advantage of Python's built-in **string** library to get a quick list of all the possible punctuation.

We convert each patent type, represented as a list of tokens (lemmas), into a vector that machine-learning models can understand.

We do that in three steps using the bag-of-words model:

1. Count how many times does a word occur in each message (Known as term frequency)
2. Weigh the counts, so that frequent tokens get lower weight (inverse document frequency)
3. Normalize the vectors to unit length, to abstract from the original text length (L2 norm)



KMeans Clustering: Method and Results

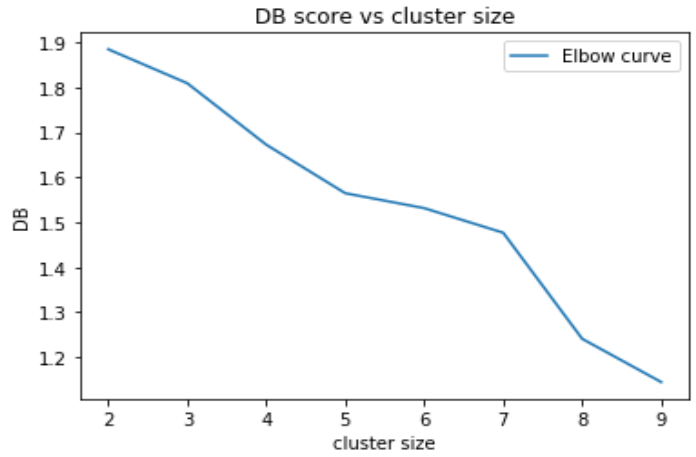
K-means looks to create a fixed number of clusters, and moves new documents to the cluster that has the most similarity, to the other documents, in that cluster.

An automated approach would be to build a collection of k-means clustering models with a range of values for k from 2 to 10 and then evaluate each model to determine the optimal number of clusters.

We use Davies-Bouldin score to evaluate each model. The score is defined as the ratio of within-cluster distances to between-cluster distances. The code for this project is placed below:

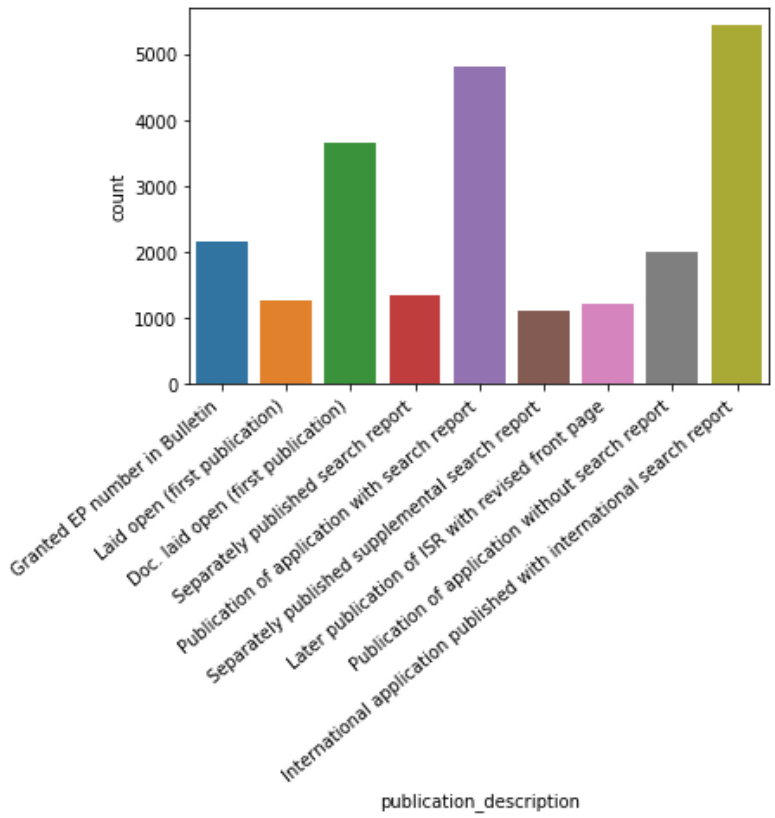
https://github.com/Harinipraveen/Arcada-ML-projects/blob/master/patent_bigData_project.ipynb

From our elbow method approach we estimate the optimal number of clusters to be 5(cluster 0,1,2,3,4) and the DB index for 5 clusters is estimated to be 1.56.

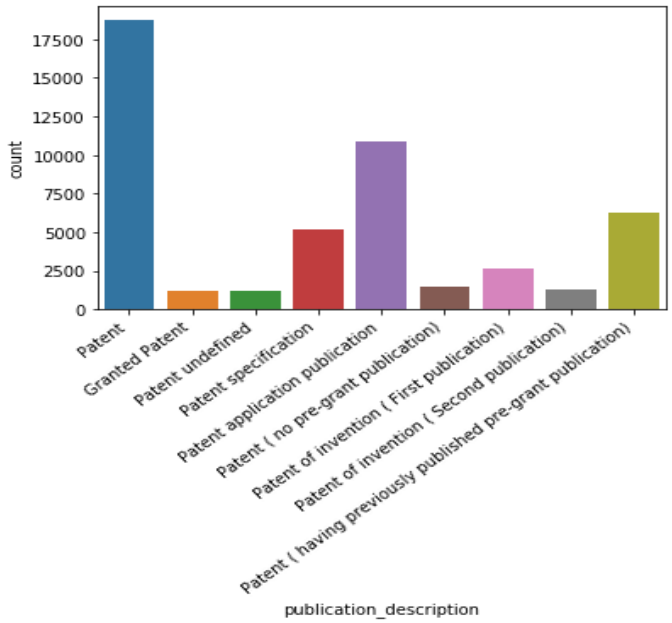


The predicted cluster labels are mapped to the index of the original dataset to determine the data points in each cluster. We observe the following results:

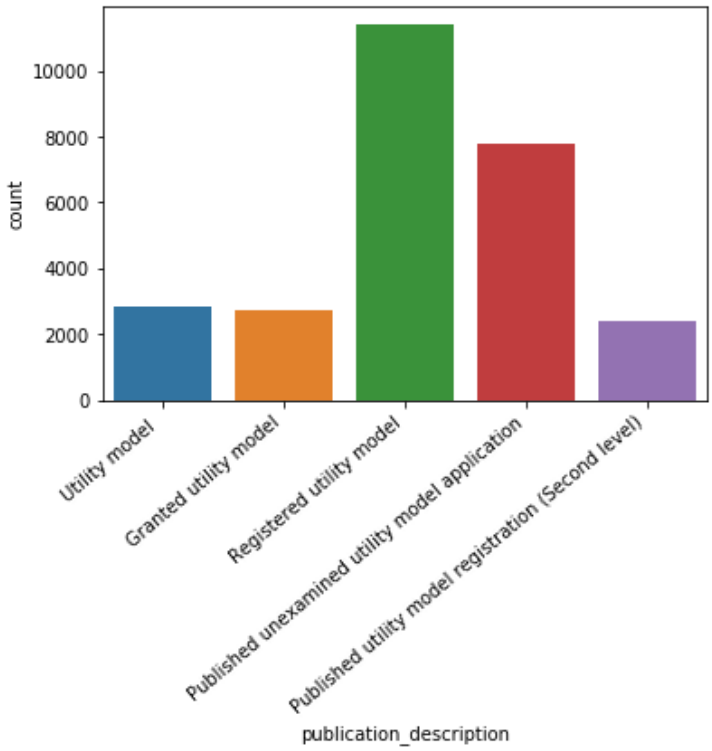
1. The total no of patents in the first cluster (cluster 0) is 22974. In this group most of the patents are Publication of application with or without search reports or laid open documents.



2. The total no of patents in the second cluster (cluster 1) is 48834. All patent of invention and patent specification are grouped are grouped together in cluster 1.



3. The total no of patents in the fourth cluster(cluster 3) is 27157. Utility models which are granted,registered or unexamined are grouped in cluster 3.



Conclusion:

In this project, K-means cluster approach is used for clustering the patent dataset based on the metadata attributes like publication number, title, top terms, country,publication_description. The experimental results show that optimal clustering in this case study is obtained for k = 5. This is evaluated using DB index for k values ranging from 2 to 10. Data points belonging to each cluster are plotted and the similarity of data points are noted. A lateral benefit is that these clusters can be used as means to speed up shifting through large sets of patent data or narrowing down to the specific technology or area we are interested in and in this way, drastically reduce the time taken to categorize large patent sets.