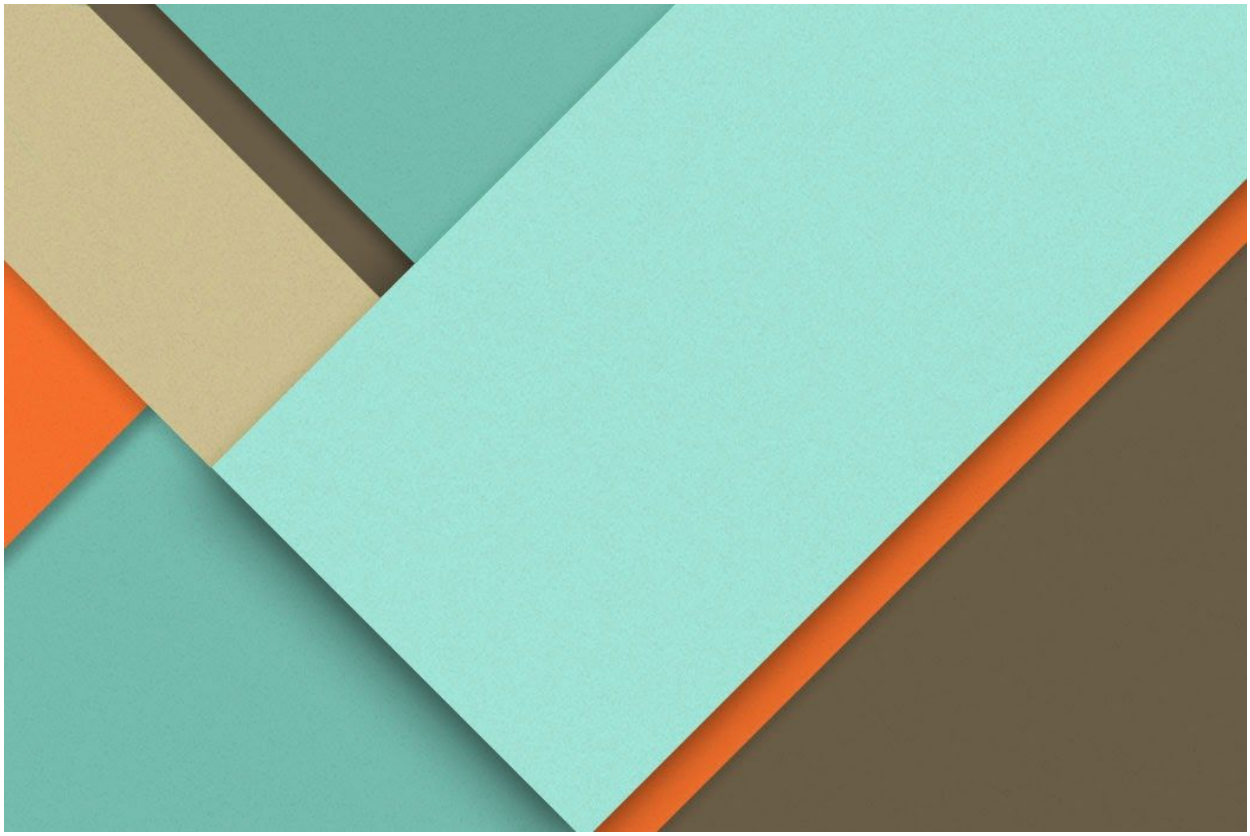


PML Project 2: Comparing Unsupervised Learning Models with Supervised Models

20.01.2021

Aslantas Serdal
Data Science- 411

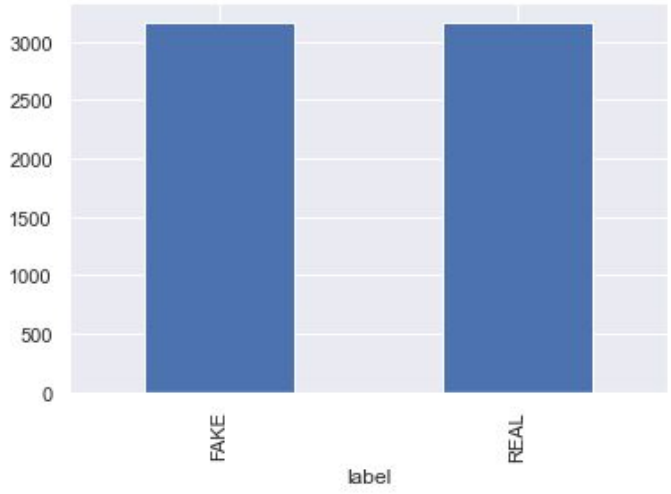
Description Of the Dataset

The dataset consists of 6335 fake and real news collected from the variety of news agencies such as Reuters as well as social media platforms. The dataset is already labelled as Real and Fake for the comparison of the performance of supervised vs unsupervised models purposes. The features of the dataset are “id, title, text, label” .

Goals

1. Measuring the performance of unsupervised learning model K-Means Algorithm and DBSCAN on detection of fake news on a given dataset.
2. Comparing the performance of supervised and unsupervised models.

Looking at the data with Pandas



The numbers of real and fake news are balanced.

I joined the title and the text and trained the only text data.

```
In [4]: 1 df.head()
```

Out[4]:

	Unnamed: 0	title	text	label	label_num
0	8476	You Can Smell Hillary's Fear	You Can Smell Hillary's Fear Daniel Greenfield...	FAKE	0
1	10294	Watch The Exact Moment Paul Ryan Committed Pol...	Watch The Exact Moment Paul Ryan Committed Pol...	FAKE	0
2	3608	Kerry to go to Paris in gesture of sympathy	Kerry to go to Paris in gesture of sympathy U....	REAL	1
3	10142	Bernie supporters on Twitter erupt in anger ag...	Bernie supporters on Twitter erupt in anger ag...	FAKE	0
4	875	The Battle of New York: Why This Primary Matters	The Battle of New York: Why This Primary Matte...	REAL	1

Preprocessing

I used Tfidf Vectorization and Word2Vec methods to convert the text data into numerical values. They both make the required data cleaning preprocessing such as tokenizing, removing the punctuation, stopwords etc.

1. Tfidf Vectorization: Term Frequency - Inverse Document Frequency :

   TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document).

   IDF(t) = log_e(Total number of documents / Number of documents with term t in it).

2. **Word2vec** is a technique for natural language processing. The word2vec algorithm uses a neural network model to learn word associations from a large corpus of text. Once trained, such a model can detect synonymous words or suggest additional words for a partial sentence. As the name implies, word2vec represents each distinct word with a particular list of numbers called a vector. The vectors are chosen carefully such that a simple mathematical function (the cosine similarity between the vectors) indicates the level of semantic similarity between the words represented by those vectors. (Wikipedia)

```
In [8]: 1 model.wv.most_similar("america")

Out[8]: [('country', 0.7263306379318237),
         ('nation', 0.6991192698478699),
         ('great', 0.6610617637634277),
         ('“make', 0.6520432233810425),
         ('conformed', 0.6460477113723755),
         ('“chairman', 0.636742353439331),
         ('soon…', 0.6360567212104797),
         ('“united”', 0.6240637898445129),
         ('coup\x94', 0.6234774589538574),
         ('serf's', 0.6234768629074097)]
```

Training The Models

I. K- Means Clustering Algorithm

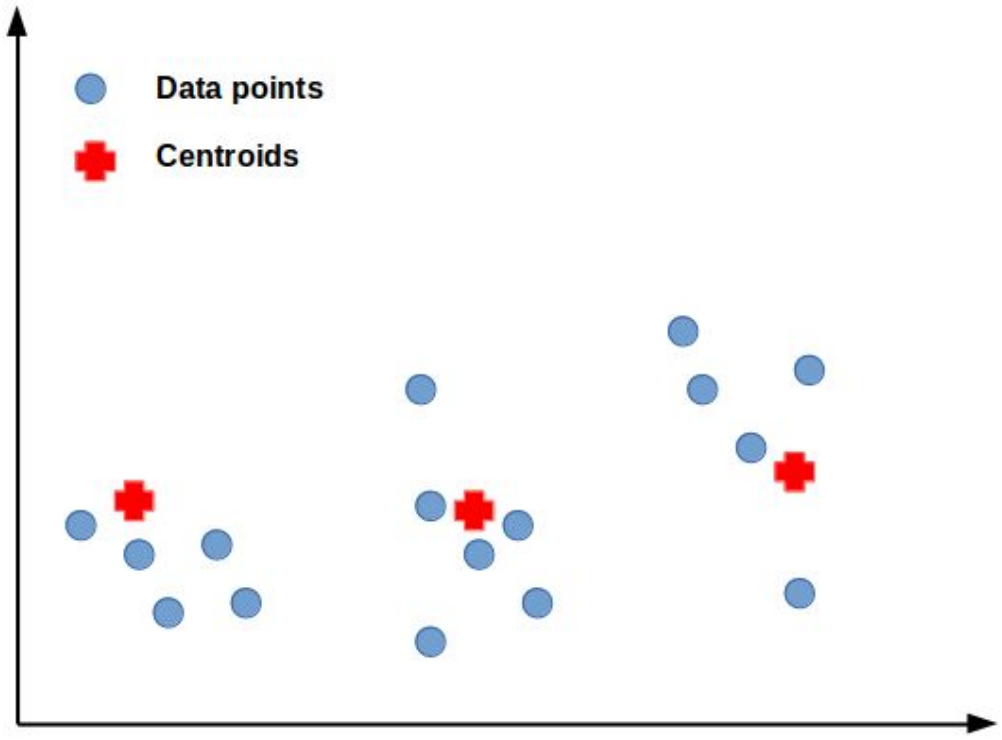
K-means algorithm is the unsupervised machine learning algorithm in which whole data is divided into K number of clusters. Every cluster has its centroid which is calculated by averaging the data points of that cluster. But what are the criteria of clustering?

Inertia: It is the measure of intra-cluster distances, which means how far away the datapoint is concerning its centroid. This indicates that data points in the same cluster should be well matched and similar to each other. For better clustering, the inertia value should be minimum. In contrast, if the inertia value is high, that means data points in the cluster are not similar to each other. This indicates a simple concept that for a given datapoint intra-cluster distance should always be less than inter-cluster distance.

How does K – means algorithm work?

- Initialize ‘ K’ and centroid values.
- Assign data points to the closest clusters, by calculating the Euclidean distance.

- When the clusters are formed, recompute their centroid values by calculating the average of data points.
- Repeat steps 2 & 3 until all the clusters are stable.



Performance of the Model

Since we need to divide the data as fake and real we set the cluster centers as 2.

kmeans = KMeans(n_clusters=2, verbose=1). Then we fit the model and predict the clusters. I obtained the following accuracy scores:

Model Accuracy with Word2Vec tuning:
Accuracy : 0.6691397000789266
Precision: 0.7001862197392924

Recall: 0.5928729107537054
F1-Score: 0.6420765027322405

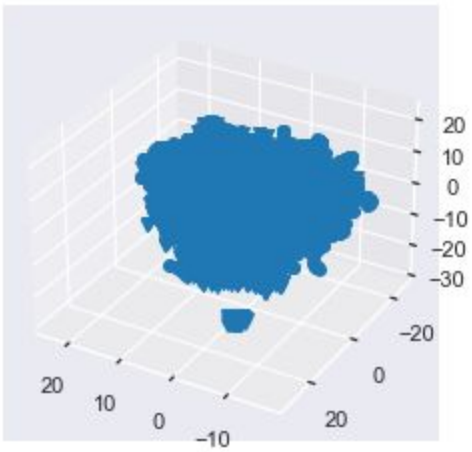
Model Accuracy with TFIDF tuning:
Accuracy : 0.6173638516179952
Precision: 0.7426900584795322
Recall: 0.36045411542100286
F1-Score: 0.4853503184713376

II. DBSCAN Clustering -Density-Based Spatial Clustering of Applications with Noise

DBSCAN is a clustering algorithm that defines clusters as continuous regions of high density and works well if all the clusters are dense enough and well separated by low-density regions.

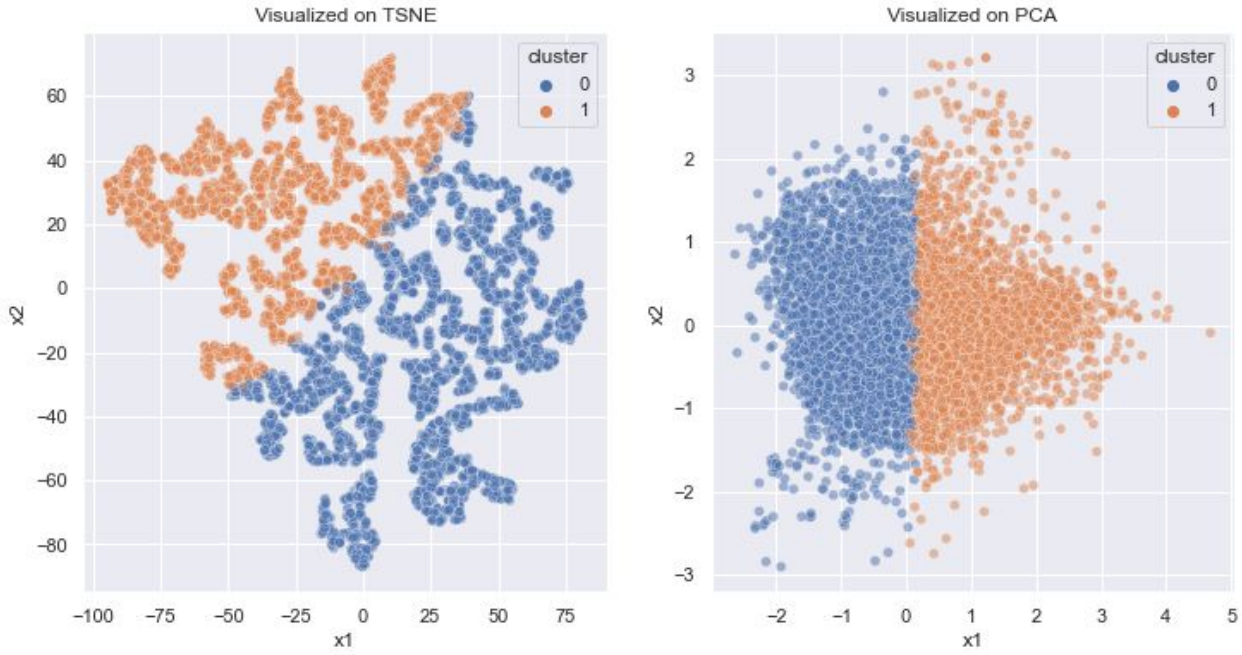
In the case of DBSCAN, instead of guessing the number of clusters, will define two hyperparameters: *epsilon* and *minPoints* to arrive at clusters.

1. ***Epsilon (ε)*:** A distance measure that will be used to locate the points/to check the density in the neighbourhood of any point.
2. ***minPoints(n)*:** The minimum number of points (a threshold) clustered together for a region to be considered dense. (mygreatlearning.com)



Visualizing The Results

I used TSNE and PCA to visualize the results of K-means Clustering.



Comparing the results

I compare the performance of K-Means Clustering and DBSCAN and a supervised Learning method PassiveAggresiveClassifier model:

	Accuracy
K-Means	0.67
DBSCAN	0.4994
PAC	0.93

Conclusion

I observed that the supervised model PAC made a better classification of the data against the unsupervised models K means and DBSCAN Clusterings.

One of the reasons for the bad performance of the unsupervised models might be unbalanced dataset.

Among the unsupervised models K means performs better on the DBSCAN Clustering.