

ANALYSIS OF CLUSTERING ALGORITHMS FOR TEXT CLASSIFICATION

A DISSERTATION

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE
OF

MASTER OF TECHNOLOGY
IN
INFORMATION SYSTEMS

Submitted By:

Anshula Raj

2K19/ISY/18

Under the supervision of

Dr. Seba Susan



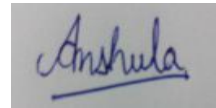
DEPARTMENT OF INFORMATION TECHNOLOGY

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

JULY, 2021

CANDIDATE'S DECLARATION

I, Anshula Raj, Roll No. 2K19/ISY/19 student of M.Tech (Information Systems), hereby declare that the project Dissertation titled “Analysis of Clustering Algorithms For Text Classification” which is submitted by me to the Department of Information Technology, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.



Place : New Delhi

Date : 24th July, 2021

Anshula Raj

(2K19/ISY/18)

CERTIFICATE

I hereby certify that the Project Dissertation titled “Analysis of Clustering Algorithms For Text Classification” which is submitted by Anshula Raj, Roll No 2K19/ISY/18 Information Technology, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.



Place : New Delhi

Date : 24th July, 2021

Prof. Seba Susan

SUPERVISOR

Professor

ACKNOWLEDGEMENT

I am very thankful to Dr. Seba Susan (Professor, Department of Information Technology) for the excellent guidance provided by her related to the topic without which this project would not be possible. My guide helped me by guiding me with ideas from time to time. The regular discussions with ma'am and her support made this project possible.

I would also like to express my gratitude to the university for providing us with the laboratories, infrastructure, testing facilities and environment which allowed us to work without any obstructions. Also, I am grateful to the university for providing us access online for our research work.

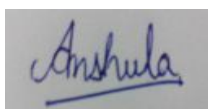
Anshula Raj

Roll No. 2K19/ISY/18

ABSTRACT

Large datasets are trending in today's world where data is generated at a swift rate every day and this data will not be of much use until some meaningful information can be obtained. Lots of analysis is done on the data and conclusions are drawn. Different methods are applied on the data after processing like, clustering, classification, regression etc. In this project, we worked on clustering on large dataset which was a text dataset called 20NewsGroups. We implemented different unsupervised clustering algorithms in Python which were K-means, fuzzy c-means, fuzzy co-clustering of documents and keywords, agglomerative clustering and density-based spatial clustering of applications with noise. We run the algorithm on the test dataset consisting of three newsgroups (rec.sport.baseball, sci.space, alt.atheism) and noted the result. We measured accuracy and F1 score.

We found out that fuzzy co-clustering of documents and keywords worked best followed by fuzzy c-means. Most ineffectual clustering algorithm for this dataset was DBSCAN. Our conclusion was that for such a large text document most effective algorithm would be the one where fuzzy concept is used because in text documents both the keywords and the individual documents association needs to be taken care of.

A rectangular box containing a handwritten signature in blue ink that reads "Anshula".

ANSHULA RAJ

2K19/ISY/18

LIST OF TABLES

TABLE NO.	TABLE NAME	PAGE NO.
3.1	Distribution of test and training data of 20NewsGroups	24
3.2	Parameters and termination condition of various algorithms	33

LIST OF FIGURES

FIGURE NO.	FIGURE NAME	PAGE NO.
1.1	Data growth over the years	1
1.2	Decision Tree	3
1.3	K-Nearest Neighbor	4
1.4	Naïve Bayes Classifier	5
1.5	Hadoop Ecosystem	7
1.6	Structured Dataset	9
1.7	Unstructured Dataset	9
1.8	Semi-structured Dataset	10
1.9	Supervised Learning	10
1.10	Reinforcement learning	11
2.1	Points after clustering	12
2.2	Document classification	14
2.3	Document clustering	14
2.4	Image segmentation on eye	16
2.5	Recognizing the objects	16
2.6	Partitional Clustering	17
2.7	Hierarchical clustering	18
2.8	Density-based clustering	19
2.9	Grid-based clustering	19
3.1	Sample article from alt.atheism category containing header and quotes	25
3.2	Sample article from alt.atheism category containing footer	26
3.3	Pre-processing steps performed on the dataset	29
4.1	Accuracy obtained after running algorithms on 20NewsGroups dataset	34
4.2	F1-Score obtained after running algorithms on 20NewsGroups dataset	35
4.3	Running time of the algorithms	35
4.4	K-means result	36
4.5	FCM result	36
4.6	FCoDoK result	36
4.7	Agglomerative clustering result	37
4.8	DBSCAN Result	37

CONTENTS

1.	CANDIDATES DECLARATION	ii
2.	CERTIFICATE	iii
3.	ACKNOWLEDGEMENT	iv
4.	ABSTRACT	v
5.	LIST OF TABLES	vi
6.	LIST OF FIGURES	vii
7.	Chapter 1: INTRODUCTION	1
	1.1 What is large dataset	1
	1.2 Commonly used algorithms for handling large data	3
	1.3 Problems with large dataset	5
	1.4 Tools used for analysing large dataset	6
	1.5 Types of data format	8
	1.6 Types of learning	10
8.	Chapter 2: LITERATURE REVIEW	12
	2.1 What is clustering	12
	2.2 Applications of clustering	14
	2.3 Clustering techniques	17
	2.4 What is text classification	20
	2.5 Research on text classification	20

9.	Chapter 3: RESEARCH METHODOLOGY	23
	3.1 Setup	23
	3.2 About the dataset	23
	3.3 Proposed methodology	26
	3.3.1 Pre-processing steps	27
	3.3.2 Algorithms implemented	29
	3.3.3 Parameters set for the algorithms	32
10.	Chapter 4: RESULTS	34
11.	Chapter 5: CONCLUSION	38
12.	REFERENCES	39
13.	LIST OF PUBLICATIONS BY THE STUDENT	45

CHAPTER 1 INTRODUCTION

1.1 WHAT IS LARGE DATASET

Rapid growth of technology and internet has led to rapid increment in the amount of data produced online. Now this data can be of high dimension and that could make the overall structure of the data complex. This is large dataset; a dataset which is of large size and thus high dimension. A simple database management system can be ineligible to handle such datasets [1]. As shown in the figure below, the rate at which the data is growing over the years; unstructured data will be growing at a faster rate.

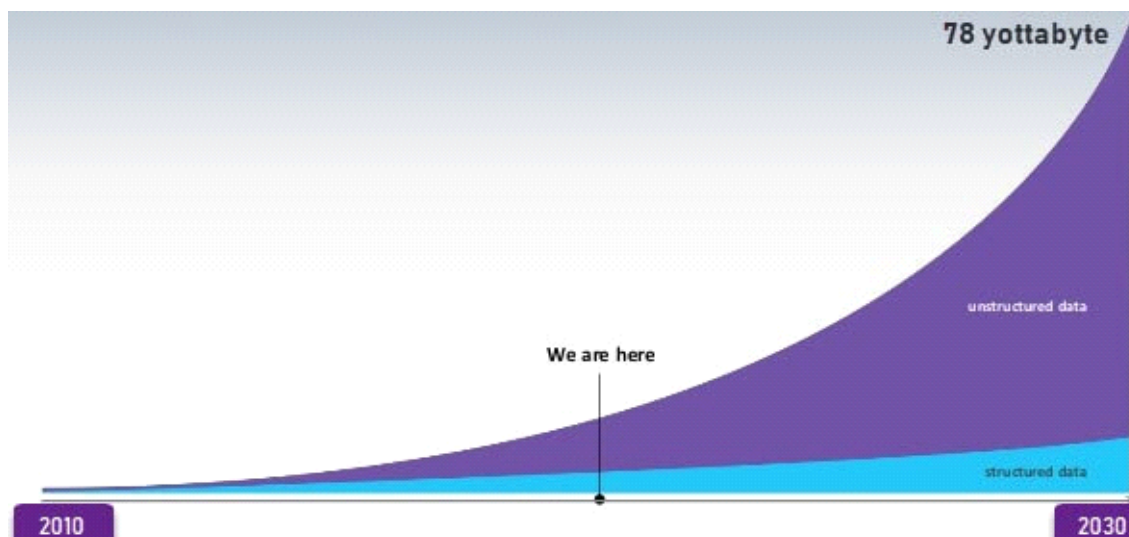


Fig 1.1: Data growth over the years

Now, researchers are trying to make sense of the large data. Research is being done in order to get some information out of those datasets. The main aim of analysis on those large datasets is to get information that can give some solid and meaningful conclusion. The conclusions should be helpful in solving problems related to various fields or even alleviate user experience. Problems such as face recognition, intrusion detection system, prediction of bugs in a software, recommendation system and many more [2] can be done by analyzing and performing some algorithms on the dataset.

Some of the sources of large dataset are [3] -

1. Social media : Data generated here is growing every day. It is in an unstructured format and is more challenging to process. But, lots of information can be concluded after applying clustering techniques on them. Even machine learning applied on this data can give information about trends, forecast, predictions etc.
2. Old data : There are so many datasets available on the internet which are very old but, even today they are used for analysis and for improving the current algorithms. Like Reuters-21578 [4] is one such dataset which consists of news articles. It has around 10000 documents. It is also unstructured and has to be structured accordingly for application. Even 20NewsGroups dataset [5] consists of newsgroup documents of twenty groups. It was collected from different newsgroups and it consists of around 18000 articles.
3. Data obtained from transactions : There is so much transactions going on online through laptop, mobile, apps that this structured information is also stored for technical analysis.
4. Images : Images have a wide application in medical field. X-rays, sensors, videos, satellite images etc. are all sources of images and they are also used for analysis like prediction of heart diseases, how the area has changed, face detection etc.

Importance of large dataset –

As said earlier, large dataset has the capability to give some information or conclusion. But it depends on how it is mined, it is processed and how the algorithm is applied on it. Every data generated holds importance in large dataset because it has features. If the large dataset is used effectively then, industries can maybe improve user experience, mitigate risks like in banks, increase sales, improving products, improving a software based on bug prediction, make predictions like about stock market or in medical field.

Applications of large dataset –

Large dataset is used for classification, clustering and for predicting something. Like in banks, it is used for predicting stock market using machine learning. Banks collect a lot of information like transactions, reviews, investment information and they can make use of this information. Suppose the dataset has a trend and if there is a slight variation in the trend then it can be analyzed for cause and decisions.

Another behavior that can be analyzed is of customer in all fields like retail, online shopping, financial sector etc. Customers can be classified or segmented in groups. Customer information can be used to observe the trends and make marketing forecasts. Even campaigns and strategies can be planned out depending on what useful result we got from the data. For example, online sites can approach those customers of the cluster who shop products same as present at their site.

Traffic analysis [6] can also be done on traffic dataset to predict traffic jams in an area and thus, solutions can be come up for curbing this problem.

1.2 COMMONLY USED ALGORITHMS FOR HANDLING LARGE DATA

For handling large data, there are some algorithms which are used a lot and they are –

1. CART (Classification and Regression Trees) [7] :

This algorithm creates a binary tree which divides the decision tree into two groups. Decision tree is a structured tree where the attributes are split using measures like, information gain, gini index etc. Every node is a feature in the decision tree. Branch tells the value. Gini index is used for choosing an attribute. Smallest gini index attribute will be divided. This process terminates when we find that an appropriate tree is constructed. This algorithm does not have much calculations and hence its an easy algorithm. But complexity increases if the problem space is big. This fall in the category of supervised learning.

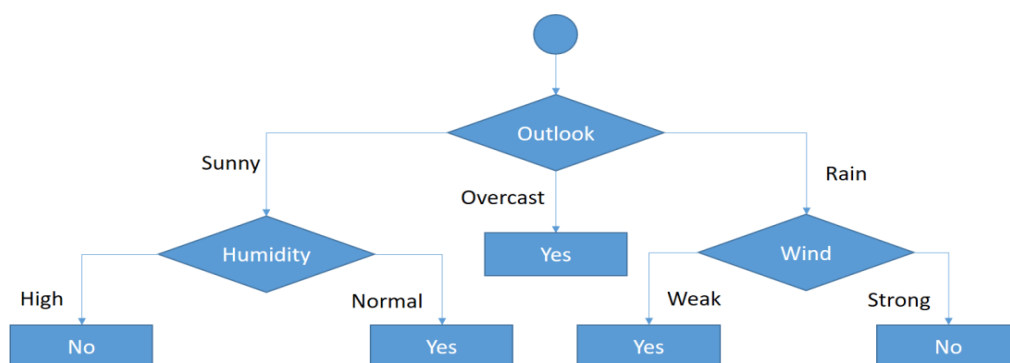


Fig 1.2 Decision Tree

2. K-Nearest Neighbor Algorithm :

It is a supervised algorithm. With an easy implementation, what it does is depending on 'K', the clusters are formed. For one point, how many other K points are in close vicinity. The distance is calculated from all points and the K nearest points are chosen and those points form a group.

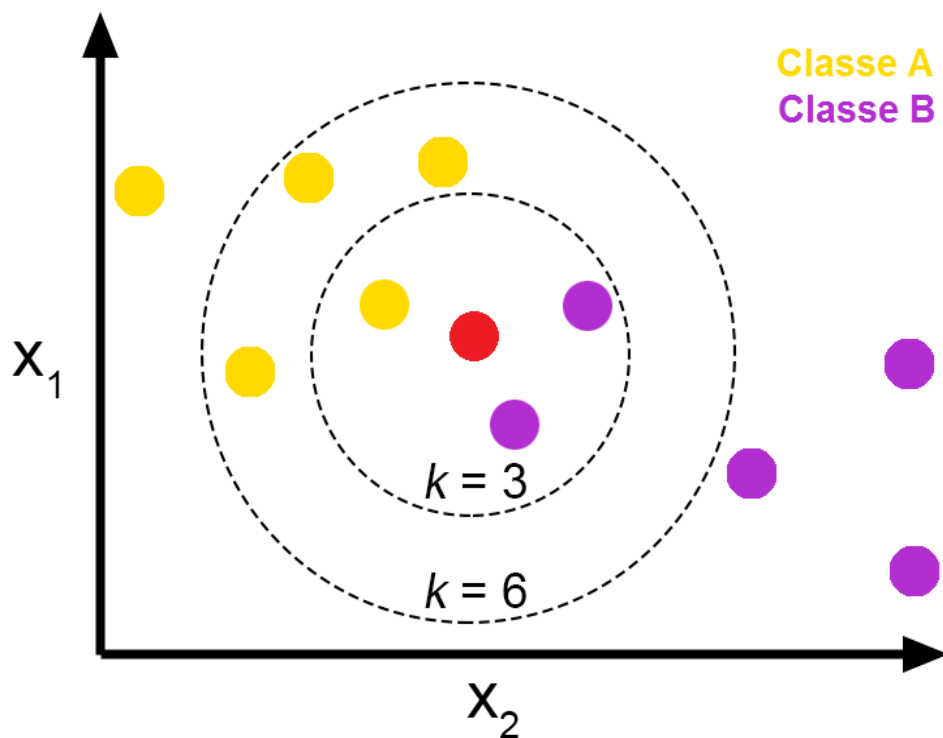


Fig 1.3 : K-Nearest Neighbor

3. Naïve Bayes Classifier [8] :

This is used for prediction. Using probability, it classifies the data. This is completely based on Bayes's theorem. Naïve Bayes is based on conditional probability. On all the classes, probability is calculated. Naïve Bayes classifier works on the fact that we already have seen some data and based on those data probability has been calculated. So, when we see a new data, then where that data will get classified. So, it classifies the data based on what it already has seen. This is supervised learning.

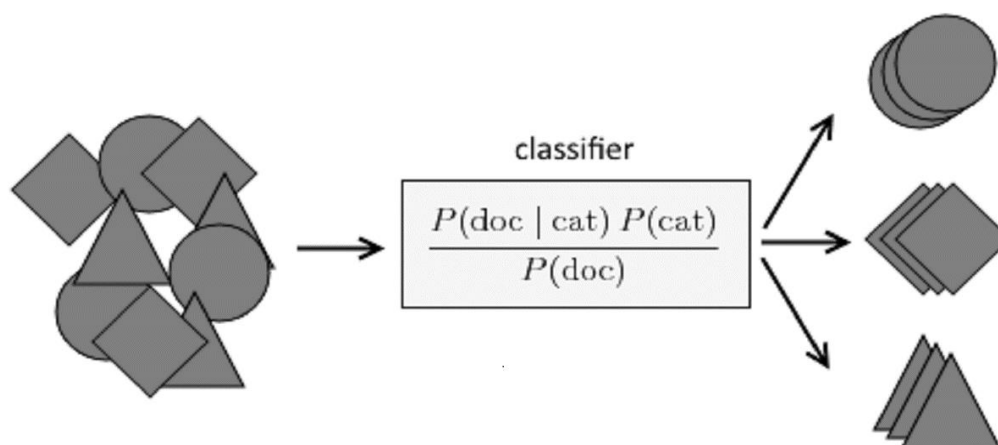


Fig 1.4 : Naïve Bayes Classifier

4. Chameleon [9] :

This is a hierarchical clustering method. This algorithm starts off by grouping the data points into small sub-groups. The sub-groups consist of points which are closely linked. Then, agglomerative clustering is used to join the groups one-by-one until no longer merging can be done. Since this is hierarchical method so, it takes a lot of time to run on large dataset. In this, input is given as adjacency matrix of the dataset. Then, using k-nearest neighbor a graph is constructed of the data points. Then, on this graph division is performed using some graph partitioning algorithm to give sub-groups. Then, sub-groups is merged by taking care of likeness of the group data points. Until it is not possible to merge anymore, merging continues.

1.3 PROBLEMS WITH LARGE DATASET

There are many algorithms for handling the large data that is algorithms to get information out them. Machine learning techniques are also applied to reduce dimensions of large data.

1. High dimension is one of the primary concerns of large data because high dimensions cause the algorithm to run for more time and sometimes low accuracy too. What researchers want is to reduce the dimension such that only essential features remain in the large dataset.

2. Also the data can be heterogenous so, processing it would require more efforts. Since the algorithms work on data that are similar in structure, making the data points similar is a must [10].
3. Since the large dataset can have large amount of size so storing it can be difficult. Storage plans needs to be thought out.
4. Some data may be missing from the original large dataset like missing an entire row of record or just some values so, then there are algorithms to predict the missing value too. But it does reduce the quality and integrity of the dataset [11].
5. Sometimes the algorithm does not even complete its run since the data is so large. It may take a lot of time and not even give satisfactory accuracy.
6. Data preprocessing can be a challenge if the data has a lot of noise, errors, modified data. This can happen for structured or non-structured or semi-structured data [12].
7. Dataset could be stored somewhere else and then the challenge arises to transport it to a data center.
8. Let's say unstructured data is integrated form different sources to create some dataset. Then, it becomes a problem because since it is unstructured the format could be different and there could be repetition of values. So, this is data integration problem.

1.4 TOOLS USED FOR ANALYSING LARGE DATASET

1. Hadoop [13] :

It is an open-source software framework for distributed storage of very large datasets on computer clusters. What happens in Hadoop is, it provides lots of clusters because one computer cannot store entire such huge amount of data so, data is distributed among the computers and processing is done using various tool like Spark, Mahout, MapReduce which provides inbuilt libraries for clustering too. It was developed by Apache. Java architecture framework supports its running.

Map Reduce is composed of two terms ; Map and Reduce. It works on divide and conquer. It is a processing technique for the clusters. The user defines a data operation such as a query and the platform “maps” the operations across all relevant nodes for distributed parallel processing. During Map stage, the input data is processed. Input is in

the form of file or directory and is stored in HDFS. Input file is passed to the mapper function line by line. Mapper processed the data and creates small chunks of data. In Reduce stage, the chunks are processed, and a new set of output is produced which is stored in HDFS. Its very fast as compared to traditional relational database.

HIVE; is used to bridge the gap of SQL i.e it helps to run queries against hadoop cluster. It was developed by Facebook.

PIG is similar to HIVE but, it uses 'Perl-like' language for querying. It was developed by Yahoo.

Mahout is a data mining framework and it mainly used for creating ML algorithms such as classification, clustering, recommendation. The algorithms are written on top of Hadoop.

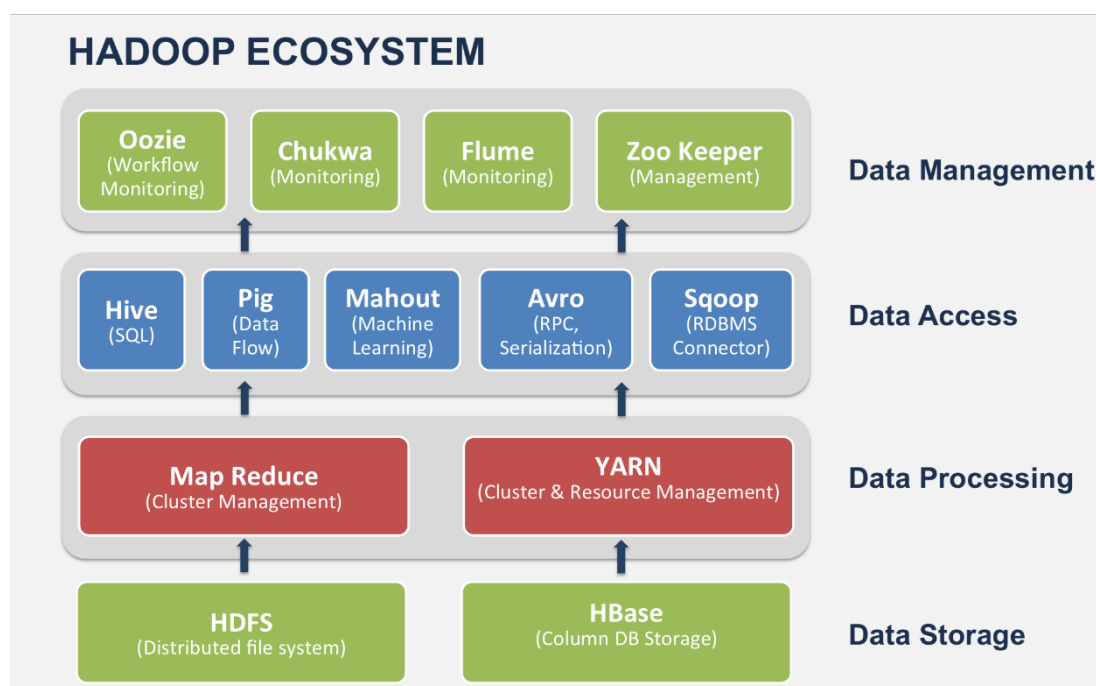


Fig 1.5 Hadoop Ecosystem

2. Python :

It is an open-source language. It's quite easy to code and it provides plethora of inbuilt library, inbuilt data structures and functions for many functions. Functions that can do data analysis and manipulation are all provided by the inbuilt libraries. Even dataset is available in the libraries like iris dataset, 20NewsGroups, American National Election Studies of 1996, breast cancer data etc. Its fast. It is being widely used for large dataset analysis. It is object oriented as well.

3. R :

It is an open-source language and an environment for statistical computing and graphics. Its used for data analysis. R provides functionalities ranging from linear modelling, classification and clustering etc. It was developed by Bell labs. R can effectively handle data and store it. It has great graphic properties for displaying graphical plots. It has inbuilt tools for analysis and also for operations on arrays. User can add their own functions. It has inbuilt dataset too like; iris dataset, tooth growth dataset, plant growth dataset etc. We can also perform machine learning tasks on data. It can run on various platforms like windows, linux etc.

4. Tableau :

It is a tool for data visualization. We can do data analysis with this tool too. It can take the data as input, create dashboards, reports, charts. It also data blending, real-time data analysis, predictive analysis. Not much coding background is required for this tool. Its an easy tool to operate. It can be used to discover patterns in the data. It can connect to both live data sources or extract data from external sources. It also supports SLQ, google analytics, Hadoop etc. The visualizations it can create are; pie chart, histogram, Gantt chart, bar chart, heat maps. If we want to perform some query on the data, then we can type our query naturally and it will give results.

1.5 TYPES OF DATA FORMATS

Data collected has various formats which are structured, unstructured and semi-structured [14].

Structured Data : It means the data collected has a fixed format i.e there are specific columns and every column has same type of data only. In relational database management system, records are stored in a tabular format, this is structured data. Structured data is more easier to comprehend and clean. Example of this is logs, spreadsheet data.

battery_level	c02_level	cca2	cca3	cn	device_id	device_name	humidity	ip
8	868	US	USA	United States	1	meter-gauge-1xbYRYcj	51	68.161.225.1
7	1473	NO	NOR	Norway	2	sensor-pad-2n2Pea	70	213.161.254.1
2	1556	IT	ITA	Italy	3	device-mac-36TWSKIT	44	88.36.5.1
6	1080	US	USA	United States	4	sensor-pad-4mzWkz	32	66.39.173.154
4	931	PH	PHL	Philippines	5	therm-stick-5gimpUrBB	62	203.82.41.9
3	1210	US	USA	United States	6	sensor-pad-6al7RTAobR	51	204.116.105.67
3	1129	CN	CHN	China	7	meter-gauge-7GeDoanM	26	220.173.179.1
0	1536	JP	JPN	Japan	8	sensor-pad-8xUD6pzsQI	35	210.173.177.1
3	807	JP	JPN	Japan	9	device-mac-9GcjZ2pw	85	118.23.68.227

Showing the first 1000 rows.

Fig 1.6 : Structured Dataset

Unstructured Data : It means the data has no fixed format. Unstructured data is more challenging to analyze due to its ‘not-fixed’ structure, but it is more promising of the fact that it can give some really good information output. Example; health records, images file, audio files, metadata etc.

For analyzing unstructured data, manual data analysis is required. We will have to process this format, extract only the useful information from it and then present it in a structured format. This format is messy and contains noise so, natural language processing techniques can be helpful for processing this data format.

bananas,2012,313.182,473,168,485,281,512,0,8,8,	carrots,2012,313.182,596,672,782,236.9,356.00,5.8,3,5.1
bananas,2013,315.427,408,334,497,234,349,0,8,8,	carrots,2013,315.427,565,401,819,245,360.00,5.8,3,5.1
bananas,2014,317.679,563,300,563,241,553,0,8,8,	carrots,2014,317.679,412,319,278,309.1,661.00,5.8,3,5.1
bananas,2015,316.119,468,251,547,223,632,0,8,8,	carrots,2015,316.119,609,582,571,351.6,625.00,5.8,3,5.1
bananas,2016,321.464,605,251,554,177,516,0,8,8,	carrots,2016,321.464,772,356,433,330.6,492.00,5.8,3,5.1
bananas,2017,323.151,415,111,353,194,406,0,8,8,	carrots,2017,323.151,498,389,586,283.9,660.00,5.8,3,5.1
mangoes,1970,203.849,363,240,411,172,602,0.034,5,14.5,	carrots,2014,317.679,608,604,807,143.3,749.00,5.8,3,5.1
mangoes,1971,206.466,622,141,397,200,527,0.04,5,14.5,	carrots,2015,316.119,597,636,629,134.4,352.00,5.8,3,5.1
mangoes,1972,208.917,425,200,608,310,371,0.039,5,14.5,	carrots,2016,321.464,526,579,619,160.6,940.00,5.8,3,5.1
mangoes,1973,210.985,471,145,587,133,627,0.051,5,14.5,	carrots,2017,323.151,494,615,511,174.1,446.00,5.8,3,5.1
mangoes,1974,212.932,432,258,385,165,564,0.059,5,14.5,	garlic,1970,203.849,565,358,759,NA,694.00,28,19,7.4
mangoes,1975,214.931,629,246,390,121,335,0.075,5,14.5,	garlic,1971,206.466,466,618,612,NA,227.00,28,19,7.4
mangoes,1976,217.095,632,218,591,122,616,0.087,5,14.5,	garlic,1972,208.917,284,334,261,1.7,825.00,28,19,7.4
mangoes,1977,219.179,630,333,372,245,441,0.068,5,14.5,	garlic,1973,210.985,774,614,708,1.4,641.00,28,19,7.4
mangoes,1978,221.477,442,177,522,220,623,0.094,5,14.5,	garlic,1974,212.932,645,668,277,1.4,402.00,28,19,7.4
mangoes,1979,223.865,378,275,628,279,384,0.095,5,14.5,	garlic,1975,214.931,576,548,614,3.7,949.00,28,19,7.4
mangoes,1980,226.451,446,180,368,262,576,0.114,5,14.5,	garlic,1976,217.095,390,359,356,5.1,458.00,28,19,7.4
mangoes,1981,228.937,397,257,389,193,390,0.091,5,14.5,	garlic,1977,219.179,314,387,298,6.6,559.00,28,19,7.4
mangoes,1982,231.157,500,119,566,205,333,0.132,5,14.5,	garlic,1978,221.477,516,339,686,42.8,258.00,28,19,7.4
mangoes,1983,233.322,642,151,643,160,647,0.197,5,14.5,	garlic,1979,223.865,337,323,341,22.8,830.00,28,19,7.4
mangoes,1984,235.385,437,165,366,200,462,0.196,5,14.5,	garlic,1980,226.451,341,318,780,22.1,831.00,28,19,7.4
mangoes,1985,237.468,463,251,381,123,368,0.194,5,14.5,	garlic,1981,228.937,477,676,781,30.1,528.00,28,19,7.4
mangoes,1986,239.638,629,118,376,330,486,0.221,5,14.5,	garlic,1982,231.157,631,634,376,24.9,615.00,28,19,7.4
mangoes,1987,241.784,588,158,424,153,402,0.253,5,14.5,	garlic,1983,233.322,405,589,502,26.3,841.00,28,19,7.4
mangoes,1988,243.981,606,346,401,350,604,0.171,5,14.5,	garlic,1984,235.385,650,450,584,27.7,507.00,28,19,7.4
mangoes,1989,246.224,398,189,628,236,524,0.232,5,14.5,	garlic,1985,237.468,739,569,449,20,346.00,28,19,7.4
mangoes,1990,248.659,390,263,457,266,372,0.244,5,14.5,	garlic,1986,239.638,305,580,843,35.7,951.00,28,19,7.4
mangoes,1991,251.889,434,291,560,180,615,0.388,5,14.5,	garlic,1987,241.784,394,370,465,25,389.00,28,19,7.4
mangoes,1992,255.214,570,223,469,138,402,0.306,5,14.5,	garlic,1988,243.981,552,638,449,32.4,428.00,28,19,7.4
mangoes,1993,258.679,547,116,399,261,641,0.405,5,14.5,	garlic,1989,246.224,421,432,722,50.4,298.00,28,19,7.4

Fig 1.7 : Unstructured Dataset

Semi-structured Data : It can contain both forms of data. The data is in a structured format but, still it's not actually defined. There is no clear definition since the structure is an indirect format. Example; an XML file, JSON format. Since it is semi-structured

there are tools available to read and analyze this data. So, the need for preprocessing might not be required.

```
<dataset id="h1">
  <dataentry id="0">
    <time>422875</time>
  </dataentry>
  <dataentry id="1">
    <time>278522</time>
  </dataentry>
</dataset>
```

Fig 1.8 : Semi-structured Dataset

1.6 TYPES OF LEARNING

There are 3 types of learning -

1. Supervised Learning :

In this, the system is trained with the examples of input-output (training set) and the goal of the system is to classify the input as one of the outputs as correctly as possible. In this learning; SVM, Decision trees, Neural network, Bayesian networks exist. Regression problems and classification problems can be solved using supervised learning. In classification, an input is given and based on the model it classifies the input into one of the outputs of the labeled observations. In regression, we have to predict the output of a given input.

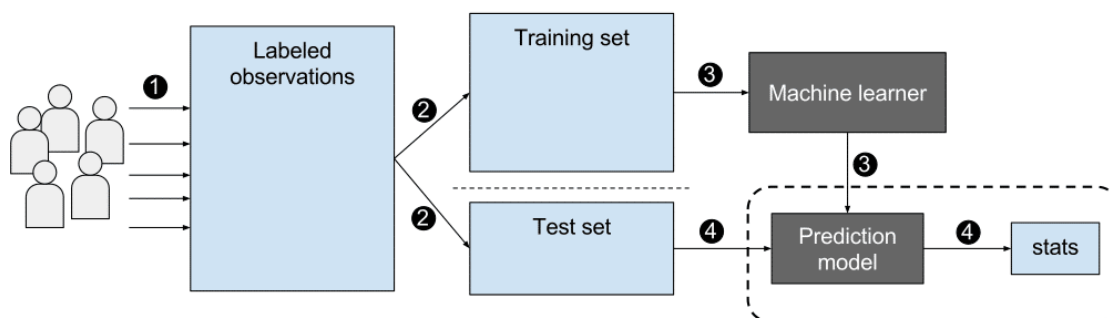


Fig 1.9 : Supervised Learning

2. Unsupervised Learning :

In this, the system is not provided with output. The system has to present an output given an input and the output is not something explicit. In this learning; k-means, hierarchical clustering, neural network are there. Example; in clustering, we cluster the similar data into one cluster by using some metric for similarity. Neural network is provided with some unlabeled dataset and neural network studies the pattern in that. Example; neural network is provided with lots of images of an animal. It will study all the images and adjusts its parameters (weight) again and again. After studying the pattern, it is given an input then neural network will classify whether the input is animal or not. This is unsupervised since no output is given for the training dataset. It is the job of neural network to identify the pattern in the training dataset and learn from it.

3. Reinforcement Learning :

In this the system learns by evaluating its environment. The system scopes out the environment, and gets feedback (as reward or punishment), and thus the system learns.

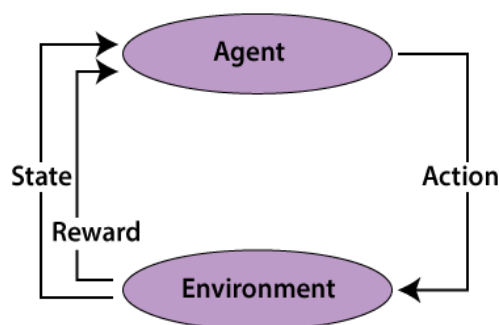


Fig 1.10 : Reinforcement learning

CHAPTER 2 LITERATURE REVIEW

2.1 WHAT IS CLUSTERING

Clustering is most popular unsupervised technique for finding the clusters of similar data. What it does is; it forms clusters of the input dataset according to some pre-set criteria in the algorithm [15]. Depending on the algorithm's effectiveness and data preprocessing, the clustering gives results accordingly. It is also used in pattern recognition. A clustering algorithm may be suitable for one problem area while for other it may not be suitable.

To handle large datasets i.e to effectively analyze the dataset clustering (grouping) is required of that dataset.

Every year, researchers have been trying to improve the clustering algorithms for different practical applications or for solving some problem. They either come up with improvements in the existing algorithms or combining two existing algorithms.

Some examples of general clustering is; grouping of similar documents based on some similarity feature, grouping of users in a shopping database based on some buying patterns.

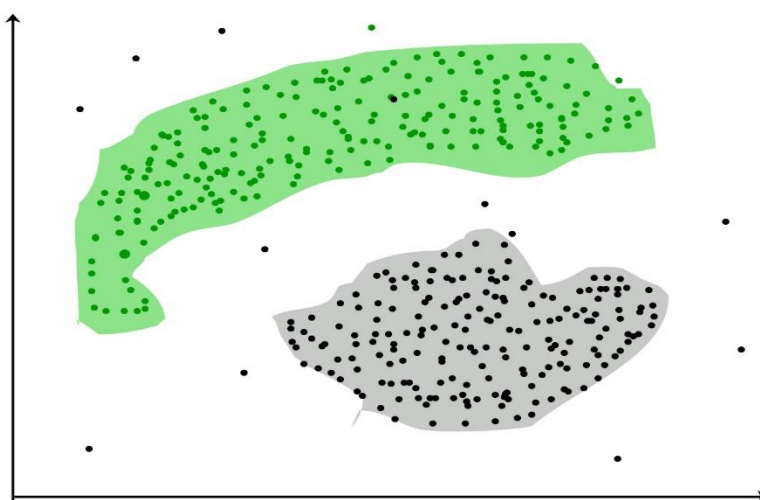


Fig 2.1 Points after clustering

What happens in clustering algorithm is-

1. Data points in a cluster are similar to each other.
2. Data points of different clusters are not that similar to each other.
3. There is some similarity measure taken as the standard for clustering. The similarity measure that is most popularly used is distance but, some other measures are; summation, angles or mean.

The steps of clustering are as follows –

1. Dimensionality reduction : This is called pre-processing the dataset. We reduce the dimension i.e the number of features of the dataset. Reducing features means to only select the important features of our dataset. This also leads to less running time of the algorithm.
2. Algorithm : Selecting a suitable clustering algorithm for our dataset in order to obtain good results is the next step.
3. Clustering done right or wrong : Next, we check whether the datapoints have been clustered correctly by seeing which group they have been allocated.
4. Observing the cluster : Next, the clustered results needs to be observed and interpreted means, what does the cluster tell us.

There are some clustering algorithms that can only handle numeric data or categorical data. But, for categorical applying a clustering algorithm which uses distance measure is challenging [16]. There are some datasets which are mixture of both like, medical dataset.

Now, one machine is not sufficient to handle large dataset clustering. Usage of distributed computing and parallel computing has increased. In this, data partitioning happens for scaling purpose and sampling happens for expediting the speed of the process [17].

What similarity measure is used, determines the success of clustering algorithm.

2.2 APPLICATIONS OF CLUSTERING

1. Document classification : For classifying that the document belongs to which category, clustering is used.

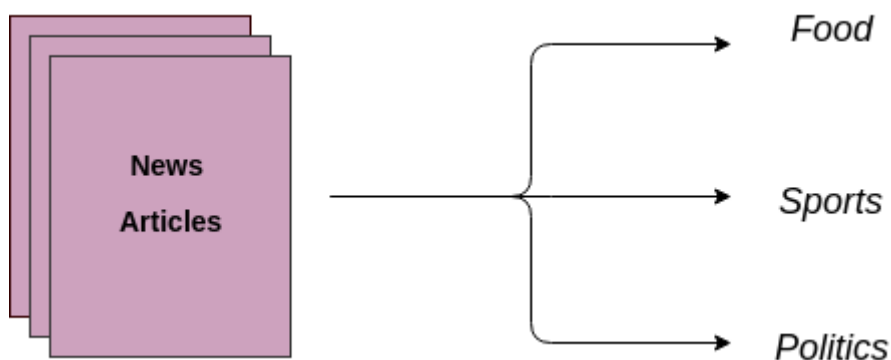


Fig 2.2 Document classification

2. Document organization : It is organizing the documents in hierarchy. Hierarchical order consists of categories. Document clustering also exists; similar documents are grouped together. It arranges the documents. Most popular algorithm used for this is k-means. Similarity in documents is based on sentence used in the documents. For checking similarity between documents; one function used is cosine function and other function is Euclidean distance [18].

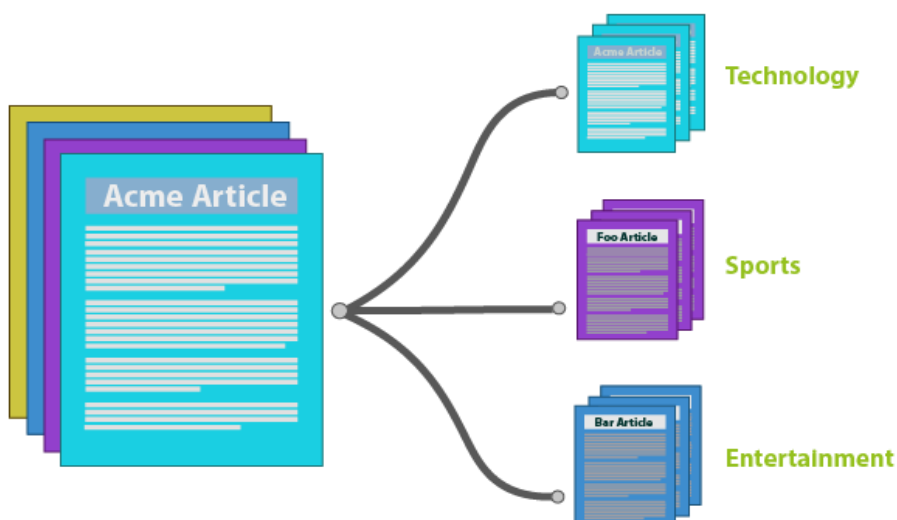


Fig 2.3 Document clustering

3. Intrusion Detection System : As the name suggests, this system detects intrusion. One algorithm as an example is FLAME (Fuzzy clustering by local approximation of memberships) [19] which performs clustering after partitioning the dataset. It is based on fuzzy clustering. The steps of this FLAME algorithm is;

- a. Neighborhood of every point is defined using KNN and a graph is constructed to connect all points.
- b. Fuzzy membership for local approximation is converged iteratively. Fuzzy value is calculated by using a linear combination of nearest neighbor.
- c. At the end, based on their membership value points are assigned to the clusters or outliers.

3. Image segmentation : An image also contains information. Its not just clustering that can be performed but, there are other algorithms that are performed on images. In image segmentation, an image is divided into different groups and those groups are identified as well i.e segmented. Similar pixels of the image get grouped together and a group is assigned some label, this is done by the algorithms. From an image, when we segment then we are extracting the useful images. Those segmented parts can be used for supervised or unsupervised learning. Most popular example of image segmentation is facial recognition system which can segment different faces and then recognize that face. One more example is biomedical image segmentation [20] in which the image of eye is segmented. The optic disk, tumor, retina vessel is segmented. These images can help in revealing problems in the eye like glaucoma, changes in optic disk, neuro disease, The images are collected and sometimes noise is added in them since the image might not be clear due to light attenuation. Bilateral filters are also applied for denoising the image. More filtering is required to restore spatial information loss. Earlier images were segmented using threshold-based, area-based segmentation etc. but complex images were difficult to segment. Fuzzy co-clustering method exists as well for image colour segmentation and it is more suitable for large dimensions [21]. Problem of outliers decreases but overlapping cluster problem still exists. One application of colour segmentation is detection of lesions in biomedical images using bacterial foraging algorithm for finding the best parameters for fuzzy co-clustering which is then applied on the image [22].

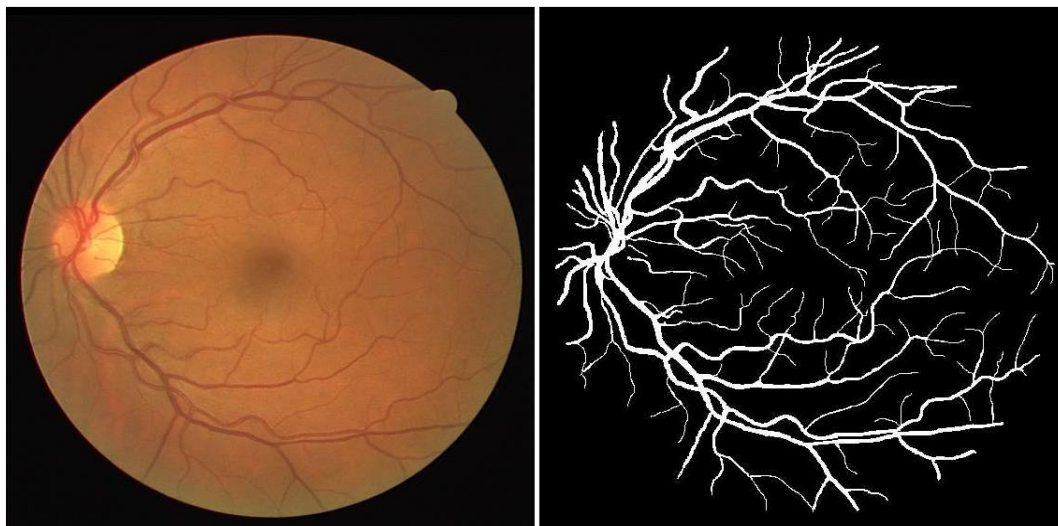


Fig 2.4 : Image segmentation on eye

4. Object recognition : This has wide applications in fields wherever objects need to be identified like; self driving cars, machine inspection, video investigation, tracking objects etc. Hierarchical clustering, k-means etc. are used for this application. But, these algorithms suffer from weaknesses like; high running time, sensitivity to initial centroids, non-convex clusters handling etc. Convex clustering can handle object detection [23]. There have been surveys to detect this. People have tried to make the objective function minimum in convex clustering for high dimensional data by separating parameters and filling missing data but, it increased running time. Another way people have come up with is by extracting colours and by changing colors to colour space and then doing convex clustering.

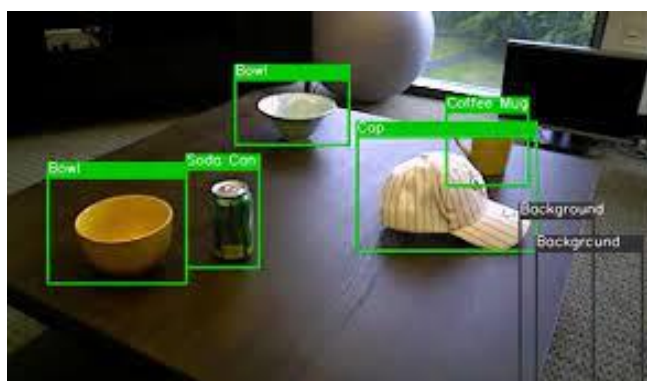


Fig 2.5 Recognizing the objects

2.3 CLUSTERING TECHNIQUES

We will list the standard unsupervised clustering methods used for all kinds of dataset –

1. Partitioning method : The aim of this clustering is to make groupings of the data points through optimization of the objective function resulting in improvement in the partitions iteratively [24]. This clustering converges quick and can differentiate between clusters of spherical shape. K-means, k-medoids, CLARA, PAM are examples of this clustering.

CLARANS [25] is Clustering Large Applications based on Randomized Search for clustering spatial data. It uses sampling technology along with partition around medoids. At every step there is randomness in the sample.

In K-medoids, how the cluster center is what makes it different from k-means. The cluster center will be from the dataset only and not average. This is called medoids. First, randomly k points are chosen and silhouette method evaluates how correct the chosen center is. Then data points are allocated their respective clusters based on the closest medoid. The point in the cluster which has minimum sum of distances from all points, that point is chosen as the new medoid. This process of assigning a point to a cluster and choosing medoid is repeated until medoids do not change.

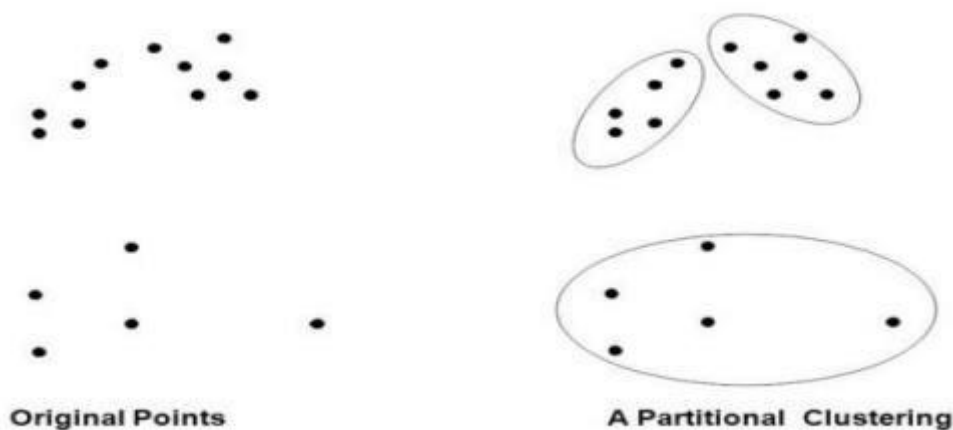


Fig 2.6 Partitional Clustering

2. Hierarchical method : In this, hierarchy of the data points is formed. Either it goes in up direction (bottom up) or down direction (top down). In bottom up, the points are combined again and again till one cluster is formed finally. In between, a threshold can

be put up and then the clustering can be stopped. Till the stoppage points, the clusters that are formed, it gives the final clusters. This is agglomerative clustering. In divisive clustering or top down, one full cluster is divided recursively until the desired number of clusters is obtained. Some hierarchical clustering methods are SLINK, BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies), DIANA (DIvisive ANALysis), CURE (Clustering Using Representatives) etc. SLINK works on bottom-up approach and single linkage clustering. DIANA uses Gini index for clustering. SLINK is very sensitive to noise and other hierarchical methods are not sensitive. SLINK can detect cluster of any shape but BIRCH and DIANA can detect convex shaped cluster. Hierarchical clustering time complexity is high.



Fig 2.7 Hierarchical clustering

3. Density-based method : The prime idea behind this is that if the density of adjacent area exceeds a threshold then the clustering continues which means, an area must have certain number of points. So, density based clustering could prove useful in excluding outliers or noise. DBSCAN, DENCLUE (DENSITY-based CLUstEring), OPTICS (Ordering points to identify the clustering structure) are example of this. Density based methods are not sensitive to noise. These methods are a little better than hierarchical clustering in terms of time. DENCLUE uses kernel density estimation. This finds the dense regions in the data points. A cluster is defined by a local maximum of the estimated density function.

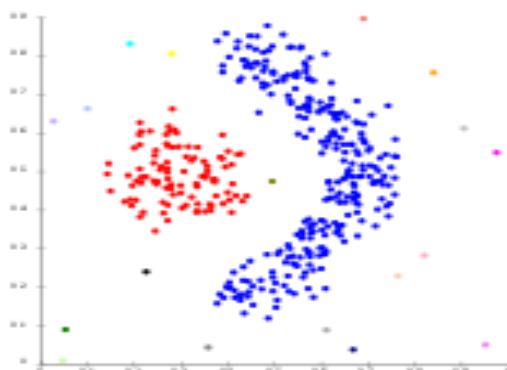


Fig 2.8 Density-based clustering

4. Grid-based method : In this, the space is presented as a grid and the cell numbers is fixed. This is a fast running method since the grid space is fixed and this method doesn't depend on the number of points. Process is run on these cells rather than all the data points, hence its fast [26]. Some grid based methods are; STING (Statistical Information Grid), OptiGrid (Optimal Grid clustering) etc. For every cell, mean, minimum, maximum i.e statistical information is calculated. Normal and exponential distribution is also calculated.

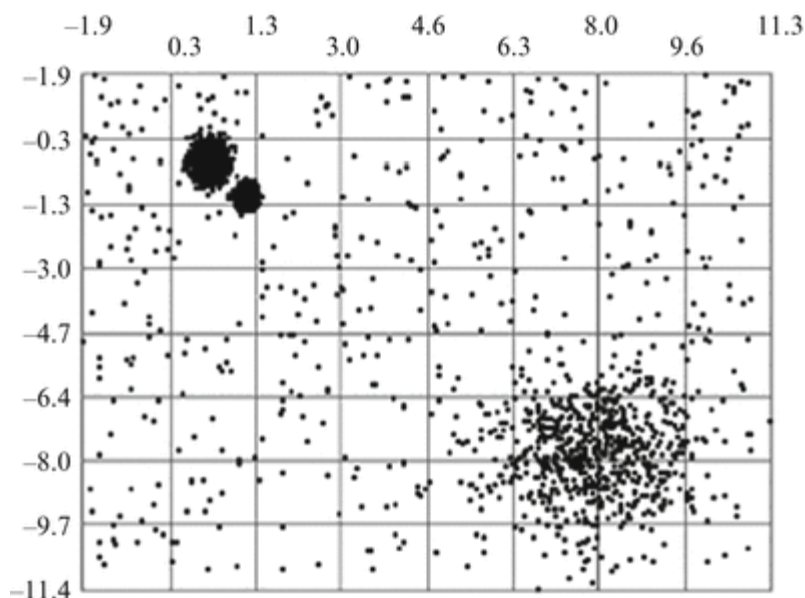


Fig 2.9 Grid-based clustering

2.4 WHAT IS TEXT CLASSIFICATION

Classification means to put into category. Text classification simply means to assign the category to which the text belongs. This is done by drawing out the features from the text given. But it is difficult to obtain only the relevant features from the text. To process textual information, machine learning methods are used. Some text classification examples are; spam detection, categorizing into search category, news category etc. Features of a text entity carries weight. Those features need to be selected which tells the relevance of the text. The relevancy will help in classification of that text. Thus, they need to be carefully selected [27].

If we are given a set of text and we want to classify them under a given set of categories, we can perform clustering. Similar text will be grouped together. Text info is unstructured dataset. If we want to apply clustering then, how to present the text; the text is represented as a vector. It can be presented as a binary or non-binary vector. The features form the vector. Vectors represent the content of the text. Text will give high dimensional vector. Semantics is ignored during clustering. A good clustering algorithm for text will be able to process the high number of features, can identify any shape of clusters and can handle noise [28]. Some algorithms require input parameters so, determining the best input parameters to get best results is a task. In the algorithm, what vector form is used (binary/non-binary) and what similarity measure is used like cosine distance or tf-idf (term frequency-inverse document frequency). In tf-idf, more the term is rarer less it occurs in the documents and those terms are given more weight because, they can help in recognizing the document's category [29]. For classification of text, there are methods such as; neural network, SVM, relevance feedback etc. There has been research on feature selection, ways to pre-process, assigning weights to the features etc. to make the algorithms more improved.

2.5 RESEARCH ON TEXT CLASSIFICATION

Some methods which have been implemented on text documents are –

1. Staged text Clustering Algorithm [30] :

This algorithm has two phase viz. splitting and merging. K-means is used in the splitting stage to divide all the text documents. Canopy clustering determines the value of k and the cluster center. For merging, hierarchical agglomeration algorithm is use to

merge all the clustered documents into cluster trees which are then further merged into one cluster tree. There is k cluster trees generated after agglomerative algorithm because it is applied on each of the k classes generated after canopy algorithm. Agglomeration further combines the k cluster trees into one. The closest cluster trees are merged recursively until the termination condition. The distance of all the points are found out and their average is taken, this has been used for see how similar two clusters are. F measure and purity was measured after this algorithm was run on Sogou corpus and this algorithm performed better than normal k-means and hierarchical clustering.

2. Parallel text clustering algorithm based on semantic tree [31] :

The author says that semantics are ignored during text clustering and the cause of lower precision is using the frequency of the terms for this. This algorithm got high precision in less time. Parallelism in processes is there to reduce time. There is a master process that partitions the data, sends the information, collects the information and clusters the result using semantic tree. Slave process takes care of word frequency stats. Using the statistical information, master process checks for cluster similarity.

3. Chi-square statistical algorithm [27] :

This algorithm uses chi-square to get the feature words. Chi-square finds the relevance between category 'k' and feature 'f'. If a category has high chi-square then, it means that the feature 'f' has more relevance. If chi-square is zero, then feature and category are independent. So, through chi-square, we get the relevant features which can then be used for different algorithm. The problem with this is; there can be relevant words with less frequency and then those words won't be considered after the end result.

4. Sentence based clustering [32] :

In this, first pre-processing is performed on the dataset. Then, every sentence is given a score using feature extraction. Then, Gensim Word2Vec generates vector for the document. Using Elbow method, value of 'k' for k-means is determined. Then, k-means is performed in which the sentences are distributed into the k clusters. The cluster which has the most sentences is containing the important sentences. This is further used for generating text summary.

5. Weighted BERT model [33] :

Three stages are there in this model; embedding module, weighting module, clustering module. BERT generates contextualized sentence embeddings. Then, weighting module gives weights to sentences. There are two schemes for weighting which gives sentence embeddings and these represent the documents. The clustering module is the last stage in which k-means is run and gives final output.

6. Choosing centroid using a fitness function [34] :

The author has proposed a fitness function to choose the clusters instead of random centroid selection. First, a topic modelling phase is run. The topic is chosen and then the collected documents are modelled on that topic. Pre-processing and tokenization is done. Then, k-means with genetic algorithm is applied after choosing centroids.

7. Combining K-means with dimension reduction [35] :

After the dataset is processed, tokenized and tf-idf weight is calculated for each term. Then, SVD (singular value decomposition) and NMF (non-negative matrix factorization) is done for reducing dimension. Then, k-means is run on the dataset.

8. Using document to vector model [36] :

In this, there are three steps for pre-processing viz. porter stemming, Lancaster stemmer and wordnet lemmatizer. Tf-idf and document to vector model is used to represent the document. What it does is it gives the words vector presentation along with maintaining semantic similarities. This decreases dimension.

CHAPTER 3 RESEARCH METHODOLOGY

3.1 SETUP

All algorithms have been written in Python on Spyder version 3.7. Python is provided by Anaconda. Anaconda is a free and open source software which provides Jupyter and Spyder for writing Python codes. We can install packages for Python through Anaconda prompt. PC used was Asus Vivobook 8th Gen i5 64 bit with 8GB RAM and 500GB hard drive. Operating system used was Windows 10. In-built scikit library of Python which contains 20Newsgroups dataset [37].

3.2 ABOUT THE DATASET

In this section, we will describe our textual dataset which we have used which is 20NewsGroups dataset [38]. This dataset contains approximately 18000 news articles. The reason why it's called 20NewsGroups is because every article in this dataset belongs to a category and there are twenty categories in this dataset. The categories are related to science, computers, atheism, sports etc. The 20 categories are;

1. alt.atheism
2. comp.graphics
3. comp.os.ms-windows.misc
4. comp.sys.ibm.pc.hardware
5. comp.sys.mac.hardware
6. comp.windows.x
7. misc.forsale
8. rec.autos
9. rec.motorcycles

10. rec.sport.baseball
11. rec.sport.hockey
12. sci.crypt
13. sci.electronics
14. sci.med
15. sci.space
16. soc.religion.christian
17. talk.politics.guns
18. talk.politics.mideast
19. talk.politics.misc
20. talk.religion.misc

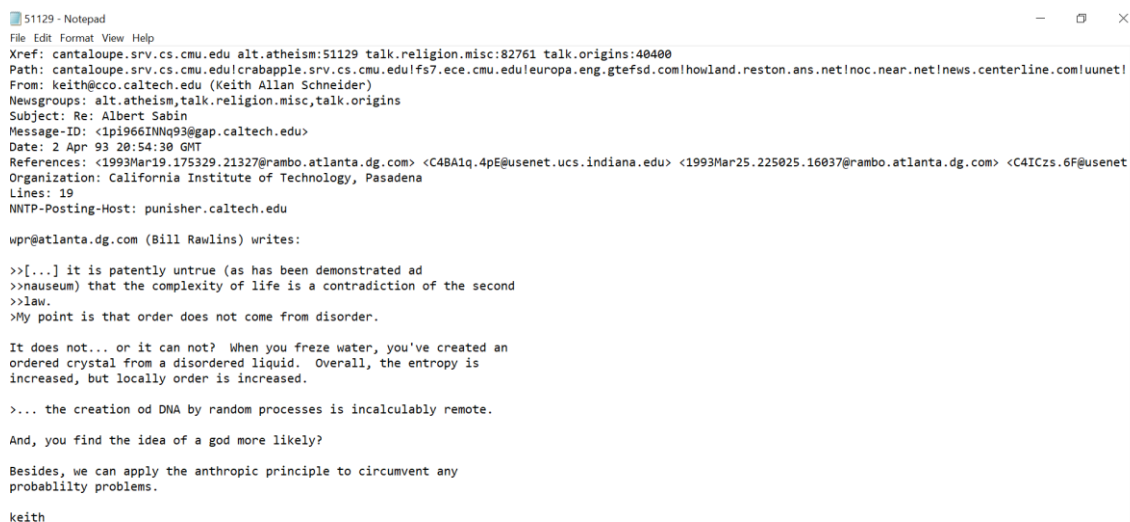
The dataset is divided into test and training data.

Table 3.1: Distribution of test and training data of 20NewsGroups

Category	No of documents in testing	No of documents in training	Total documents
comp.graphics	389	584	973
comp.os.ms-windows.misc	394	591	985
comp.sys.ibm.pc.hardware	392	590	982
comp.sys.mac.hardware	385	578	963
comp.windows.x	395	593	988
rec.autos	396	594	990
rec.motorcycles	398	598	996
rec.sport.baseball	397	597	994
rec.sport.hockey	399	600	999
sci.crypt	396	595	991

sci.electronics	393	591	984
sci.med	396	594	990
sci.space	394	593	987
misc.forsale	390	585	975
talk.politics.guns	364	546	910
talk.politics.mideast	376	564	940
talk.politics.misc	310	465	775
talk.religion.misc	251	377	628
alt.atheism	319	480	799
soc.religion.christian	398	599	997
Total	7532	11314	18846

Each of these categories contains many articles and every article has a header, footer and quotes section. There are 18000 articles and total of 71000 raw words. Test dataset consists of 7500 articles and training dataset contains 11000 articles.



```

51129 - Notepad
File Edit Format View Help
Xref: cantaloupe.srv.cs.cmu.edu alt.atheism:51129 talk.religion.misc:82761 talk.origins:40400
Path: cantaloupe.srv.cs.cmu.edu/crabapple.srv.cs.cmu.edu/ifs7.ece.cmu.edu/leuropa.eng.gtcfdsd.com/howland.reston.ans.net/inoc.near.net/news.centerline.com/luenet1
From: keith@cco.caltech.edu (Keith Allan Schneider)
Newsgroups: alt.atheism,talk.religion.misc,talk.origins
Subject: Re: Albert Sabin
Message-ID: <1p1966INNq93@gap.caltech.edu>
Date: 2 Apr 93 20:54:30 GMT
References: <1993Mar19.175329.21327@rambo.atlanta.dg.com> <C4BA1q.4pE@usenet.ucs.indiana.edu> <1993Mar25.225025.16037@rambo.atlanta.dg.com> <C4ICzs.6F@usenet
Organization: California Institute of Technology, Pasadena
Lines: 19
NNTP-Posting-Host: punisher.caltech.edu

wpr@atlanta.dg.com (Bill Rawlins) writes:

>>[...] it is patently untrue (as has been demonstrated ad
>>nauseum) that the complexity of life is a contradiction of the second
>>law.
>>My point is that order does not come from disorder.

It does not... or it can not? When you freeze water, you've created an
ordered crystal from a disordered liquid. Overall, the entropy is
increased, but locally order is increased.

... the creation of DNA by random processes is incalculably remote.

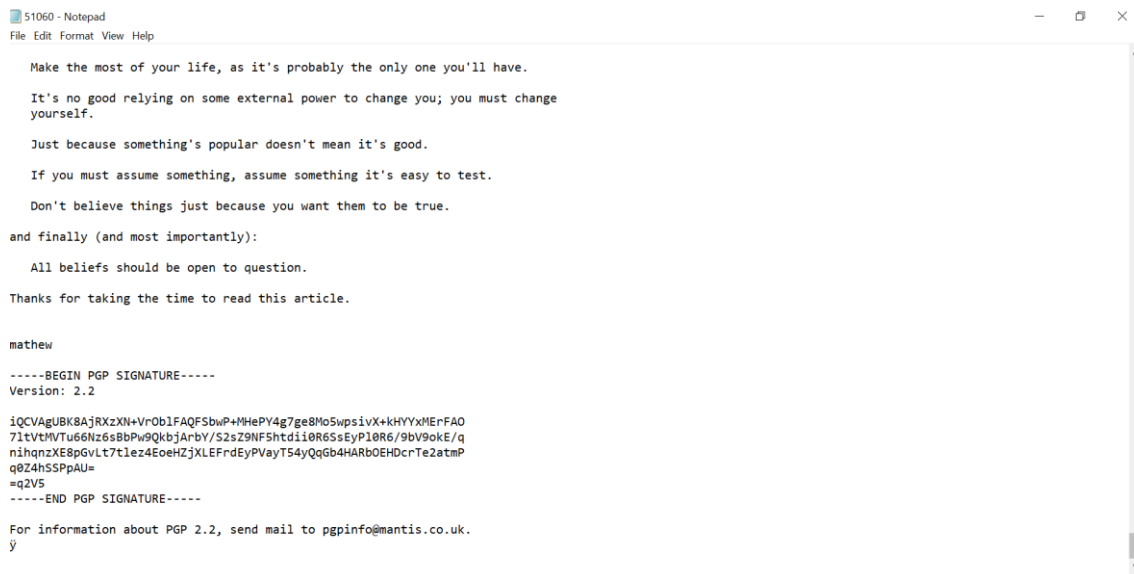
And, you find the idea of a god more likely?

Besides, we can apply the anthropic principle to circumvent any
probability problems.

keith

```

Fig 3.1 Sample article from alt.atheism category containing header and quotes



```

51060 - Notepad
File Edit Format View Help

Make the most of your life, as it's probably the only one you'll have.

It's no good relying on some external power to change you; you must change
yourself.

Just because something's popular doesn't mean it's good.

If you must assume something, assume something it's easy to test.

Don't believe things just because you want them to be true.

and finally (and most importantly):

All beliefs should be open to question.

Thanks for taking the time to read this article.

mathew
-----BEGIN PGP SIGNATURE-----
Version: 2.2

iQCVAgUBK8AjRXzXN+VrOb1FAQFSbwP+MHePY4g7ge8Mo5wpsivX+kHYyxMErFAO
71tvtMTu66Nz6sBbPw9QkbjArbY/S2sZ9NF5htdii0R65sEyP10R6/9bv9okE/q
nihqzXE8pGvLt7t1ez4EoeHZjXLEFrEYpVayT54yQqGb4HARb0EHDCr-Te2atmP
q0Z4h5SPpAU=
=q2V5
-----END PGP SIGNATURE-----

For information about PGP 2.2, send mail to pppinfo@mantis.co.uk.
y

```

Fig 3.2 Sample article from alt.atheism category containing footer

The articles also contain header, footer and quotes section. Header is newsgroup headers and this section contains from, to, date etc. information. Footer at the end of the article is the block that contains signatures, email id etc. Quotes are the lines which are actually quoting another post and it begins with '>' sign.

3.3 PROPOSED METHODOLOGY

For analysis of unsupervised clustering on 20NewsGroups dataset, the methodology followed is:

1. From the test dataset, three categories viz. alt.atheism, sci.space, rec.sport.baseball is selected from the 20NewsGroups dataset on which clustering algorithms will be performed.
2. From the dataset, stop words like is, and, to etc. are removed so that only relevant words stay in the dataset.
3. From the dataset, header and footer part is deleted.
4. Only those words are considered in the dataset which are present in at least three documents.
5. Lastly, porter stemming is performed.

6. Binary vector is generated of the dataset.
7. The different clustering algorithms are run on this dataset and results are noted.
8. Compare the results.

3.3.1 Pre-processing steps

As we saw earlier, the unstructured data growth is more than structured data growth. Researchers are trying for knowledge discovery on the unstructured data. In order to convert unstructured into structured, we also have to pre-process the data. This is an important step today if some discovery needs to be made from these data. Pre-processing steps can transform the unstructured data into something more meaningful. Pre-processing is necessary due to the following reasons [39] –

1. It decreases the dimension of the dataset.
2. Only relevant words come into consideration.
3. Unnecessary words are reduced to a great extent.
4. Similar words are taken as one due to stemming.
5. Overall size of the dataset is also reduced.
6. Less dimension means less complexity and hence, better running time of algorithm.

We had to do some pre-processing on the three categories of the dataset before actually running the clustering algorithms on the dataset. The reason is; since the dataset was of high dimension and high number of documents, the running time of the algorithm becomes high and also, the accuracy given by the algorithm is very low. We shall explain the pre-processing done.

Stop words removal –

Stop words don't have much power in the document because they are pronouns or connecting small words like 'and', 'a', 'the' etc. Also, they occur in large number in textual data thus, increasing the dimension. So, eliminating them can make the actual relevant text more focused as they don't have effect on prediction [40]. We have simply

removed the standard stop words. This is actually the classic method [41]. Python removed the standard English stop words.

Header and footer –

Header and footer section doesn't contain that relevant information. Header contains from, to, date etc. information and footer contains signatures, email id etc. These do not hold importance and hence they were removed.

Putting a criterion on the features –

We have put a threshold of three documents on the words i.e features. It means; we only want to include the words in our dataset that are present in at least three documents.

Porter stemming –

There are many algorithms for stemming but Porter stemming [42] is the most popular one used for English language. Developed in 1980, this algorithm mainly focusses on removal of suffix so that, the words are treated as one which are differing in suffix but are semantically same [43]. Stemming means to cut the root. The words are cut short to their very basic meaning. For example; involved, involve, involving, these three words would be stemmed to 'involve'. It is cut short to the very basic word by cutting the suffix. Stemming leads to reduction in dimension since the terms are reduced to their basic form. We have used the inbuilt Python library nltk (Natural Language Toolkit) which contains the Porter Stemming function for the purpose of stemming.

Tokenization –

Token is the smallest unit in a document. Every word in the document is a actually a token. The number of tokens represents the number of features of the dataset [44]. In order to be clustered, the articles needed to be converted to be vectorized i.e bag of word model was used. But, we used Boolean values for representation which is; 0 for absent word and 1 for present word. Although we had tried to use the normal bag of word model but, the Boolean bag of word model gave better results.

After preprocessing, 1110 were the total documents and number of features were reduced to 5502.

The following figure shows the pre-processing steps done :

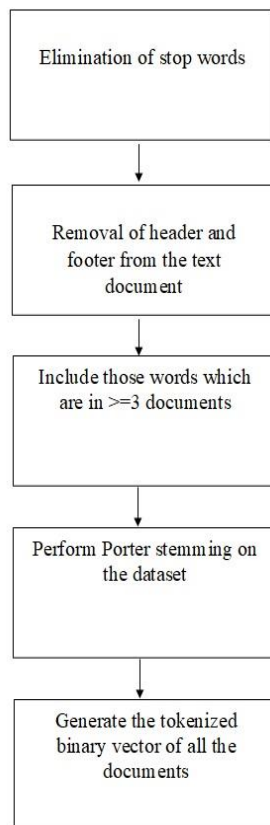


Fig 3.3 Pre-processing steps performed on the dataset

3.3.2 Algorithms implemented

For the clustering algorithms analysis on the 20NewsGroups dataset, we have implemented the following unsupervised clustering algorithms on Python, have run it and noted the results –

1. K-Means [45]:

This is the most basic and simple unsupervised clustering algorithm which is almost the first clustering algorithm to be tested for dataset. Its simplicity and less steps make it easy to implement and test. The steps of K-means are –

- i) First K random points are chosen from the dataset which initially represents K clusters i.e centroids.
- ii) The Euclidian distance is calculated for the remaining points from each cluster according to the formula given by equation 3.1.
- iii) The point having minimum distance from all the centroids will be included in that cluster. Then, mean centroid is calculated when all the points have been assigned a cluster.

- iv) The process of calculating Euclidean distance, checking the minimum distance from centroid for a point is repeated and updating centroids is done till no change in centroid centers occurs.

$$Dist = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (\text{Equation 3.1})$$

Although this algorithm has simple iterative calculation but, it also costs a lot due to overhead and low efficiency [46].

2. Fuzzy C-Means (FCM) [47]:

This clustering algorithm also falls under unsupervised category. This algorithm calculates fuzzy weights by using reciprocal of distances. The distance which is calculated for the points from each cluster is Euclidean distance. FCM allows a data point to be associated with more than one cluster by making use of degree of membership concept for every datapoint [48]. The steps of Fuzzy c-means are –

- i) First we generate M_{ka} . This is degree of ath data point membership to cluster k. Sum of M_{ka} along column is always 1.
- ii) Then we calculate the centroid value O_{kb} based on M_{ka} . (w is fuzzification parameter taken as 2) Formula used is equation 3.2.
- iii) Then, calculate Euclidean distance between the points and the ‘k’ centroids using equation 3.3.
- iv) Using all these distances we will calculate the new membership.
- v) Then, the absolute maximum difference of old and new membership is compared.
- vi) If it is less than epsilon, then we can stop otherwise we will have to calculate new centroids using the new M_{ka} .
- vii) Once algorithm stops then, we have to compare the maximum M_{ka} (membership value) out of all the ‘k’ clusters for a point. The maximum M_{ka} gives the cluster to which the point belongs.

$$O_{kb} = \frac{\sum_a (M_{ka})^w Feat_a}{\sum_a (M_{ka})^w} \quad (\text{Equation 3.2})$$

$$M_{ka} = \frac{1 / (E_{ka})^{1/w-1}}{\sum_{l=1}^k \left(1 / E_{la} \right)^{1/w-1}} \quad (\text{Equation 3.3})$$

3. Fuzzy Co-Clustering of Documents and Keywords (FCoDoK) [49]:

Co-clustering is there because there are two memberships involved, one is of the data points with the cluster (M_{ka}) and other is of the feature with the cluster (N_{kb}) [50].

Sum of column in M_{ka} is 1. Sum of row in N_{kb} is 1. E is the feature matrix i.e the dataset points and features form the matrix E.

Y_g and Y_h are weighting parameters which specify the degree of fuzziness. It can be changed accordingly. The steps of FCoDoK are –

- i) First M_{ka} is generated randomly. Then, N_{kb} is calculated using equation 3.4.
Using this N_{kb} we calculate new M_{ka} using equation 3.5.
- ii) We take the maximum absolute difference of old M_{ka} and new M_{ka} and check if it is less than epsilon. If it is so, then we can stop. Otherwise, we will again have to calculate new N_{kb} and then new M_{ka} .
- iii) Once algorithm stops then, we have to compare the maximum M_{ka} (membership value) out of all the c clusters for a point. The maximum M_{ka} represents the cluster association of the point.

$$N_{kb} = \frac{1}{l} + \frac{1}{2Y_h} \left[\sum_{a=1}^Z M_{ka} E_{ab} - \frac{1}{l} \sum_{u=1}^l \sum_{a=1}^Z M_{ka} E_{au} \right] \quad (\text{Equation 3.4})$$

$$M_{ka} = \frac{1}{K} + \frac{1}{2Y_g} \left[\sum_{b=1}^l N_{kb} E_{ab} - \frac{1}{K} \sum_{u=1}^K \sum_{b=1}^l N_{ub} E_{ab} \right] \quad (\text{Equation 3.5})$$

4. Agglomerative Hierarchical Clustering [51]:

One of the types of hierarchical clustering is agglomerative clustering. The meaning of agglomerative is clustered together. What happens in this is a hierarchy of cluster is formed; hence, it is called hierarchical clustering. First, the data points are grouped together depending on how near the points are. Initially, all the points are just an individual with no association to any cluster. Then, depending on which linkage criteria is used, the points are grouped accordingly. This grouping continues till there is a single cluster is formed at the top which contains all the data points. Since, this starts from the

bottom and builds up to the top with one cluster, agglomerative is also called bottom-up approach.

There are various linkage criteria used in agglomerative viz. complete linkage, single linkage, and ward linkage. We have used the ward linkage in our implementation. In ward linkage, at each step we check the possible groupings and only that grouping is accepted which leads to minimum sum of square for the group. Sum of square is also called variance as we check the sum of square of difference of the point with the mean according to the formula;

$$SS = \sum (A_i - A_{mean})^2 \quad (\text{Equation 3.6})$$

5. DBSCAN (Density Based Spatial Clustering of Applications with Noise) [52]:

As the name suggests, this algorithm is for clustering dataset with noise. Also, it is density based hence areas with higher density i.e more points are clubbed into one cluster. Even if there are noise in the dataset, it will detect the clusters. The disadvantage with this algorithm is every point has just one association, either it belongs to a cluster, or it does not. So, after one point is allocated to a cluster, it is done. There is no fuzzy concept which takes care of association with every cluster. This algorithm just clusters the dense region. So, there is a chance of close points to be allocated to wrong clusters.

This algorithm uses two parameter which are; epsilon and minPts. Epsilon represents the radius of circle and this circle is made around every data point to check how much density i.e how many more data points are in that circle. Now, minPts represent the least number of points that needs to be present in the circle for it to be considered a cluster. So, clusters identified by seeing the density of the points.

3.3.3 Parameters set for the algorithms

We have set various parameters and ending condition for the algorithms which are listed below-

Table 3.2: Parameters and termination condition of various algorithms

S. No	Algorithm	Parameters	Terminating condition
1.	K-Means	-	Maximum iterations=200
2.	Fuzzy C-Means (FCM)	Fuzzification parameter 'w'=2 Epsilon=0.01	Maximum iterations=200 or Frobenius Norm is less then epsilon
3.	Fuzzy Co-Clustering of Documents and Keywords (FCoDoK)	Weighting Parameters; $Y_g=0.01$ and $Y_h=1.5$ Epsilon=0.00001	Maximum iterations=50 or Frobenius Norm is less then epsilon
4.	Agglomerative clustering	Ward linkage	-
5.	Density Based Spatial Clustering of Applications with Noise (DBSCAN)	Epsilon=37.5	-

CHAPTER 4 RESULTS

When we ran the unsupervised clustering on our dataset, we got different results. The results are published online in our paper [53]. For our results analysis, we have taken two metrics which are accuracy and F1-score. Accuracy conveys how much final output is the same as the original value i.e how accurate were the results, how close were the results to the actual output. If high number of data points results are not same as the original output, then accuracy would be low. But, if lots of data points got allocated into correct clusters then accuracy would be high. That is how accuracy works.

F1 score combines both precision and recall. Precision is the ratio of out of the total positive cases identified, how many of them were correctly identified as positive. Recall is the ratio of positive cases which are identified correctly and the total actual positive and negative cases. The formula for F1 score is-

$$\text{F1 score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (\text{Equation 4.1})$$

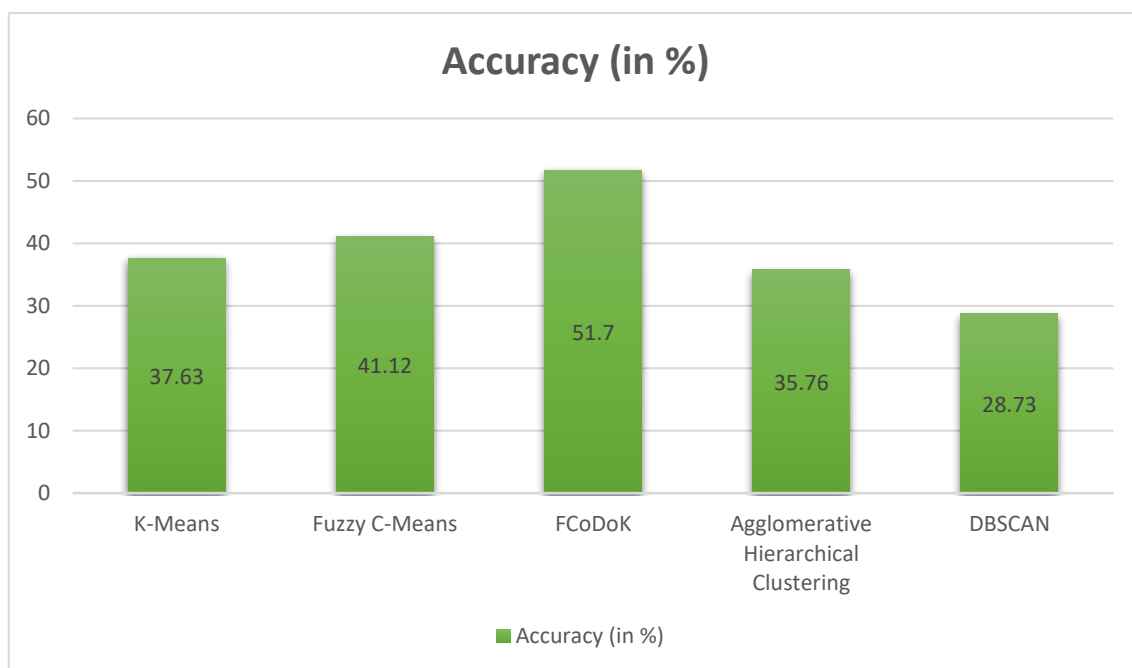


Fig 4.1 Accuracy obtained after running algorithms on 20NewsGroups dataset

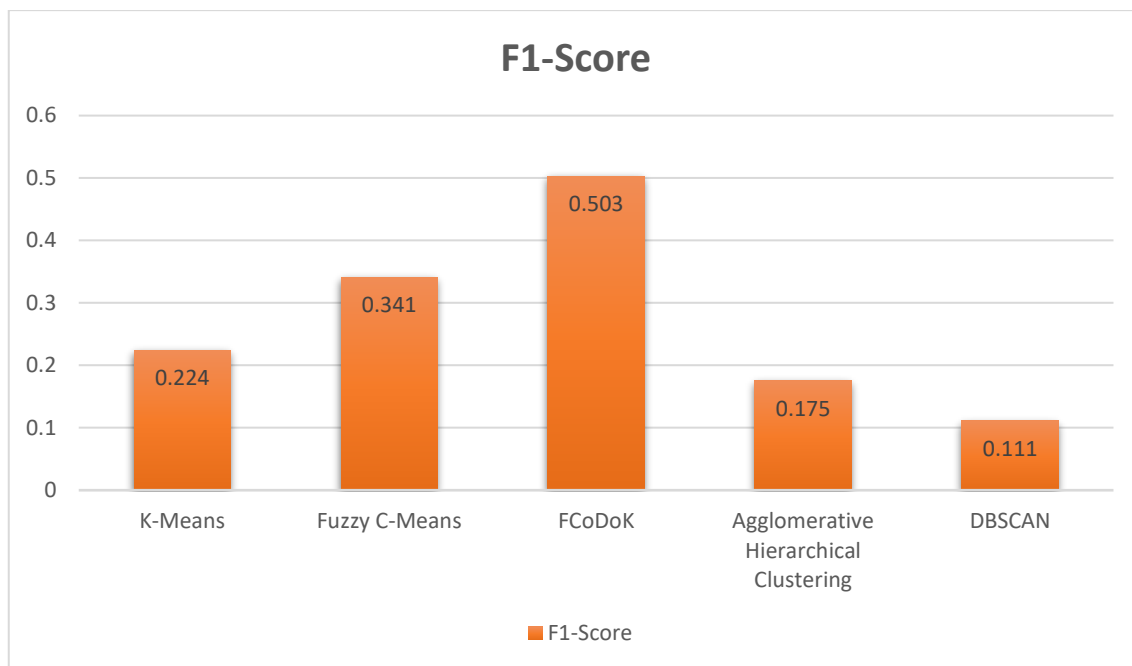


Fig 4.2 F1-Score obtained after running algorithms on 20NewsGroups dataset

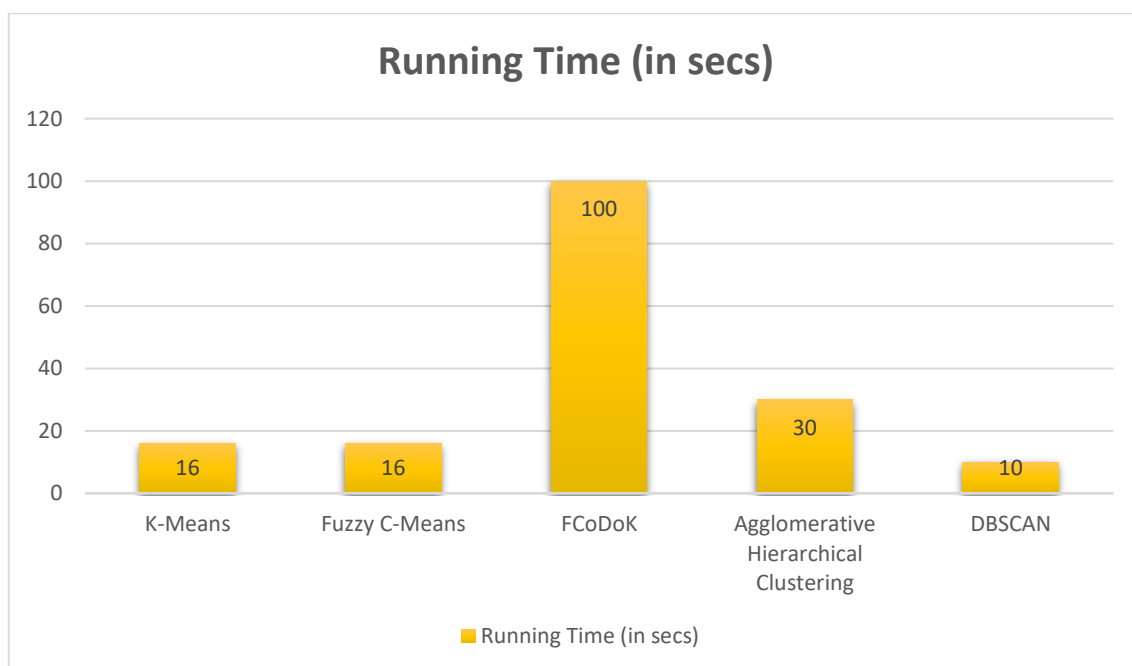


Fig 4.3 Running time of the algorithms

From the results above we observe that most effective algorithm is Fuzzy Co-Clustering of Documents and Keywords (FCoDoK) followed by Fuzzy C-Means (FCM). But, most inefficient algorithm was Density Based Spatial Clustering of Applications with Noise (DBSCAN) followed by agglomerative clustering then K-Means. Even best F1 score

was given by FCoDoK. We have taken average accuracies and F1-score. In FCoDoK we had achieved maximum accuracy of 60%. On taking average, we got around 50%. In terms of time, FCoDoK takes 100 seconds to complete the clustering process. Agglomerative takes 30 seconds to completely run the algorithm but accuracy is low. Fuzzy c-means and k-means take 16 seconds each. DBSCAN takes least amount of time but, accuracy is the lowest.

```
Accuracy2= 38.55855855855856
F1 score= 0.24584783408312824
Runtime of the program is 16.389148235321045
```

Fig 4.4 : K-means result

```
Accuracy2= 39.189189189189186
F1 score= 0.3065748684305385
Runtime of the program is 16.783209085464478
```

Fig 4.5 : FCM result

```
0.0001817520901490367 0.3333333333333333 0.3333333333333333 50.0
(1110, 6989)
(1110, 5502)
319 397 394
Normalization of Uci done.
1st parameter of Vcj done.
2nd parameter of Vcj done
Vcj done.
Normalization of entire row of Vcj.
Normalization of Vcj done.
Finding out average of the 3 original classes features :
[[319. 397. 394.]]
maximum difference= 10.08490681249024
Iteration No= 1
maximum difference= 0.0
FCoDoK stablized. No of iterations= 2
No of iterations= 2
Accuracy= 29.099099099099103
Finding out average of the 3 final classes features :
Distance between final and original= [[3.8183098 1.31162342
3.26451879]
 [1.96571478 3.98120416 3.3910999 ]
 [4.0636943 4.45897094 1.69676867]]
[1 0 2]
Accuracy 2= 60.18018018018017
F1 score= 0.5814763440997123
Runtime of the program is 105.17607474327087
```

Fig 4.6 : FCoDoK result

```
Accuracy w/o no. of clusters= 35.765765765765764
Accuracy of ward linkage= 28.82882882882883
Accuracy of complete linkage= 28.82882882882883
Accuracy of average linkage= 28.82882882882883
Accuracy of single linkage= 28.82882882882883
F1 score= 0.17585825027685495
Runtime of the program is 31.44048237800598
```

Fig 4.7 : Agglomerative clustering result

```
Accuracy= 28.73873873873874
F1 score= 0.11177295024526979
Runtime of the program is 10.093780279159546
```

Fig 4.8 : DBSCAN Result

CHAPTER 5 CONCLUSION

From the results that we have obtained on our test subset of three categories (alt.atheism, rec.sport.baseball, sci.space) of 20NewsGroups, we observed the following points –

1. Maximum accuracy was given by FCoDoK which was 51.7%.
2. Minimum accuracy was given by DBSCAN which was 28.73%.
3. Fuzzy c-means is just after FCoDoK in terms of results followed by K-means.
4. Agglomerative is a just a little better than DBSCAN but, still its not that good result.

We can conclude that the most effective algorithm for textual data clustering is FCoDoK. Due to the reason that it gives importance to both tokens and also the individual document. Whereas fuzzy c-means only gives importance to one association. For close clusters, FCoDoK is the most effective, followed by FCM. K-means, DBSCAN and hierarchical was not effective since the clusters are close. For this dataset, density based and hierarchical clustering is not an effective algorithm. Fuzzy concept should be taken into consideration for this dataset since those algorithms involving association and weights have given good results.

In the future, results can be improved for FCoDoK for these three categories and maybe, those algorithms which involve fuzzy concept can be tested against this dataset.

REFERENCES

- [1] Jaiswal, Anku, and Purrushottam Bagale. "A Survey on Big Data in Financial Sector." In 2017 International Conference on Networking and Network Applications (NaNA), pp. 337-340. IEEE, 2017.
- [2] Terol, Rafael Munoz, Alejandro Reina Reina, Saber Ziaei, and David Gil. "A Machine Learning Approach to Reduce Dimensional Space in Large Datasets." IEEE Access 8 (2020): 148181-148192.
- [3] Rama Devi Gunnam, PratyushaGudavalli, ReshmaPothuri,"Survey on Importance and Tools Used: Big Data", International Journal of Advanced Research in Computer Science and Software Engineering.
- [4] <https://archive.ics.uci.edu/ml/datasets/reuters-21578+text+categorization+collection>
- [5] <https://archive.ics.uci.edu/ml/datasets/Twenty+Newsgroups>
- [6] Dauletbak, Dalyapraz, and Jongwook Woo. "Traffic Data Analysis and Prediction using Big Data." In Proc. of KSII The 14th Asia Pacific International Conference on Information Science and Technology (APIC-IST) 2019.
- [7] Loh, Wei-Yin. "Classification and regression trees." Wiley interdisciplinary reviews: data mining and knowledge discovery 1, no. 1 (2011): 14-23.
- [8] Bhatia, Sugandh, and Jyoteesh Malhotra. "Naïve Bayes Classifier for Predicting the Novel Coronavirus." In 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), pp. 880-883. IEEE, 2021.
- [9] Oyelade, Jelili, Itunuoluwa Isewon, Olufunke Oladipupo, Onyeka Emebo, Zacchaeus Omogbadegun, Olufemi Aromolaran, Efosa Uwoghiren, Damilare Olaniyan, and Obembe Olawole. "Data Clustering: Algorithms and Its Applications." In 2019 19th International Conference on Computational Science and Its Applications (ICCSA), pp. 71-81. IEEE, 2019.
- [10] Gaikwad, Sayali, Pranali Nale, and Ravindra Bachate. "Survey on Big data Analytics for digital world." In 2016 IEEE International Conference on Advances in Electronics, Communication and Computer Technology (ICAECCT), pp. 180-186. IEEE, 2016.

- [11] HongJu, Xiao, Wang Fei, Wang FenMei, and Wang XiuZhen. "Some key problems of data management in army data engineering based on big data." In 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA), pp. 149-152. IEEE, 2017.
- [12] Pandey, Kamlesh Kumar. "Challenges of Big Data to Big Data Mining with their Processing Framework." In 2018 8th International Conference on Communication Systems and Network Technologies (CSNT), pp. 89-94. IEEE, 2018.
- [13] Praveena, MD Anto, and B. Bharathi. "A survey paper on big data analytics." In 2017 International Conference on Information Communication and Embedded Systems (ICICES), pp. 1-9. IEEE, 2017.
- [14] Sambrekar, Kuldeep, Vijay S. Rajpurohit, and Jui Joshi. "A Proposed Technique for Conversion of Unstructured Agro-Data to Semi-Structured or Structured Data." In 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), pp. 1-5. IEEE, 2018.
- [15] Aggarwal, Charu C., and ChengXiang Zhai. "A survey of text clustering algorithms." In Mining text data, pp. 77-128. Springer, Boston, MA, 2012.
- [16] Ahmad, Amir, and Shehroz S. Khan. "Survey of state-of-the-art mixed data clustering algorithms." *Ieee Access* 7 (2019): 31883-31902.
- [17] Mahmud, Mohammad Sultan, Joshua Zhexue Huang, Salman Salloum, Tamer Z. Emara, and Kuanishbay Sadatdiynov. "A survey of data partitioning and sampling methods to support big data analysis." *Big Data Mining and Analytics* 3, no. 2 (2020): 85-101.
- [18] Gunta, Aniali, and Rahul Dubey. "An Improved Document Clustering Approach with Multi-Viewpoint Based on Different Similarity Measures." In 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), pp. 152-157. IEEE, 2018.
- [19] Nikolova, Evgeniya, and Veselina Jecheva. "Applications of clustering methods to anomaly-based intrusion detection systems." In 2015 8th International Conference on Database Theory and Application (DTA), pp. 37-41. IEEE, 2015.
- [20] Yin, Pengshuai, Rui Yuan, Yiming Cheng, and Qingyao Wu. "Deep Guidance Network for Biomedical Image Segmentation." *IEEE Access* 8 (2020): 116106-116116.
- [21] Hanmandlu, Madasu, Om Prakash Verma, Seba Susan, and Vamsi Krishna Madasu. "Color segmentation by fuzzy co-clustering of chrominance color features." *Neurocomputing* 120 (2013): 235-249.

- [22] Hanmandlu, M., S. Susan, V. K. Madasu, and B. C. Lovell. "Fuzzy co-clustering of medical images using Bacterial Foraging." In 2008 23rd International Conference Image and Vision Computing New Zealand, pp. 1-6. IEEE, 2008.
- [23] Divakara, Madhura P., Keerthi V. Trimal, Adithi Krishnan, and V. Karthik. "Object Detection Using Convex Clustering–A Survey." In International conference on Computer Networks, Big data and IoT, pp. 984-990. Springer, Cham, 2018.
- [24] Gao, Zhipeng, Yidan Fan, Kun Niu, and Ting Wang. "An Adaptive Initial Cluster Centers Selection Algorithm for High-Dimensional Partition Clustering." In 2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), pp. 1119-1126. IEEE, 2017.
- [25] Cheng, Guojian, and Lianhong Liu. "Survey of image segmentation methods based on clustering." In 2020 IEEE International Conference on Information Technology, Big Data and Artificial Intelligence (ICIBA), vol. 1, pp. 1111-1115. IEEE, 2020.
- [26] Brown, Daniel, Arialdis Japa, and Yong Shi. "A fast density-grid based clustering method." In 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), pp. 0048-0054. IEEE, 2019.
- [27] Zhai, Yujia, Wei Song, Xianjun Liu, Lizhen Liu, and Xinlei Zhao. "A chi-square statistics based feature selection method in text classification." In 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), pp. 160-163. IEEE, 2018.
- [28] Liu, Fasheng, and Lu Xiong. "Survey on text clustering algorithm." In 2011 IEEE 2nd International Conference on Software Engineering and Service Science, pp. 901-904. IEEE, 2011.
- [29] Susan, Seba, and Juli Keshari. "Finding significant keywords for document databases by two-phase Maximum Entropy Partitioning." *Pattern Recognition Letters* 125 (2019): 195-205.
- [30] Rong, Youjin. "Staged text clustering algorithm based on K-means and hierarchical agglomeration clustering." In 2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), pp. 124-127. IEEE, 2020.
- [31] Liu, Gangfeng, Yunlan Wang, Tianhai Zhao, and Dongyang Li. "Research on the parallel text clustering algorithm based on the semantic tree." In 2011 6th International

Conference on Computer Sciences and Convergence Information Technology (ICCIT), pp. 400-403. IEEE, 2011.

[32] Haider, Mofiz Mojib, Md Arman Hossin, Hasibur Rashid Mahi, and Hossain Arif. "Automatic Text Summarization Using Gensim Word2Vec and K-Means Clustering Algorithm." In 2020 IEEE Region 10 Symposium (TENSYP), pp. 283-286. IEEE, 2020.

[33] Li, Yutong, Juanjuan Cai, and Jingling Wang. "A Text Document Clustering Method Based on Weighted BERT Model." In 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), vol. 1, pp. 1426-1430. IEEE, 2020.

[34] Sen, Arghyadeep, Manjusha Pandey, and Krishna Chakravarty. "Random Centroid Selection for K-means Clustering: A Proposed Algorithm for Improving Clustering Results." In 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA), pp. 1-4. IEEE, 2020.

[35] Kumbhar, Rutuja, Snehal Mhamane, Harshada Patil, Sukruta Patil, and Shubhangi Kale. "Text Document Clustering Using K-means Algorithm with Dimension Reduction Techniques." In 2020 5th International Conference on Communication and Electronics Systems (ICCES), pp. 1222-1228. IEEE, 2020.

[36] Radu, Robert-George, Iulia-Maria Radulescu, Ciprian-Octavian Truica, Elena-Simona Apostol, and Mariana Mocanu. "Clustering documents using the document to vector model for dimensionality reduction." In 2020 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR), pp. 1-6. IEEE, 2020.

[37] https://scikit-learn.org/0.19/datasets/twenty_newsgroups.html

[38] Nguyen, Ngoc Thanh, Ryszard Kowalczyk, and Jacek Mercik, eds. Transactions on Computational Collective Intelligence XXV. Vol. 9760. Springer, 2016.

[39] Vijaya, S., and R. Radha. "Preprocessingpubmed abstracts." In 2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT), pp. 1-6. IEEE, 2017.

[40] Kaur, Jashanjot, and Preetpal Kaur Buttar. "A systematic review on stopword removal algorithms." International Journal on Future Revolution in Computer Science & Communication Engineering 4, no. 4 (2018): 207-210.

[41] Ladani, Dhara J., and Nikita P. Desai. "Stopword Identification and Removal Techniques on TC and IR applications: A Survey." In 2020 6th International

Conference on Advanced Computing and Communication Systems (ICACCS), pp. 466-472. IEEE, 2020.

[42] Porter, Martin F. "An algorithm for suffix stripping." *Program* (1980).

[43] Farrar, David, and Jane Huffman Hayes. "A comparison of stemming techniques in tracing." In 2019 IEEE/ACM 10th International Symposium on Software and Systems Traceability (SST), pp. 37-44. IEEE, 2019.

[44] Fahsi, Mahmoud, and Sidi Mohamed Benslimane. "Studying the effects of conflicting tokenization on LSA dimension reduction." In 2014 International Conference on Multimedia Computing and Systems (ICMCS), pp. 542-546. IEEE, 2014.

[45] Sinha, Ankita, and Prasanta K. Jana. "A novel K-means based clustering algorithm for big data." In 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 1875-1879. IEEE, 2016.

[46] Saxena, Amit, Mukesh Prasad, Akshansh Gupta, Neha Bharill, Om Prakash Patel, Aruna Tiwari, Meng Joo Er, Weiping Ding, and Chin-Teng Lin. "A review of clustering techniques and developments." *Neurocomputing* 267 (2017): 664-681

[47] Ghosh, Soumi, and Sanjay Kumar Dubey. "Comparative analysis of k-means and fuzzy c-means algorithms." *International Journal of Advanced Computer Science and Applications* 4, no. 4 (2013).

[48] Susan, Seba, Puneet Sharawat, Sandeep Singh, Ramkesh Meena, Amit Verma, and Mukesh Kumar. "Fuzzy C-means with non-extensive entropy regularization." In 2015 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES), pp. 1-5. IEEE, 2015.

[49] Kummamuru, Krishna, Ajay Dhawale, and Raghu Krishnapuram. "Fuzzy co-clustering of documents and keywords." In *The 12th IEEE International Conference on Fuzzy Systems, 2003. FUZZ'03.*, vol. 2, pp. 772-777. IEEE, 2003.

[50] Susan, Seba, Meetu Agarwal, Seetu Agarwal, Anand Kartikeya, and Ritu Meena. "Binary clustering of color images by fuzzy co-clustering with non-extensive entropy regularization." In 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I), pp. 512-517. IEEE, 2016.

[51] Sharma, Shweta, and Neha Batra. "Comparative Study of Single Linkage, Complete Linkage, and Ward Method of Agglomerative Clustering." In 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), pp. 568-573. IEEE, 2019.

[52] Ester, Martin, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. "A density-based algorithm for discovering clusters in large spatial databases with noise." In *Kdd*, vol. 96, no. 34, pp. 226-231. 1996.

[53] Raj, Anshula, and Seba Susan. "Clustering Analysis for Newsgroup classification." *International Conference on Intelligent Computing and Communication*. Springer, Singapore, 2021.

LIST OF PUBLICATIONS OF THE CANDIDATE'S WORK

1. Anshula Raj and Seba Susan "Clustering Analysis for Newsgroup classification" accepted in conference 'International Conference on Intelligent Computing and Communication (ICICC - 2021)'.