Analysis on 20 Newsgroup & Yelp Dataset

Datasets:

- **20 Newsgroup Bydate Dataset (http://qwone.com/~jason/20Newsgroups/)**
- **Yelp Dataset (https://www.yelp.com/dataset/challenge)**

20 Newsgroup Dataset

I) Data

As the name suggest, 20 Newsgroup data set is the collection of 20 different newsgroups partitioned (nearly) evenly, with a total of 20,000 documents (1000 Usenet articles were taken from each the newsgroups). The data thus available is in term-document format. This dataset is widely used in text applications of machine learning techniques such as text clustering and classification. One of the characteristics of this dataset is that approximately 4% of the articles were crossposted. These articles have headers which includes subject, organization, article etc. The data which was partitioned across 20 newsgroups, each corresponds to a different topic. From the available information, it was observed that some newsgroups belonged to same category while others didn't.

comp.graphics comp.os.ms-windows.misc comp.sys.ibm.pc.hardware comp.sys.mac.hardware comp.windows.x	rec.autos rec.motorcycles rec.sport.baseball rec.sport.hockey	sci.crypt sci.electronics sci.med sci.space
misc.forsale	talk.politics.misc talk.politics.guns talk.politics.mideast	talk.religion.misc alt.atheism soc.religion.christian

Table 1.1

Table 1.1 categorizes 20 newsgroups based on their similarity in subject matter, example comp category(contain articles related to computer graphics, hardware etc.), sci category(contain articles related to science) and etc. This leaves us with 6 subject matters. There is a possibility of getting a common word between different newsgroup belonging to the same subject matter.

Serial No.	Topics	Documents	Unique Words
1	alt.atheism	1000	12152
2	comp.graphics	1000	15513
3	comp.os.ms-windows.misc	1000	31673
4	comp.sys.ibm.pc.hardware	1000	11858
5	comp.sys.mac.hardware	1000	11085
6	comp.windows.x	1000	19693
7	misc.forsale	1000	13216
8	rec.autos	1000	11982
9	rec.motorcycles	1000	11336
10	rec.sport.baseball	1000	10981
11	rec.sport.hockey	1000	12763
12	sci.crypt	1000	13766
13	sci.electronics	1000	12090
14	sci.med	1000	16703
15	sci.space	1000	15027
16	soc.religion.christian	997	12043
17	talk.politics.guns	1000	14128
18	talk.politics.mideast	1000	16427
19	talk.politics.misc	1000	15413
20	talk.religion.misc	1000	13940

Table 1.2

Table 1.2 was generated on original unmodified 20 Newsgroup dataset. Here we took 1 newsgroup from each subject matter and found total number of documents and unique words. The table depicts that the dataset provided is balanced because all the newsgroups have same total number of documents(except soc.religion.christian).

Unique words for each newsgroup is quite high. Also, the newsgroups with same subject matters will tend to have common words. Based on this and the world knowledge, it can be said that there might exist semantic groups. Ex: articles related to same group will tend to have common words. This can be proven by word cloud formed further in the report.

II) Experiments

II.A) Data Preprocessing:

Based on the memory available on the system on which the analysis will be performed, 3 newsgroups are chosen(newsgroups from sport, science and technology), to provide better analysis and understanding. Since newsgroups selected belongs to 3 different subject matter, clustering, LSA & LDA will perform well on this selected dataset.

Newsgroups	Documents
comp.sys.ibm.pc.hardware	1000
rec.sport.hockey	1000
soc.religion.christian	997

The data obtained from above 3 newsgroup consists punctuations, whitespace, stopwords, numbers and newsgroup-identifying headers like subject, article etc. This is a raw data which needs to be cleaned so as to get meaningful analysis. Ex: word "Organization" will appear in almost all news articles, therefore it's frequency will be more and will come up in most frequent words leading to the conclusion that it might be of importance.

Serial No.	Word	Frequency
1	Will	2782
2	Univers	1761
3	Game	1763
4	Know	1688
5	Christian	1687
6	Like	1639
7	Think	1612
8	Just	1509
9	Dont	1615
10	Drive	1444

Table 2.1

Fig-2.1

Ten top most frequent words are shown in Table 2.1. These words are frequent in the 3 selected newsgroup. WordCloud in Fig-2.1 further sheds light on the importance of each words via its size.

Following tasks were completed under data preprocessing section:

- Stopwords, whitespace, numbers and punctuations were removed using the tm_map() which cleans the raw data and prepare data for further analysis.
- Document term matrix was created using the inbuilt DocumentTermMatrix(). This matrix describes the frequency of terms which occurs in a document.
  - dtm<-DocumentTermMatrix(train.data.p,control = list(wordLengths=c(4,Inf)))
- Stemming was performed on the cleaned data because the classifier doesn't understand verbs(i.e. organized & organizing) and treats them as different words.

- Pruning words by frequency: words that occurs in very few documents were removed. The vocabulary size after pruning decreased from 24875 to 1546, which is good because the entire focus is on the words which have high frequency. Therefore during LSA/LDA/Clustering, important data is analyzed. Both LSA and LDA are topic models.
- Tf-Idf weights were used on Document Term Matrix which is one of the popular term-weighting schemes. It is a statistical measure which evaluates the importance of word in the document.

```
> dtm_tfidf<-weightTfIdf(dtm)
> dtm_tfidf
<<DocumentTermMatrix (documents: 2997, terms: 24875)>>
Non-/sparse entries: 256494/74293881
Sparsity           : 100%
Maximal term length: 20
Weighting          : term frequency - inverse document frequency (normalized) (tf-idf)
```

Fig-2.2

II.B) Clustering Experiments:

1. Document vectors from tf-idf document-term-matrix are clustered with k-means for comparisons.
   - K-Means: For k-means clustering, kmeans() is used. With the help of plot(), plots were obtained for k = 3(in Fig-2.3). To set parameter K, there are various methods like: Elbow Method, Average Silhouette Method etc.
     - Average Silhouette Method will be used here for determining the optimal number of clusters. This approach determines how well each object lies within its cluster. Optimal of clusters k is the one with maximum average silhouette.
     - Fig-2.4 shows the optimal number of clusters. The process to compute average silhouette method is then wrapped up in a single function (*fviz_nbclust(matrix_tfidf, kmeans, method = "silhouette")*) which belongs to the library "factoextra".
     - Result: 3 clusters maximizes the average silhouette values. Therefore, K=3

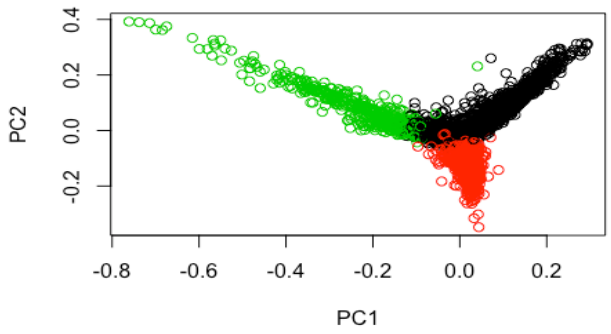


Fig-2.3

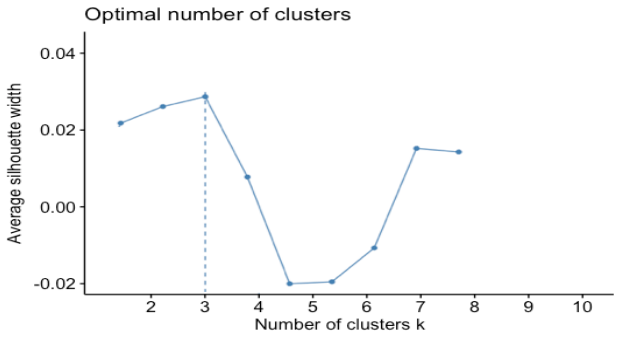


Fig-2.4

   - NbClust: Function NbClust() was used to determine the best numbers of cluster. Parameters: minimum number of clusters = 3, maximum number of clusters = 9, distance = euclidean, index = silhouette.
     - NbClust(as.matrix(tfidf), distance = "euclidean", min.nc = 2, max.nc = 9, method = 'kmeans', index = 'silhouette')
     - $Best.nc: Number_clusters = 3 ; Value_Index = 0.4678
     - Result: Best number of clusters is 3.

2. LSA: It is a form of topic modeling which uses reduced-rank SVD to project document term matrix into semantic spaces, thus making clustering faster.

   - SVD on Document Term Matrix: *svd_dtm <- svd(dtm)*
     - Dimensions : D [1:316] ; U [1:2997,1:316] ; V [1:1415, 1:1316]
     - SVD(Document-Term Matrix) = U*D*V^T ; this is nothing but an interpretation of SVD for Document-Term matrix.

- Here, U relates terms to topics, V relates documents to topics & D gives importance of topics. Thus, D*V^T provides k dimensional LSA document vectors whereas D*U provides k dimensional LSA word vectors.

- Compute d = 50,100,200 dimensional representation of TDM: Algorithm is as follows -
  - S <- svd(dtm)
  - Dd <- diag(S$d)[1:d , 1:d] gives the importance of topics where d is 50,100 or 200.
  - Mat <- S%v[,1:d] %*% Dd %*% t(S%u[,1:d])

- Frequent words for top 5 concepts :

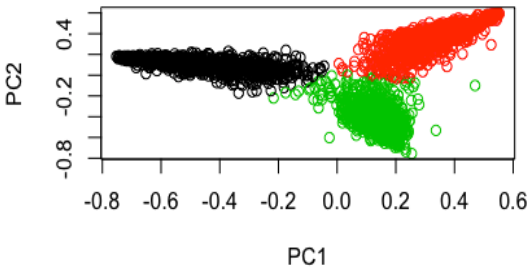
No. Of Concept	Frequent Words
1	Christian, people, believ, Jesus,church
2	Game, hockey, time, first, monitor
3	love, sound, call, software, drive
4	Book, like, also, will, come
5	Machine, card, read, christ, week

3. LDA:

- LDA model will be estimated using Gibbs Sampling. The main parameters for LDA() are as follows:
  - burnin - starting period, steps which does not reflect distribution property are removed.
  - iter - number of iterations
  - thin - number of iteration at which correlation between samples is avoided
  - seed - an integer for each starting point
  - nstart - number of runs at different start points
  - best - return result of best run
  - LDA(dtm,4, method="Gibbs", control=list(nstart=nstart, seed = seed, best=best, burnin = burnin, iter = iter, thin=thin))

- Cluster LDA vectors with kmeans using same k as above: K = 3(Fig-2.5) & SSE Measure
-

- For each of the top 5 concepts report the most representative words:(Fig-2.6)




```
> ldaOut.terms
      Topic 1                Topic 2           Topic 3
 [1,] "drive"                "game"            "christian"
 [2,] "newsgroup"            "team"            "socreligionchristian"
 [3,] "system"               "play"            "peopl"
 [4,] "messageid"            "recsporthockey"  "will"
 [5,] "scsi"                 "hockey"          "believ"
 [6,] "card"                 "year"            "approv"
 [7,] "compsysibmpchardwar"  "player"          "christianaramisrutgersedu"
 [8,] "sale"                 "messageid"       "church"
 [9,] "miscforsal"           "first"           "jesus"
[10,] "email"                "will"            "mean"
```

Sum of Square	Sum of Square	Sum of Square
312	103	33%

II.C) Evaluation:

- Comparing Clustering results by evaluating SSE:
  Within-SSE Ratio <- ((k$tot.withinss / k$totss) * 100)

Clustering with k-means:

1) SSE Measure:

Number of Clusters	Total Sum of Squares	Total Within Sum of Squares	Within-SSE Ratio
2	1287	1256	97%
3	1287	1221	94%
4	1287	1222	95%

2) Confusion Matrix:

Accuracy <- (sum(apply(c_matrix,1,max))/sum(k3$size))*100

	1	**2**	**3**
comp.sys.ibm.pc.hardware	962	34	4
rec.sport.hockey	19	960	21
soc.religion.christian	0	6	991

Accuracy : 94%

3) Precision, Recall & F1:

	comp.sys.ibm.pc.hardware	**rec.sport.hockey**	**soc.religion.christian**
Precision	0.86	0.98	0.99
Recall	0.98	0.82	0.96
F1	0.91	0.90	0.97

LSA: SSE Measure:

Concepts	Sum of Square	Total Within SSE	Ratio
50	2374	2065	87%
100	2487	2263	91%
200	2538	2385	94%

Confusion Matrix : for d = 200(same way it can be computed for d = 50,100)

	1	**2**	**3**
comp.sys.ibm.pc.hardware	996	0	4
rec.sport.hockey	7	993	0
soc.religion.christian	1	4	992

Accuracy : 99%

LDA:

SSE Measure :

Sum of Square	Sum of Square	Sum of Square
62	31	51%

Confusion Matrix: with accuracy = 85%

	1	2	3
comp.sys.ibm.pc.hardware	996	0	4
rec.sport.hockey	7	969	24
soc.religion.christian	22	4	971

Precision, Recall & F1:

	comp.sys.ibm.pc.hardware	rec.sport.hockey	soc.religion.christian
Precision	0.69	0.92	0.97
Recall	0.95	0.67	0.85
F1	0.82	0.71	0.95

Yelp Dataset

I) Data

- Yelp dataset was released for academic challenges which is quite bigger when compared to 20 Newsgroup Dataset. The dataset downloaded from the website is 5.79Gb in size, with 6 files in JSON format.
- This dataset contains user, business, review, checkin, tip & photo information about local businesses in 12 metropolitan areas across 4 countries, with around 156639 businesses.
- Since, the yelp dataset is bulky, it is not a feasible plan to work on the entire dataset. Rather two of the json files are selected so that sub-problem infer-categories can be worked upon. Therefore, business.json and review.json are chosen.
- This requires preprocessing both the business and review json files. Jsonlite library can be used to work upon json files and extract what is required. There are total 1240 categories with various reviews. Out of these categories, three are selected to perform our analysis, which are “mobile phones”, “real state” & “active life”. The selected 3 categories are the reviews in the dataset.
- Since these 3 categories belong to different fields but the analysis performed on the reviews showed that there might have been a mix of topics. Example: reviews for mobile phones are mostly about electronics like where to buy from, repair shops etc.
- The raw data thus obtained contains many meaningless and redundant data which needs to be processed before performing any analysis.

II) Experiments

II.A) Data Preprocessing:

Before preprocessing, the raw data must be converted from .json to .csv format. The required fields must be extracted. From business.json, business_id and categories must be extracted and saved in a business.csv file, where as from review.json, business_id and text must be extracted and saved to review.csv. Both of the files should be merged on their common business_id.

The three categories which were decided in previous section will be used to provide better analysis and understanding. Since they belong to 3 different fields, clustering, LSA & LDA will perform well on this selected data.

Categories:

- Real Estate
- Active Life
- Mobile Phone

The data obtained contains whitespace, punctuation & stop words which are removed and is then stemmed to make the data available meaningful. The following table shows the most frequent words & figure provides the word cloud.

Serial No.	Word	Frequency
1	Apart	897
2	Live	868
3	Place	768
4	Time	751
5	Move	737
6	Manage	636
7	Like	559
8	Great	511
9	Just	509
10	Work	490

Following tasks were completed after preprocessing:

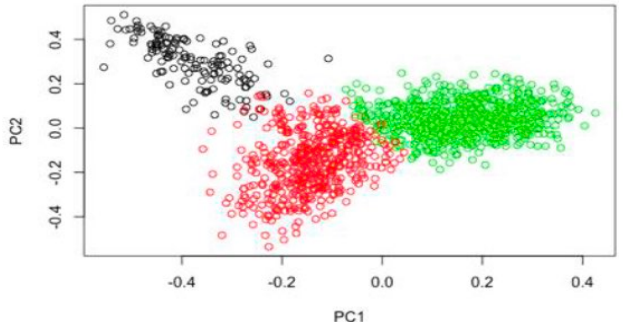
- Document term matrix was created using the inbuilt DocumentTermMatrix(). This matrix describes the frequency of terms which occurs in a document.
- Stemming was performed on the cleaned data because the classifier doesn't understand verbs(i.e. organized & organizing) and treats them as different words.
- Pruning words by frequency: words that occurs in very few documents were removed. The vocabulary size after pruning decreased from 36452 to 1415, which is good because the entire focus is on the words which have high frequency. Therefore during LSA/LDA/Clustering, important data is analyzed. Both LSA and LDA are topic models.

- Tf-Idf weights were used on Document Term Matrix which is one of the popular term-weighting schemes which will be used for document-term matrix. It is a statistical measure which evaluates the importance of word in the document.

II.B) Clustering Experiment:

1. K-Means & NbClust: This is same as what was done for 20 Newsgroup Dataset. Just like our previous dataset, k-means is plotted for k = 2,3,4. Again, Silhouette method was used to discuss and set the value of K. Following command was used to evaluate k-means.

```
k3 <- kmeans( dtm_tfidf, 3)
plot(prcomp(tfidf)$x,col=k3$cluster)
```




```
#Optimal Clusters
avg_sil <- function(k) {
  km.res <- kmeans(tfidf, centers = k, nstart = 25)
  ss <- silhouette(km.res$cluster, dist(tfidf))
  mean(ss[, 3])
}

# Compute and plot wss for k = 2 to k = 9
k.values <- 2:9

# extract avg silhouette for 2-9 clusters
avg_sil_values <- map_dbl(k.values, avg_sil)
```

Average Silhouette method is used next to set K value.
Result: The graph plotted for silhouette method gives the optimal number of clusters as 3.
Confusion Matrix:
Accuracy: 79%

	1	2	3
Real Estate	476	9	128
Mobile Phone	23	349	1
Active Life	0	65	275

NbClust: Function NbClust() was used to determine the best numbers of cluster. Parameters: distance = euclidean, minimum number of clusters, maximum number of clusters, method, index.

- NbClust(as.matrix(tfidf), distance = "euclidean", min.nc = 2, max.nc = 9, method = 'kmeans', index = 'silhouette')
- $Best.nc: Number_clusters = 3 ; Value_Index = 0.4678
- Result: Best number of clusters is 3.

2. LSA:

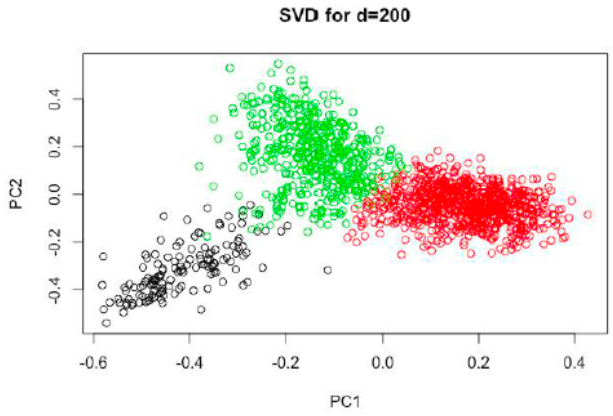
Most Frequent words for 5 concepts:

No. Of Concept	Frequent Words
1	apart, manag, time, live, month
2	phone, repair, great, servc, like
3	life, golf, play, will, golf
4	store, help, nice, staff, leas
5	clean, cours, green, like, est

SSE Measure for LSA

Concepts	Sum of Square	Sum of Square	Sum of Square
50	873	794	91%
100	834	775	93%
200	924	877	95%

For d = 200: Cluster plot & Confusion matrix. Accuracy = 68% (same way it can be shown for d = 50,100)



	1	2	3
Real Estate	476	9	128
Mobile Phone	23	321	29
Active Life	29	12	299

3. LDA: Same procedure as 20 Newsgroup dataset.

Most representative words

```
      Topic 1  Topic 2  Topic 3
 [1,] "time"   "place"  "apart"
 [2,] "just"   "like"   "live"
 [3,] "work"   "great"  "move"
 [4,] "call"   "staff"  "manag"
 [5,] "need"   "year"   "month"
 [6,] "even"   "nice"   "offic"
 [7,] "come"   "look"   "rent"
 [8,] "back"   "cours"  "leas"
 [9,] "servic" "good"   "peopl"
[10,] "never"  "friend" "complex"
```

SSE Measure

Sum of Square	Sum of Square	Sum of Square
62	31	51%

II.D) Results Summary:

So far, 3 types of document representations were evaluated, which are tf-idf, LDA & LSA. Few useful observations are as follows:

- Tf-idf used word frequency counts as document vector's feature.
- LSA & LDA represents documents as vectors in space.
- LDA's results were not satisfactory in both the cases.
- LSA's result was much better than tf-idf. When 200 dimensional LSA representation is used, best clustering is obtained.

Below table summarizes the results so far:

20 NewsGroup	Accuracy	Yelp	Accuracy
Tf-idf	94%	**Tf-idf**	79%
LSA(d=50)	89%	**LSA(d=50)**	73%
LSA(d=100)	97%	**LSA(d=100)**	70%
LSA(d=200)	99%	**LSA(d=200)**	68%
LDA	85%	**LDA**	51%

III) Analysis:

- The analysis performed here was designed in a way to make it easier to grasp the concepts behind mining of the data.
- Datasets used for our analysis were 20 Newsgroup Dataset and Yelp Dataset, with different characteristics. The raw data was then preprocessed at initial stage.
- Pruning words by i.e. removing words which occur in very less documents is a better way to prepare data for analysis so that focus remains on the most important data.
- Tf-idf weights were then used on document term matrix to reflect how important a word is to a document in a collection.
- The K value obtained via k-means was further supported by Silhouette method and NbClust in both the datasets.
- LSA, depending upon the dataset can handle synonymy problems. It has better runtime since it only involves decomposing document term matrix. It becomes less efficient in front of deep neural networks.
- LSA was computed for different dimensions, but a word of caution, computation of LSA for different dimensions will affect the distance between vectors of document. Hence, the increase in SSE for clustering LSA. Although, LSA with 200 dimension gave highest accuracy, meaning clustering was not that worse.
- LSA concepts when more in number gives better result, this was proved when high accuracy was achieved for 200 dimensional LSA.
- The clustering plots shows that the documents from each of the groups were nicely separated.
- The most frequent words obtained as a result of LDA representation in 20 Newsgroup shows that LDA successfully discovered semantic topics of 3 newsgroups. Topics discovered were related to Christian, sports and technology.
- Yelp data's result were not that satisfactory when compared with 20 newsgroup.

- This project gave me an opportunity to work on real data, data obtained from newspaper and yelp. I learned to :
    - Work on data right from the scratch, from when the data is raw and meaningless to meaningful data.
    - Perform cluster experiments on the data and find the word that belongs to the same group.
    - LSA & LDA, which taught how to get words belonging to a certain topic.