# Tutorial on mining of biomedical literature with the help of R Package

## Vinaitheerthan Renganathan

## www.vinaitheerthan.com

### Abstract

This paper provides step by step overview of process involved in mining of biomedical literature using R-Statistical Package. Abstract from PubMed database on a given topic are retrieved, stored, pre-processed using R programming codes. The resultant term document matrix is used to find association between terms and frequency of the terms in each document. Finally the clouds of words and clustering of documents are created using the R software to discover the association between the documents. The results from the process provided a step by step understanding of the retrieval of abstracts, pre-processing of abstracts and clustering of abstracts using the user based query term

**Keywords:** Biomedical, Clustering, Classification, R Software, Text mining

## 1. Introduction

This paper assumes that the readers have knowledge about text mining concepts and especially in biomedical domain. Those who are interested to get an overview of text mining and its application biomedical domain are encouraged to refer to the author's paper on text mining in Biomedical domain [1] - Renganathan.V. (2017). Text Mining in Biomedical Domain with Emphasis on Document Clustering Healthcare informatics research, 23(03) Pages 141-146

## 2. R Package

The R [2] package is an open source statistical computing software which is useful for carrying out various statistical tests and methods, graphics, text and data mining procedures. The R software can be downloaded from the software website [1]. The R software can be used in various integrated development environment (IDE) such R-Studio, Eclipse and StatET. This paper use R-Studio [3] IDE which is an open source software and can be downloaded from the R-Studio website [3].

The R software works with the concepts called packages which is a compilation user created codes and can be used to perform specific functions.

The following steps describe the use of R codes in mining the abstracts from the PubMed database on a given topic or search query. It will help in extracting information, association between terms and creating clustering of abstracts based on the similarity measures

### Step1: Installing required packages

Before starting the text mining process, the following packages needs to downloaded and installed in the R Environment using the following codes

**Required packages description**

**tm package[4]**

tm packages used to perform the following steps during the text mining process : data import, corpus(collection of document) handling, pre-processing, meta data management, and creation of term-document matrices

**Wordcloud [5]**

Wordcloud packages allows us to build a graphical representation of words with font size equals the frequency of the word appearing in the given Term document matrix in relation to the other words

**RISmed [6]**

RISmed package is used to extract bibliographic information article abstracts from the Databases such as PubMed, NCBI

**Cluster [7] and fpc [8]**

Cluster and fpc package is used to group the documents into similar groups and grouped documents in the particular documents will be similar to each other and will be dissimilar to the documents in the other cluster

***R-Code***

```
>Install.packages( wordcloud)
>Install.packages(tm)
>Install.packages(RISmed)
>Install.packages(fpc)
>Install.packages(cluster)
```

## Step2: Calling the packages in the R environment using library function

Once the packages are installed packages needs to be called in the R current working environment

***R-Code***

```
#Specification of the required libraries (lines starting with # comment line for reference purpose only)
>library (wordcloud)
>library (tm)
>library (RISmed)
>library (cluster)
```

## Step 3: Retrieving the documents from PubMed and creating document corpus

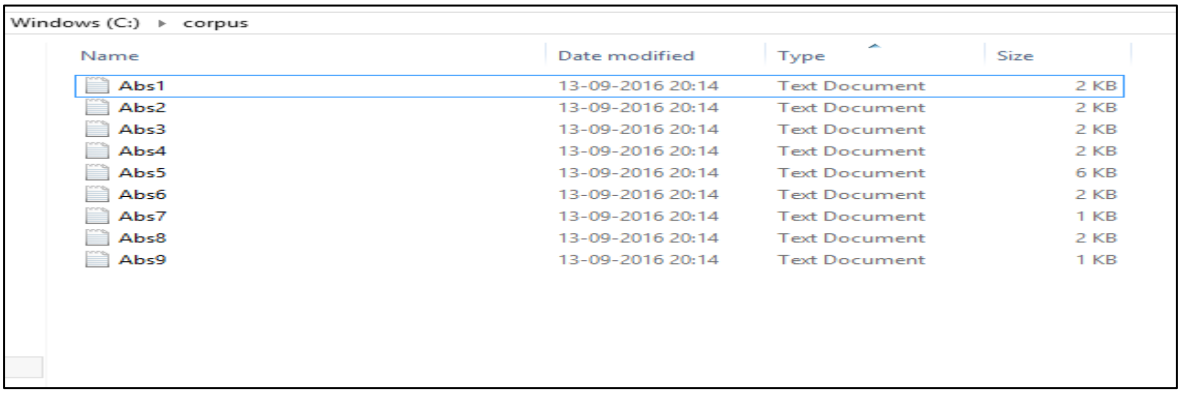
In the next step using the RISMED the abstracts needs to be downloaded and saved into the local directory (Ex: C:/corpus) for the text mining purpose

***R-Code***

```
#input of query terms

>query <- 'cancer'
```

```
#Specification of query Criteria with minimum and maximum date of publication of Article, number of Articles to return
>query_level2 <- EUtilsSummary(query, retmax=100, mindate=2016, maxdate=2016)
>query_level3<- EUtilsGet(query_level2)
>class(query_level3)
#Retrieval of abstracts from PubMed
>pubmed_data <- data.frame('Abstract'= AbstractText(query_level3))
#Storage of retrieved abstract in the form of individual text document in a directory called C:/Corpus which is displayed as figure-1
>for (Abs in 1:9)
>{
>doc1 <- data.frame(pubmed_data[Abs, ])
>doc2 <- file.path("c:/corpus", paste0(Abs, ".txt"))
>write.table(doc1, file = doc2, sep = "", row.names = FALSE, col.names = FALSE, quote = FALSE,
  append = FALSE)
>}
```



Windows (C:) ▸ corpus

| Name | Date modified | Type | Size |
|---|---|---|---|
| Abs1 | 13-09-2016 20:14 | Text Document | 2 KB |
| Abs2 | 13-09-2016 20:14 | Text Document | 2 KB |
| Abs3 | 13-09-2016 20:14 | Text Document | 2 KB |
| Abs4 | 13-09-2016 20:14 | Text Document | 2 KB |
| Abs5 | 13-09-2016 20:14 | Text Document | 6 KB |
| Abs6 | 13-09-2016 20:14 | Text Document | 2 KB |
| Abs7 | 13-09-2016 20:14 | Text Document | 1 KB |
| Abs8 | 13-09-2016 20:14 | Text Document | 2 KB |
| Abs9 | 13-09-2016 20:14 | Text Document | 1 KB |

**Figure-1 – Document Corpus**

```
#Setting of Directory for Text Mining
source <- DirSource("c:/corpus")
testdoc <- Corpus(source)
```

**Step4 : Pre-processing of Documents and preparing the Term Document Matrix**

Once the documents are stored in the corpus, the documents needs to pre-processed by removing the stop words such as ‘is’, ‘are’ etc. ,removing the numbers, removing the white spaces. After pre-processing of documents, Term Document matrix to be prepared using tm package

```
#Removal of Stop words

testdoc1 <- tm_map(testdoc, removeWords, c("may","are","use","can","the", "then", "this", "is",
"a", "well", stop words("english")))

#Removal of whitespace, stemming of words to its root word, removal of numbers

>testdoc2 <- TermDocumentMatrix (testdoc1, control = list(tokenize=scan_tokenizer,  stopwords =
TRUE,  removePunctuation = TRUE,  stripWhitespace = TRUE,  stemming = TRUE,  removeNumbers=
TRUE
 ))
```

**Step 6: Finding Term Frequency (TF), Inverse Document Frequency (IDF) and association between the terms**

After the preparation of document matrix, the frequency of the terms, association between a term and other terms in the documents are prepared

```
#Calculation of the frequency of the terms and inverse term frequency in the documents
#Conversion of Term-document matrix  to a normal matrix and term frequencies are calculated and
which is shown in the table -3
>testdoc3 <- as.matrix(testdoc2)
>testdoc4 <- sort(rowSums(testdoc3),decreasing=TRUE)
>testdoc5 <- data.frame(word = names(testdoc4),freq=testdoc4)
>head(testdoc5, 10)
```

| Terms | Frequency in the documents |
|---|---|
| patient | 27 |
| cancer | 24 |
| identifi | 14 |
| autoantibodi | 12 |
| cell | 11 |
| lung | 11 |
| autoimmun | 10 |
| report | 10 |
| effect | 9 |
| novel | 9 |

**Table-3- Term Frequency**

```
#Association found for the term infect with other term in document corpus and results are given in
Table-4

>findAssocs(x=testdoc2, term="infect", corlimit=0.6)
```

| | | | | |
|---|---|---|---|---|
| among | chain | conclus | confirm | conflict |
| 1 | 1 | 1 | 1 | 1 |
| contribut | cutan | dna | eightyon | eightythre |
| 1 | 1 | 1 | 1 | 1 |
| exist | formalinfix | fraction | harbor | hpv |
| 1 | 1 | 1 | 1 | 1 |
| hpvs | laryng | marker | methodsresult | mucos |
| 1 | 1 | 1 | 1 | 1 |
| nasopharyng | oral | oropharyng | papilloma | papillomasmethod |
| 1 | 1 | 1 | 1 | 1 |
| papillomavirus | paraffinembed | pcrbase | polymeras | preval |
| 1 | 1 | 1 | 1 | 1 |
| reaction | sampl | support | test | twentyfour |
| 1 | 1 | 1 | 1 | 1 |
| head | neck | lesion | also | analyz |
| 0.95 | 0.95 | 0.88 | 0.66 | 0.66 |
| background | howev | human | inc | need |

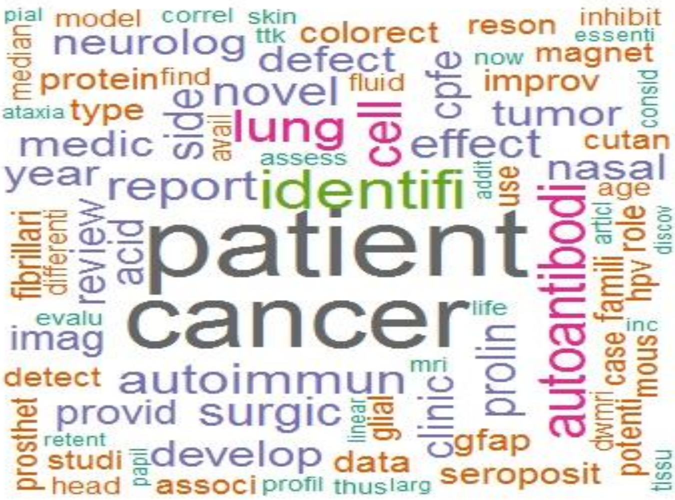
| | | | | |
|---|---|---|---|---|
| 0.66 | 0.66 | 0.66 | 0.66 | 0.66 |
| period | squamous | wiley | case | |
| 0.66 | 0.66 | 0.66 | 0.61 | |

**Table-4 – Association between Term Infect and other terms in the document collection**

**Step 7: preparation of word cloud**

Clouds of words are constructed using wordcloud package to pictorially represent the importance of terms in the document corpus

```
# Construction of the word cloud. The sample word cloud is given in the figure -2
>set.seed(1234)
>wordcloud(words = testdoc5$word, freq = testdoc5$freq, min.freq = 1,
        max.words=200, random.order=FALSE, rot.per=0.35,
        colors=brewer.pal(8, "Dark2"))
```



**Firgure-2 :** Word cloud constructed from the document corpus larger font represents words with more frequency

**Step 8: Creation of Hierarchical and K-Means word cluster using Cluster and fpc packages**

Words are clustered using the Hierarchical clustering and K-Means clustering techniques. We Sparse terms are removed and distance between the terms are calculated

```
#Creation of cluster of words using Hierarchical clustering technique
# removal of Sparse Terms
>testdoc5 <- removeSparseTerms(testdoc2, 0.70)
#Conversion of Term Document Matrix to normal matrix
>c1 <- as.matrix(testdoc5)
#Calculation of distances
>c2 <- dist(c1)
>c3 <- hclust(c2, method="ward.D")
#Dendogram is shown in the figure-3
>plot(c3, hang=-1)
```

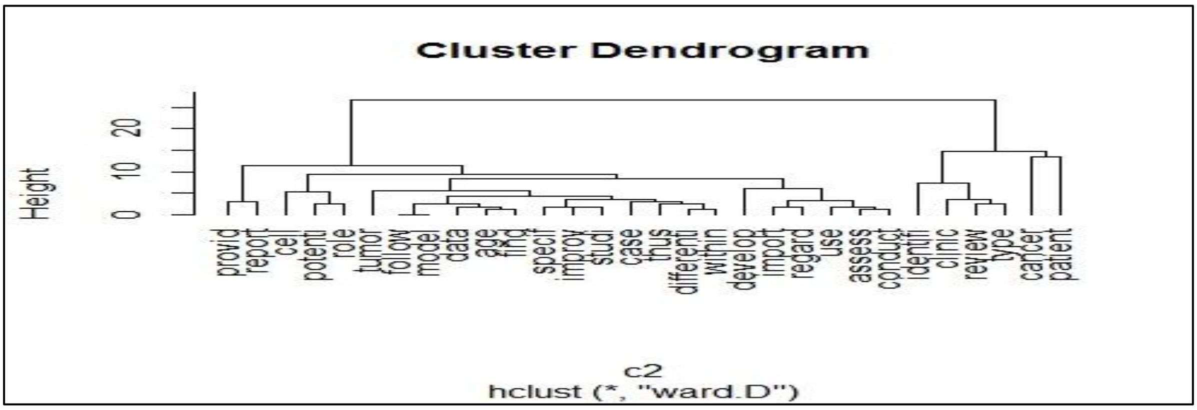


**Figure-3**: Cluster of words

```
>plot.new()
>plot(c3, hang=-1)
```

#Creation of cluster K-mean clustering by defining prior the number of clusters

```
>km1 <- kmeans(c2, 2)
>clusplot(as.matrix(c2), km1$cluster, color=T, shade=T, labels=2, lines=0)
```

**Step9: Preparation of Document cluster**

For creating a document cluster, Document Term Matrix (DTM) needs to be created first from the document

```
>doctest <- DocumentTermMatrix (testdoc1, control = list (tokenize=scan_tokenizer, stopwords =
TRUE, removePunctuation = TRUE,
                                   stripWhitespace = TRUE,
                                   stemming = TRUE,
                                   removeNumbers= TRUE

))
>c1 <- as.matrix(testdoc5)
>c2 <- dist(c1)
>c3 <- hclust(c2, method="ward.D")
```

#Dendogram for document cluster is shown in the figure-4

```
>plot(c3, hang=-1)
>plot.new()
>plot(c3, hang=-1)
```

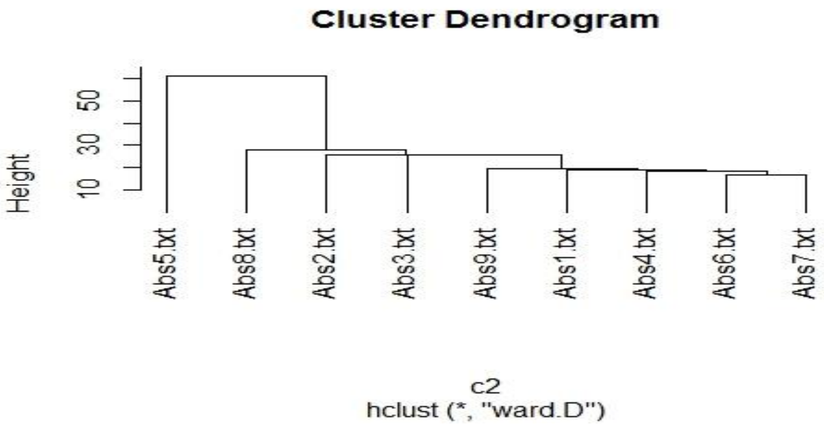


**Figure-4:** Cluster of Abstracts

## 3. Results and Discussions :

The text mining process resulted in finding association between terms in the document like " infect ion" and "nasopharyngeal " is 1 and "infection" and "squamous" is 0.95 from the above table-4. It also provided cloud of words which displayed the terms having higher frequency with higher font size.(figure-2) . The figure 3 depicted the word cluster helped to know which words are closer to each other like "cancer" and "Patient" and similarly the figure-4 helped to understand which studies are closer to each other abstract 6 and abstract 7.

## 4. Limitation:

Currently the non-availability of free full text article will be a draw back for text mining research purpose

## 5. Future work:

R-Code for classifying the above documents requires training set and validation test. In future the author is planned to further develop the R-codes for the classification part

## 6. Conclusions :

Text mining process was explained with the help of R-Statistical process using the abstracts from PubMed online database. The results from the process provided a step by step understanding of the retrieval of abstracts, pre-processing of abstracts and clustering of abstracts using the user based query term.

## References

1. Renganathan.V. (2017). Text Mining in Biomedical Domain with Emphasis on Document Clustering, Healthcare Informatics Research, 23(03), Pages 141-146
2. https://cran.r-project.org/
3. https://www.rstudio.com/
4. https://cran.r-project.org/web/packages/tm/index.html
5. https://cran.r-project.org/web/packages/wordcloud/index.html
6. https://cran.r-project.org/package=RISmed
7. https://cran.r-project.org/web/packages/cluster/index.html
8. https://cran.r-project.org/package=fpc