ISSN 2090-3359 (Print)
ISSN 2090-3367 (Online)

ΑΔΣ

Advances in Decision Sciences

Volume 25
Issue 3
September 2021

Michael McAleer

Editor-in-Chief
University Chair Professor
Asia University, Taiwan

Published by Asia University, Taiwan

ADS@ASIAUNIVERSITY

A Detailed Guide on How to Use Statistical Software R for Text Mining

Kim-Hung Pho

Fractional Calculus, Optimization and Algebra Research Group,

Faculty of Mathematics and Statistics, Ton Duc Thang University, Ho Chi Minh City, Vietnam

Ngoc-Hien Nguyen

Design Innovation Center (DBZ), Faculty of Engineering, Mondragon University, Spain

Huu-Nhan Huynh

Department of Mathematics and Informatics,

Vietnam Aviation Academy, Ho Chi Minh City, Vietnam

Wing-Keung Wong*

Department of Finance, Fintech Center, and Big Data Research Center, Asia University, Taiwan

Department of Medical Research, China Medical University Hospital, Taiwan

Department of Economics and Finance, the Hang Seng University of Hong Kong, Hong Kong

* Corresponding author: wong@asia.edu.tw

Acknowledgments: The authors thank the Editors and two anonymous referees for their helpful comments which help to improve our manuscript significantly. The fourth author would like to thank Robert B. Miller and Howard E. Thompson for their continuous guidance and encouragement. This research has been supported by Ton Duc Thang University, Mondragon University, Vietnam Aviation Academy, Asia University, China Medical University Hospital, The Hang Seng University of Hong Kong, Research Grants Council (RGC) of Hong Kong (project numbers 12502814 and 12500915), and Ministry of Science and Technology (MOST, Project Numbers 106-2410-H-468-002 and 107-2410-H-468-002-MY3), Taiwan. However, any remaining errors are solely ours.

Abstract

Purpose: Text mining is a very important issue in Statistics, Applied Mathematics, and many other areas in Sciences, Engineering, and Business because its applications are extremely rich and varied. Text mining can help academics and practitioners with some specific issues such as spam filtering, personal background matching, sentiment analysis, document classification, etc.

Design/methodology/approach: The statistical software R is an exceedingly widely used software in Science because of its outstanding and completely free features. To contribute to the literature related to text mining, this study provides detailed instructions on how to use the statistical software R for text mining.

Findings: To implement this goal, we first introduce the algorithm for text mining. We then discuss how to use the software R to approach each step of the algorithm in detail. As an application, the proposed algorithm is studied with an actual data set.

Originality/value: This study provides detailed instructions on how to use the statistical software R for text mining which is new in the literature.

Implications: The results found in this study will help academics and practitioners understand how to use the statistical software R to analyze text mining. This paper is very useful for both academics and practitioners in the study of text mining.

Keywords: Guide, Text Mining, Statistics, software R.

JEL: J16, K38, M14.

Paper type: Research paper

Introduction

Text database is developing very quickly and attracting research interest around the world nowadays. The reason is quite simple due to the rapid increase in the amount of information in digital form, which can be easily mentioned such as electronic documents, emails, and websites. In addition, it can be seen that most of the information of organizations, industries, and schools is digitized and stored in this form of a database. Hence, studying and analyzing the text database is extremely interested in the current phase. As we know, an algorithm that is very popularly used to analyze this problem is text mining.

Text mining is a branch of data mining with the primary purpose is to find and extract information contained in the text. Currently, with the rapid growth of text data, text mining is becoming more and more crucial and meaningful in practice. Some specific applications can be effortlessly encountered such as spam filtering, personal background matching, sentiment analysis, and document classification. Therefore, researching and learning about text mining is of great significance and importance in the current period.

Some prominent studies on text mining are contained in Cohen and Hersh (2005) who conducted a survey to study some current work of text mining in biomedical science. The approaches and applications for text mining are discussed in Radovanovic and Ivanovic (2008). In addition, Gupta and Lehal (2009) conducted a survey of using different techniques on text mining with applications. Besides, Truyens and Van Eecke (2014) studied text mining in the direction of the legal aspects, and Vijayarani et al. (2015) presented an overview of preprocessing techniques for text mining.

Text mining continues to be studied very strongly as shown in Salloum et al. (2017) who conducted a survey of text mining in different social media. Text mining in organizational research is contained in Kobayashi et al. (2018). Recently, Pejic-Bach et al. (2020) have discussed job advertisements for the text mining of industry 4.0. Moreover, Hassani et al. (2020) introduced big-data text mining.

On the other hand, the statistical software R is an extremely ubiquitous software and is widely used in scientific research. It is very meaningful if the statistical software R is used to study text mining. Motivated by this, we present detailed instructions on how to use the statistical software R for text mining in this work. The progress of this work is described as follows. Section 2 briefly presents the origin of text mining. The algorithm for text mining is shortly introduced in Section 3. Section 4 presents the use of the statistical software R for text mining. Section 5 provides a specific application for text mining. Conclusions are shown in the last section.

Text mining

Since ancient times humans have sought to store contact information, information is often stored in its most primitive form is in text form. It is even easier to see that the amount of data that exists as text is much greater than that of other structured data. In fact, recent studies have shown that up to 80% of the information of organizations is documented. That can be documents, papers, forms, complaints, rights resolution, emails, information on websites. It is collectively referred to as the database. When studies of databases appeared in the 1960s, it was often thought that any contact information could be stored as structured data.

But in fact, after almost 50 years of extremely strong development, we still need to use archiving systems in the form of documents and even more often. Since then, text mining technology was formed and became an extremely popular tool for database analysis. Text mining is the process of extracting patterns and knowledge of valuable information from documents on text mining. This process is an extension of traditional text mining, as we already know traditional text mining is geared towards discovering knowledge from structured databases.

Both theory and applications of text mining have been presented in some documents as in Williams and Simoff (2006) provided the theory, methodology, techniques, and applications for data mining. Solka, J. L. (2008) presented theory and methods for text data mining. Zanini and Dhawan (2015) introduced the theory and some applications for text mining. Nowadays, with the very strong development of computational tools, highly applicable software has emerged. One of the popular software, easy to install and use is the statistical software R. Hence we have presented detailed and specific instructions on how to use R for text mining in this study.

Algorithm for text mining

In mathematics and many other sciences, an algorithm is a finite set of well-defined instructions. It can be done with a computer, often to solve some problems or to perform a certain calculation. Algorithms are frequently presented explicitly and are mostly used to specify the performance of calculations, data processing, and other inferences. A good algorithm is seen as an efficient approach, it will be presented in a well-defined formal language for the purpose of calculating and analyzing a particular problem.

Text mining is a process of processing and extracting information contained in the text that is part of text analysis in data mining. Text mining is a very important and meaningful issue in scientific research. So text mining algorithm is essential for all readers interested in this issue. Generally speaking, text mining is usually carried out through the following four steps:

Step 1: count the frequency of words in the text,

Step 2: clean up the text,

Step 3: building word cloud, and

Step 4: construct a chart of linking words.

As we know, the statistical software R is a tool for analyzing statistical graphs and data with a variety of statistical analysis techniques from linear and non-linear models. R is proposed by Ihaka and Gentleman (1996) and is widely used in applied science. It has statistical testing techniques, time series analysis, and countless other advanced algorithms like "machine learning" or "deep learning". Moreover, another notable strength of R software is its support for very flexible and quality graphing tools. Therefore it can be seen that this software is very practical and interesting for everyone doing scientific research.

Besides, the statistical software R is a completely free, open-source, royalty-free software that has a lot of data analysis features from statistics to finance, time series forecasting. Another very interesting thing is that it is always updated by researchers around the world, even readers can contribute to the development of R. Readers who are interested in using this software can be found in Feinerer (2008), Zuur et al. (2009), and Verzani (2018). Because of the outstanding advantages of R, thus it would be interesting to use the statistical software R to analyze the algorithm for text mining. Motivated by this, it will be detailed in the next section.

Using the statistical software R for text mining

In this section, we aim to present how to use statistical software R to analyze text mining through the four steps mentioned in the previous section.

Frequency of words in the text

A basic requirement in text mining analysis is to determine the frequency of words appearing and thereby can exploit the important information that the text wants to convey to the reader. This is a very important technique in statistics, it has a very diverse range of applications in practice and especially in cryptographic analysis.

In short, a frequency analysis is a method commonly used to analyze classical cryptography, it usually works by calculating the frequency of characters or groups of characters in the ciphertext and comparing it with the actual frequency in plain text. The packages and commands in R to do this can be summarized in **Table 1**.

Table 1. The commands to find frequency of words in the text.

A1	Install package	install.packages("tm") and install.packages ("wordcloud")
A2	Activate package	library("tm") and library("wordcloud")
A3	Set the directory to the path containing the desired file	setwd(".../directoryPath")
A4	Declaration text data	text.doc = readLines("textfile.txt", warn = FALSE)
A5	Statistics of the frequency of words in the text	docs.summ1 = VCorpus(VectorSource(text.doc))
		docs.summ2 = TermDocumentMatrix(docs.summ1)
		matrix = as.matrix(docs.summ2)
		vector = sort(rowSums(matrix), decreasing = TRUE)
		fre.table = data.frame(freq = vector)
A6	Show n the first line	head(fre.table, n)

where textfile.txt is represented as the name of the text file (.txt) to be analyzed, decreasing select TRUE: that is ordered in descending order, and n is used to represent the displayed number of rows in the table of the frequency of words.

Besides, it should be noted that, if we have operated steps A1-A4 (see **Table 1**) in Section 4.1, the following sections do not need to proceed, only need to do it once. Put plainly, we do not need to do steps A1-A4 in Sections 4.2, 4.3, and 4.4.

Clean up the text

As usual in the text, there will be special characters or some words, not important that you want to ignore during text mining. The main purpose of this is to help readers easily access the text because it removes words and characters that are not really necessary and important in the reading process. Besides, this work can help people read the text faster, and thereby will capture the main content of the text more accurately. The following commands are used to refine the text prior to analysis can be provided in **Table 2**.

Table 2. The commands to do clean up the text.

B1	Statistics of how often words appear in the text	docs.summ1 = VCorpus(VectorSource(text.doc))
B2	Replace characters or words with a space	toSpace = content_transformer(function(x, pattern) gsub(pattern, " ", x))
		docs.summ1 = tm_map(docs.summ1, toSpace, "letter")
B3	Convert uppercase characters to lower case	docs.summ1 = tm_map(docs.summ1,content_transformer(tolower))
B4	Remove the numbers	docs.summ1 = tm_map(docs.summ1, removeNumbers)
B5	Remove any strings	docs.summ1 = tm_map(docs.summ1, removeWords, c("string1", "string2", ..., "stringi"))
B6	Remove the dots	docs.summ1 = tm_map(docs.summ1, removePunctuation)
B7	Remove extra spaces	docs.summ1 = tm_map(docs.summ1, stripWhitespace)

where letter denotes special characters such as: /, @, #, and string1, .., stringi ie the ith string of characters. For example: the, which.

Building word cloud

A word cloud shows how often the words aim to visualize important keywords in the text. A word cloud is actually an image, as we know, a vivid picture worth more than thousands of words. Therefore, using images to emphasize the highlights, keywords of the text are very meaningful and essential in practice. The essence of images is to make things more receptive, making them represent a clear contrast between two things, one and the other, before and after. From there we can see that a word cloud is of great significance in text mining analysis. Besides, it also makes text readers enjoy, want to read them more, which is the intangible value that they bring to the reader of the text.

Naturally, the more frequency a word is used, the larger its size in the built cloud. Images can complement text and data without being gimmicky. By using more animated images, we can help the text reader reach the main content of the text as easily as possible. This is also the current main trend of most books and texts, the more images make them easier to understand, more accessible, leading to more readers. The following command is used to build a word cloud can be shortly presented below:

```
wordcloud(words = row.names(fre.table), freq = fre.table$freq, min.freq = 1, rot.per = 0.4,
max.words = 100, random.order = FALSE, colors = brewer.pal(n, "Dark2"))
```

where fre.table is denoted as the table of how often the words appear in the text, freq is the column of frequency, min.freq refers to the word with the smallest frequency displayed in the word cloud, rot.per denotes the proportion of words in total rotated by 90 degrees, max.words is the total number of words displayed in the word cloud in descending order of frequency, colors used to format the color of words (n from 1 to 8), random.order with TRUE: The words are displayed randomly, and FALSE: The words are displayed in descending order of frequency.

Constructing a chart of linking words

Similar to a word cloud, a chart of linking words is also in the form of images. It will help readers understand the main content quickly, help readers enjoy reading more text, save time reading text. Thus to show the frequency of words appearing in the text, the following functions and parameters will help us to retrieve information about the relationships, or the links between words in the text. First, we need to install the package: BiocManager with the following two commands:

install.packages ("BiocManager") and BiocManager::install("Rgraphviz")

Note that during the installation process with the above two commands, if asked to update the appropriate packages, we should choose to type "a" and press "enter" to update all packages.

```
Update all/some/none? [a/s/n]:  a  (Selecting a, to update all packages)
```

The following command is used to draw a chart of linking words:

```
freq.terms = findFreqTerms(x, lowfreq = 0),
```

```
plot(x, term = freq.terms, corThreshold = 0.4, weighting = TRUE, attrs = list(node = list(fillcolor = "green",fontsize = 40), edge = list(color = "black")))
```

where x denotes the resulting matrix from the TermDocumentMatrix function, lowfreq is used to specify the smallest frequency to be displayed on the link chart, weighting select TRUE: The thickness and boldness of the links represent the strength of the links between words.

A specific application for text mining

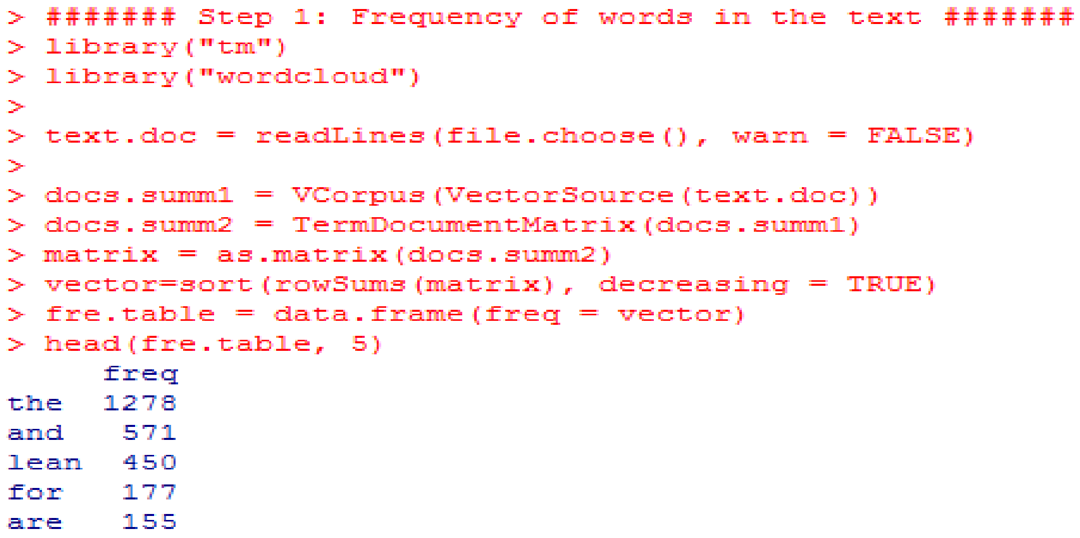
The real data set is executed in this work based on the master's thesis of Hien (2016). It is actually a short summary on a subject in medicine, to understand the meaning of this thesis quickly, we use the text mining algorithm mentioned in part 2 to analyze. Besides, the detailed analysis for this data set has also been mentioned in the Vietnamese version of the book "Data Mining with R" written by Hien et al. (2021).

Step 1: Frequency of words in the text.

```
rm(list=ls())          ### Clear the memory so that R runs best

install.packages("tm")

install.packages("wordcloud")

library("tm")

library("wordcloud")

text.doc = readLines(file.choose(), warn = FALSE)

docs.summ1 = VCorpus(VectorSource(text.doc))

docs.summ2 = TermDocumentMatrix(docs.summ1)

matrix = as.matrix(docs.summ2)

vector=sort(rowSums(matrix), decreasing = TRUE)

fre.table = data.frame(freq = vector)

head(fre.table, 5)
```

After using R to run the above code, the result is shown in **Figure 1**.

Figure 1. The result of Step 1: Frequency of words in the text.



```
> ####### Step 1: Frequency of words in the text #######
> library("tm")
> library("wordcloud")
>
> text.doc = readLines(file.choose(), warn = FALSE)
>
> docs.summ1 = VCorpus(VectorSource(text.doc))
> docs.summ2 = TermDocumentMatrix(docs.summ1)
> matrix = as.matrix(docs.summ2)
> vector=sort(rowSums(matrix), decreasing = TRUE)
> fre.table = data.frame(freq = vector)
> head(fre.table, 5)
      freq
the   1278
and    571
lean   450
for    177
are    155
```

As seen **Figure 1** that, the words "the", "and", "lean", "for" and "are" appear 1278, 571, 450, 177, and 155 times, respectively. Nevertheless, the words "the", "and", "for" and "are" are not keywords. Hence we can remove those words during text mining with the commands presented in the following step:

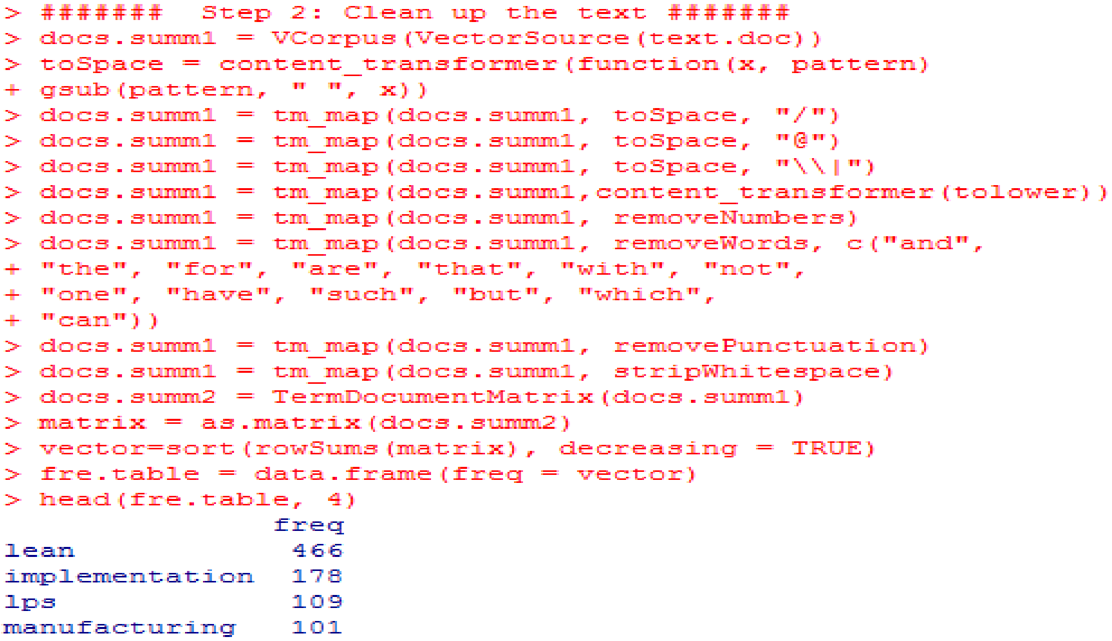
Step 2: Clean up the text.

```
docs.summ1 = VCorpus(VectorSource(text.doc))

toSpace = content_transformer(function(x, pattern)

gsub(pattern, " ", x))

docs.summ1 = tm_map(docs.summ1, toSpace, "/")

docs.summ1 = tm_map(docs.summ1, toSpace, "@")

docs.summ1 = tm_map(docs.summ1, toSpace, "\\|")

docs.summ1 = tm_map(docs.summ1,content_transformer(tolower))

docs.summ1 = tm_map(docs.summ1, removeNumbers)

docs.summ1 = tm_map(docs.summ1, removeWords, c("and", "the",
"for", "are", "that", "with", "not", "one", "have", "such",
"but", "which", "can"))
```

```
docs.summ1 = tm_map(docs.summ1, removePunctuation)

docs.summ1 = tm_map(docs.summ1, stripWhitespace)

docs.summ2 = TermDocumentMatrix(docs.summ1)

matrix = as.matrix(docs.summ2)

vector=sort(rowSums(matrix), decreasing = TRUE)

fre.table = data.frame(freq = vector)

head(fre.table, 4)
```

After using R to run the above code, the result is shown in **Figure 2**.

Figure 2. The result of Step 2: Clean up the text.

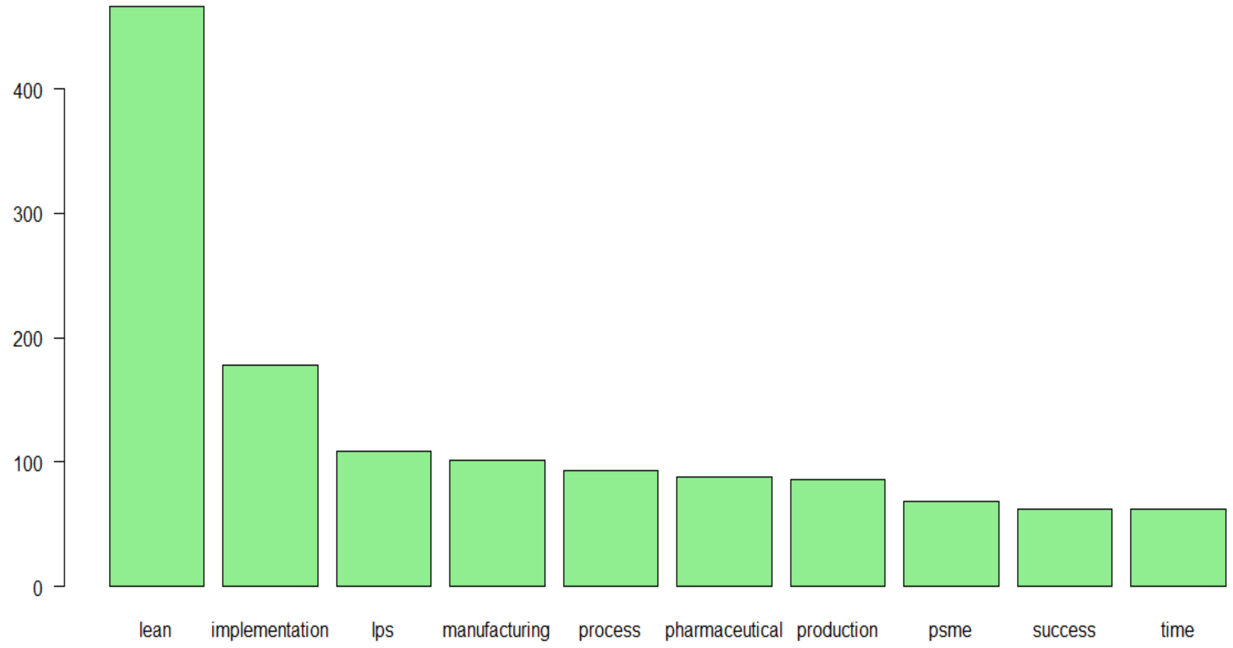


```
> #######  Step 2: Clean up the text #######
> docs.summ1 = VCorpus(VectorSource(text.doc))
> toSpace = content_transformer(function(x, pattern)
+ gsub(pattern, " ", x))
> docs.summ1 = tm_map(docs.summ1, toSpace, "/")
> docs.summ1 = tm_map(docs.summ1, toSpace, "@")
> docs.summ1 = tm_map(docs.summ1, toSpace, "\\|")
> docs.summ1 = tm_map(docs.summ1,content_transformer(tolower))
> docs.summ1 = tm_map(docs.summ1, removeNumbers)
> docs.summ1 = tm_map(docs.summ1, removeWords, c("and",
+ "the", "for", "are", "that", "with", "not",
+ "one", "have", "such", "but", "which",
+ "can"))
> docs.summ1 = tm_map(docs.summ1, removePunctuation)
> docs.summ1 = tm_map(docs.summ1, stripWhitespace)
> docs.summ2 = TermDocumentMatrix(docs.summ1)
> matrix = as.matrix(docs.summ2)
> vector=sort(rowSums(matrix), decreasing = TRUE)
> fre.table = data.frame(freq = vector)
> head(fre.table, 4)
                freq
lean             466
implementation   178
lps              109
manufacturing    101
```

As observed in **Figure 2** that, after cleaning text remove special case characters as well as unimportant words and switch from uppercase to lower case, then the words "lean", "implementation", "lps" and "manufacturing" appear 466, 178, 109 and 101 times, respectively.

Besides, readers can also display the 10 words with the most frequency in the text by the following command: barplot(vector[1:10], col = "lightgreen", las = 2), the result of this command is displayed in **Figure 3**.

Figure 3. The frequency of the 10 words appears most in the text.

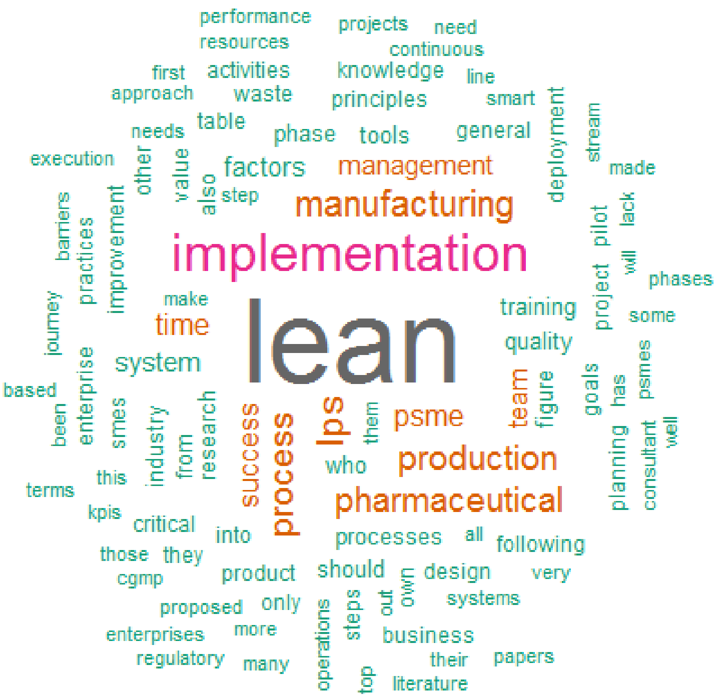


Step 3: Building word cloud.

```
wordcloud(words = row.names(fre.table), freq = fre.table$freq, min.freq = 1, rot.per = 0.4, max.words = 100, random.order = FALSE, colors=brewer.pal(8,"Dark2"))
```

, the result of this command is displayed in **Figure 4**.

Figure 4. A word cloud.

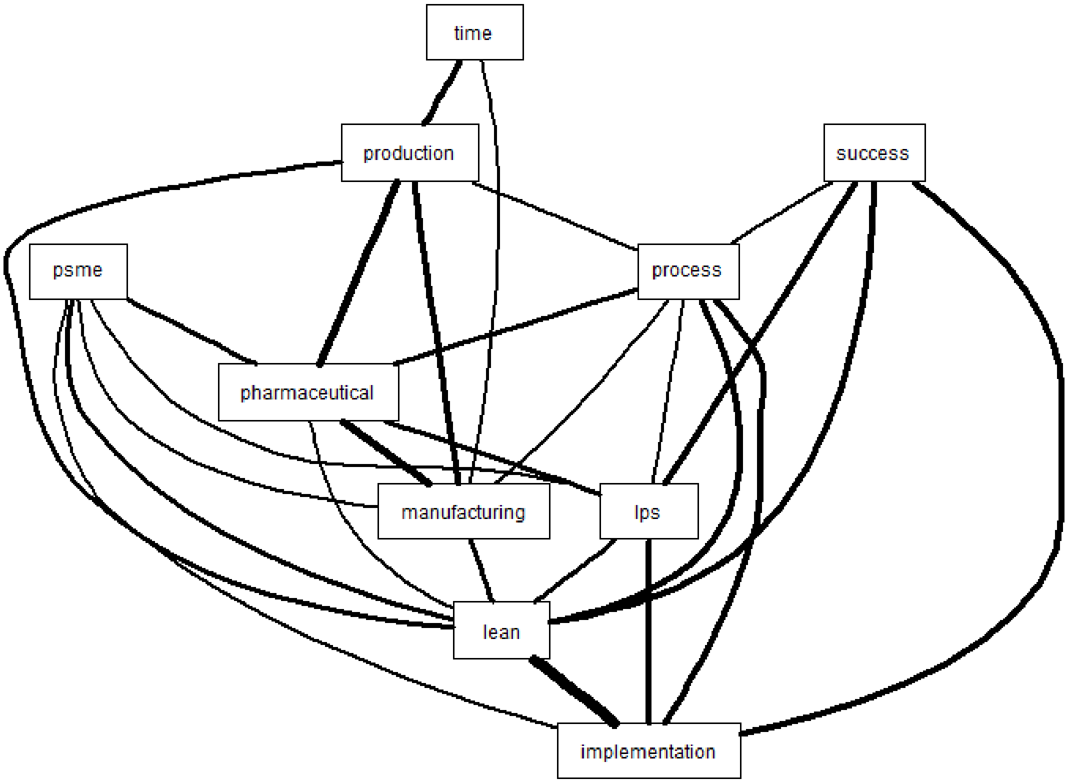


Step 4: Constructing a chart of linking words.

```
install.packages("BiocManager")

BiocManager::install("Rgraphviz")

library("graph")

library("Rgraphviz")

docs.summ2 = TermDocumentMatrix(docs.summ1)

freq.terms = findFreqTerms(docs.summ2, lowfreq = 62)

plot(docs.summ2,   term   =   freq.terms,   corThreshold   =   0.2,
weighting = TRUE)
```

After using R to run the above code, the result is shown in **Figure 5**.

Figure 5. A chart of linking words.



As seen in **Figure 5** that the chart shows that the text contains main content around the main keywords like "lean", "implementation", "lps", "manufacturing", "process", "pharmaceutical", and demonstrates a strong link between "lean" and "implementation", "manufacturing" and "pharmaceutical". Thereby, readers can confidently know that the main text of the text above is about "lean implementation in pharmaceutical manufacturing".

Conclusions

Text mining is an extremely crucial and significant issue in Statistics and many other Sciences because its application is easy to encounter in practice. This study has provided detailed and complete instructions on how to use the statistical software R for text mining. Our results will help readers access text mining through the statistical software R easily and quickly. In addition, we provided the text mining algorithm and covered in detail how to use the software R to approach each step of the algorithm. On the other hand, for practical applications, we have proposed the algorithm stated in Section 2 to study a research thesis on a problem in medicine. Based on the findings obtained in this study, it will help readers clearly understand how to use R statistics software to analyze text mining.

The primary contributions of this work are to introduce detailed and complete instructions on how to employ the statistical software R for text mining. To do this objective, we have presented the algorithm for text mining. Thereafter, we discuss how to employ the statistical software R to address each step of the algorithm. As an application, our proposed approach was implemented with an actual data set. Based on the results were found in this research, it will help readers and scientists understand how to employ the statistical software R to analyze text mining. Thus, this study is important for practitioners and academics in the research about text mining.

Readers should note that this paper could be extended in terms of applications in areas such as Decision Sciences, Statistical Software R, and Text Mining. Some typical studies include Abuelfadl (2017), Chang, et al. (2012, 2018), Gabrielsen, et al. (2015), Hau et al. (2020), Hieu et al. (2020), Li, et al. (2021), Lu, et al. (2018, 2021), McAleer (2021), Naseem et al. (2021), Nguyen and Vo (2019), Niu et al. (2021), Sigmund and Ferstl (2021), Truong, et al. (2019), Wang and Lo (2021), and many others. Readers may refer to Wong (2020), Tiwari et al. (2021), and Alghaith et al. (2021) for other important issues that the algorithm developed in this paper could apply for.

References

Abuelfadl, M. (2017). Individual foreign exchange investors, return predictability and market timing. *Annals of Financial Economics*, 12(01), 1750001.

Chang, C. L., McAleer, M., & Tansuchat, R. (2012). Modelling long memory volatility in agricultural commodity futures returns. *Annals of Financial Economics*, 7(02), 1250010.

Chang, C. L., McAleer, M., & Wong, W. K. (2018). Decision sciences, economics, finance, business, computing, and big data: Connections. *Advances in Decision Sciences*, 22(A), 1-58.

Cohen, A. M., & Hersh, W. R. (2005), A survey of current work in biomedical text mining. *Briefings in bioinformatics*, 6(1), 57-71.

Feinerer, I. (2008). An introduction to text mining in R. *R News*, 8(2), 19-22.

Gabrielsen, A., Kirchner, A., Liu, Z., & Zagaglia, P. (2015). Forecasting value-at-risk with time-varying variance, skewness and kurtosis in an exponential weighted moving average framework. *Annals of Financial Economics*, 10(01), 1550005.

Gupta, V., & Lehal, G. S. (2009), A survey of text mining techniques and applications. *Journal of Emerging Technologies in Web Intelligence*, 1(1), 60-76.

Hassani, H., Beneki, C., Unger, S., Mazinani, M. T., & Yeganegi, M. R. (2020). Text mining in big data analytics. *Big Data and Cognitive Computing*, 4(1), 1-34.

Hau, N. H., Tinh, T. T., Tuong, H. A., & Wong, W. K. (2020). Review of matrix theory with applications in education and decision sciences. *Advances in Decision Sciences*, 24(1), 1-41.

Hien, N. N (2016). Implementation of lean production systems for small-medium sized enterprises (Unpublished master's thesis). Vietnamese-Germany University, Vietnam.

Hien, N. N., Nhan, H. H., Hung, P. K., & Cuong, N. T. (2021). *Khai thac du lieu voi R* (1st ed.). Ho Chi Minh City, Vietnam: Thanh Nien Pusblishing House. ISBN: 978-604-334-956-6.

Hieu, N. T., Huy, L. M., Phat, H. M., Anh, N. N. P., & Wong, W. K. (2020). Decision sciences in education: The STEMtech model to create stem products at high schools in Vietnam. *Advances in Decision Sciences*, 24(2), 1-50.

Ihaka, R., & Gentleman, R. (1996). R: a language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5(3), 299-314.

Kobayashi, V. B., Mol, S. T., Berkers, H. A., Kismihók, G., & Den Hartog, D. N. (2018). Text mining in organizational research. *Organizational Research Methods*, 21(3), 733-765.

Li, H., Bai, Z., Wong, W. K., & McAleer, M. (2021). Spectrally-corrected estimation for high-dimensional Markowitz mean-variance optimization. Econometrics and Statistics. Forthcoming.

Lu, R., Hoang, V. T., & Wong, W. K. (2021). Does Lump-Sum Investing Strategy Outperform Dollar-Cost Averaging Strategy in Uptrend Markets?, Studies in Economics and Finance, forthcoming.

Lu, R., Yang, C. C., & Wong, W. K. (2018). Time diversification: Perspectives from the economic index of riskiness. *Annals of Financial Economics* 13(3), 1850011.

McAleer, M. (2021). A critique of recent medical research in JAMA on COVID-19. *Advances in Decision Sciences*, 25(1), 1-102.

Naseem, U., Khushi, M., Khan, S. K., Shaukat, K., & Moni, M. A. (2021). A comparative analysis of active learning for biomedical text mining. *Applied System Innovation*, 4(1), 23.

Nguyen, T. D. T., & Vo, D. H. (2019). The determinants of systematic risk in Vietnam. *Advances in Decision Sciences*, 23(2), 1-21.

Niu, M., Wandy, J., Daly, R., Rogers, S., & Husmeier, D. (2021). R package for statistical inference in dynamical systems using kernel based gradient matching: KGode. *Computational Statistics*, 36(1), 715-747.

Radovanović, M., & Ivanović, M. (2008). Text mining: Approaches and applications. *Novi Sad Journal of Mathematics*, 38(3), 227-234.

Salloum, S. A., Al-Emran, M., Monem, A. A., & Shaalan, K. (2017). A survey of text mining in social media: facebook and twitter perspectives. *Advances in Science, Technology and Engineering Systems Journal*, 2(1), 127-133.

Sigmund, M., & Ferstl, R. (2021). Panel vector autoregression in R with the package panelvar. *Quarterly Review of Economics and Finance*, 80, 693-720.

Solka, J. L. (2008). Text data mining: theory and methods. *Statistics Surveys*, 2, 94-112.

Truong, B. C., Van Thuan, N., Hau, N. H., & McAleer, M. (2019). Applications of the Newton-Raphson method in decision sciences and education. *Advances in Decision Sciences*, 23(4), 1-28.

Truyens, M., & Van Eecke, P. (2014). Legal aspects of text mining. *Computer Law and Security Review*, 30(2), 153-170.

Pejic-Bach, M., Bertoncel, T., Meško, M., & Krstić, Ž. (2020). Text mining of industry 4.0 job advertisements. *International Journal of Information Management*, 50, 416-431.

Verzani, J. (2018). Using R for introductory statistics. *CRC press*.

Vijayarani, S., Ilamathi, M. J., & Nithya, M. (2015). Preprocessing techniques for text mining-an overview. *International Journal of Computer Science and Communication Networks*, 5(1), 7-16.

Wang, L. L., & Lo, K. (2021). Text mining approaches for dealing with the rapidly expanding literature on COVID-19. *Briefings in Bioinformatics*, 22(2), 781-799.

Wong, W. K. (2020). Review on behavioral economics and behavioral finance. *Studies in Economics and Finance*. https://doi.org/10.1108/SEF-10-2019-0393.

Williams, G. J., & Simoff, S. J. (2006). Data mining: Theory, methodology, techniques, and applications. Springer.

Zanini, N., & Dhawan, V. (2015). Text Mining: An introduction to theory and some applications. *Research Matters*, 19, 38-45.

Zuur, A., Ieno, E. N., & Meesters, E. (2009). A beginner's guide to R. *Springer Science and Business Media*.