2ND ANNUAL BIG DATA & ANALYTICS SUMMIT CANADA

Optimize your business value NOW!

# Text Mining 101: Basic Tools for Big Data Novices

Sujoy Paul

Alberta Health Services

Manager, Database Services

Alberta Health Services

# Goals of Workshop

Definition of text mining

- Definition of text mining

Theoretical aspects of text mining

- Theoretical aspects of text mining

Applications of text mining

- Applications of text mining

Hands-on tutorial on text mining using R

- Hands-on tutorial on text mining using R

# Definition of Text Mining

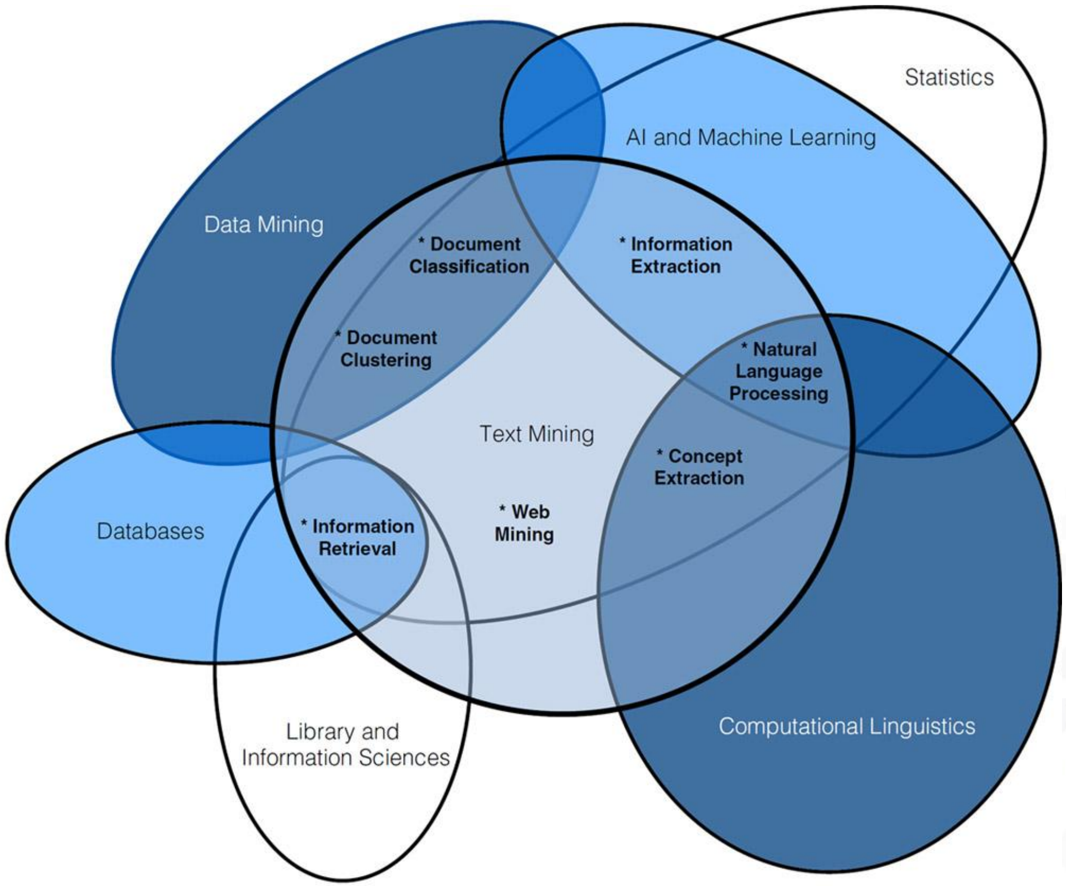
From Wikipedia(1), the free encyclopedia:

- **Text mining**, also referred to as *text data mining*, roughly equivalent to **text analytics**, refers to the process of deriving high-quality information from text. High-quality information is typically derived through the devising of patterns and trends through means such as statistical pattern learning. Text mining usually involves the process of structuring the input text (usually parsing, along with the addition of some derived linguistic features and the removal of others, and subsequent insertion into a database), deriving patterns within the structured data, and finally evaluation and interpretation of the output. 'High quality' in text mining usually refers to some combination of relevance, novelty, and interestingness. Typical text mining tasks include text categorization, text clustering,concept/entity extraction, production of granular taxonomies, sentiment analysis, document summarization, and entity relation modeling (*i.e.*, learning relations between named entities).
- Text analysis involves information retrieval, lexical analysis to study word frequency distributions, pattern recognition,tagging/annotation, information extraction, data mining techniques including link and association analysis, visualization, and predictive analytics. The overarching goal is, essentially, to turn text into data for analysis, via application of natural language processing (NLP) and analytical methods.

2ND ANNUAL BIG DATA & ANALYTICS SUMMIT CANADA

Optimize your business value NOW!

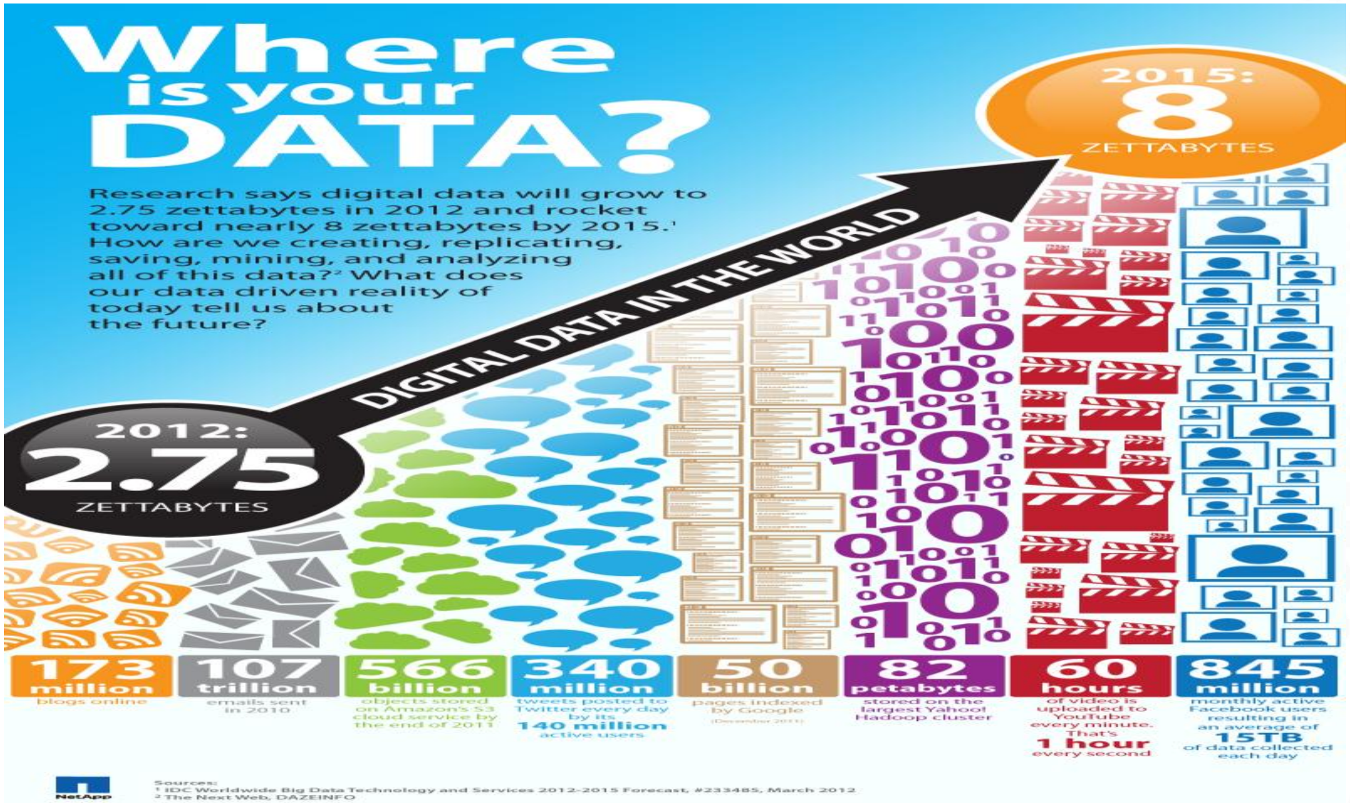
# What is Text Mining ?




**Venn Diagram showing the inter-disciplinary nature of Text Mining (2)**

2ND ANNUAL
BIG DATA & ANALYTICS
SUMMIT CANADA
Optimize your business value NOW!

# How much Data ? (3)



# Amount of Data

- Most of the data is unstructured
- Computer World states that unstructured information might account for more than 70%–80% of all data in organizations
- So does Gartner: 80%
- Unstructured data  is “content that does not have a pre-defined data model” (4)

# Structured /Unstructured Data

- Unstructured data are of 2 types (2) :

Semi-structured and Weak-structured

- Semi-structured is defined as:

documents with extensive and consistent format elements in which field-type metadata can be more easily inferred –files with heavy document templating/style-sheet constraints. Eg available in Word, PPT, PDF, email, HTML

# Weak-structured

- Weak-structured:

  Most scientific research papers, business reports, legal memoranda, and news stories. Since they have little in the way of strong typographical, layout, or markup indicators to denote structure

# Need for Text Mining

- Humans cannot deal with information overload
- Need to structure data and automate:
- Synthesize data from varied sources
- Find valid information and integrate knowledge
- Build structure to generate knowledge
- Help establish links from various sources eg concept extraction

# Practice Areas of Text Mining (2)

- **Search and information retrieval (IR):** Storage ,retrieval of text documents, including search engines and keyword search.
- **Document clustering:** Grouping and categorizing terms, snippets, paragraphs, or documents, using data mining clustering methods.
- **Document classification:** Grouping and categorizing snippets, paragraphs, or documents, using data mining classification methods, based on models trained on labeled examples.
- **Web mining:** Data and text mining on the Internet, with a specific focus on the scale and interconnectedness of the web.

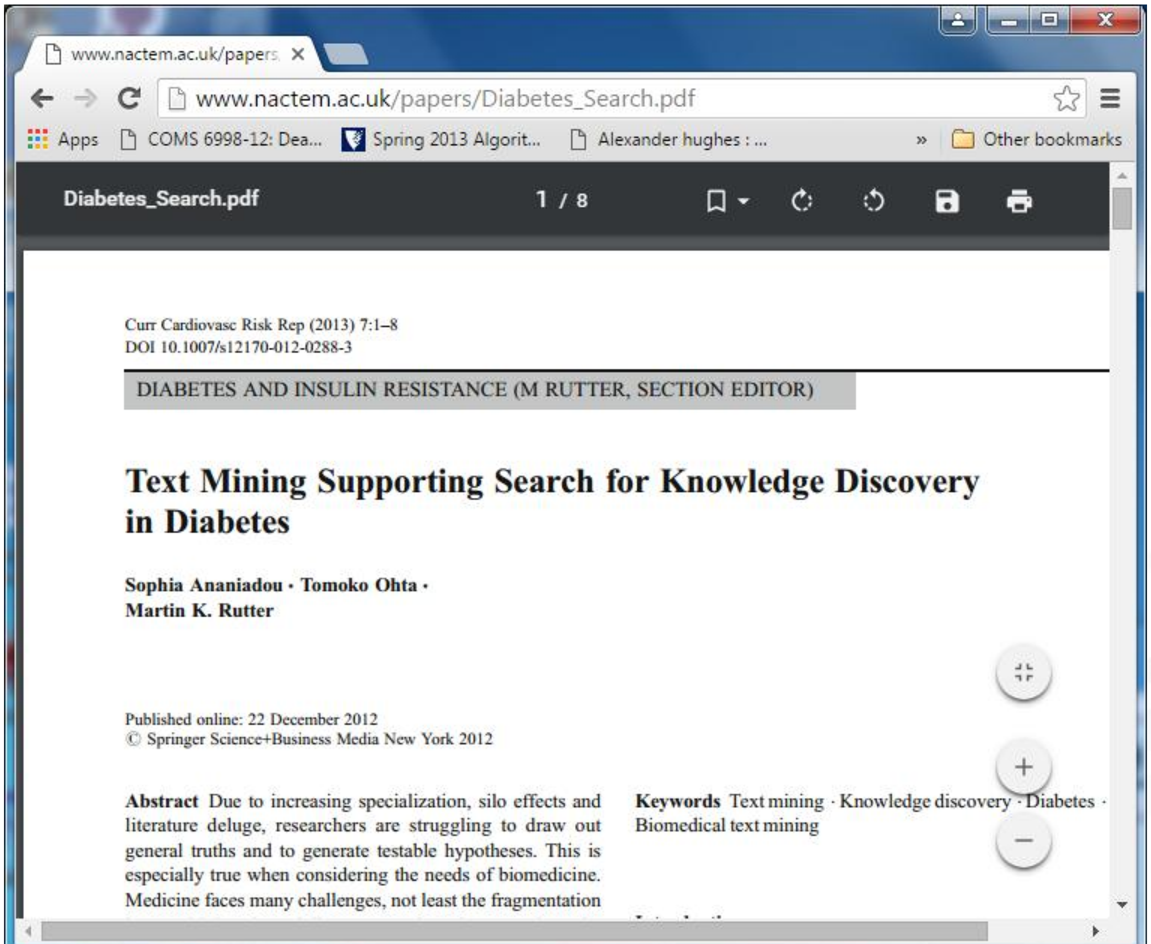
# Areas of Text Mining

- **Information extraction (IE):** Identification and extraction of relevant facts and relationships from unstructured text; the process of making structured data from unstructured and semistructured text.
- **Natural language processing (NLP):** Low-level language processing and understanding tasks (e.g., tagging part of speech); often used synonymously with computational linguistics.
- **Concept extraction:** Grouping of words and phrases into semantically similar groups.

# Applications of Text Mining

- Extracting "meaning" from unstructured text:
  Sentiment analysis, fraud detection, warranty claims
- Automatic categorization of Text:
  by summarizing the data in a document
- Used in medicine: Toxicity prediction, associations between diseases, associations between genes and diseases
- Text Analytics and Taxonomies for Fraud and Abuse Detection in Medical Insurance Claims

# Text mining in Biomedical Domain




Curr Cardiovasc Risk Rep (2013) 7:1–8
DOI 10.1007/s12170-012-0288-3

DIABETES AND INSULIN RESISTANCE (M RUTTER, SECTION EDITOR)

**Text Mining Supporting Search for Knowledge Discovery in Diabetes**

**Sophia Ananiadou · Tomoko Ohta · Martin K. Rutter**

Published online: 22 December 2012


**Abstract** Due to increasing specialization, silo effects and literature deluge, researchers are struggling to draw out general truths and to generate testable hypotheses. This is especially true when considering the needs of biomedicine. Medicine faces many challenges, not least the fragmentation

**Keywords** Text mining · Knowledge discovery · Diabetes · Biomedical text mining

2ND ANNUAL
BIG DATA & ANALYTICS
SUMMIT CANADA Optimize your business value NOW!

# Text Mining in Cancer Research


Methodological Review

## Biomedical text mining and its applications in cancer research

Fei Zhu [a,b], Preecha Patumcharoenpol [c,d], Cheng Zhang [a], Yang Yang [a,b], Jonathan Chan [c], Asawin Meechai [e], Wanwipa Vongsangnak [a], Bairong Shen [a,f,*]

[a] Center for Systems Biology, Soochow University, Suzhou 215006, China
[b] School of Computer Science and Technology, Soochow University, Suzhou 215006, China
[c] School of Information Technology, King Mongkut's University of Technology Thonburi, Thailand
[d] School of Bioresources and Technology, King Mongkut's University of Technology Thonburi, Thailand
[e] Department of Chemical Engineering, King Mongkut's University of Technology Thonburi, Thailand
[f] Institute for Translational Bioinformatics and Systems Medicine, School of Biomedical Informatics, Suzhou University of Science and Technology, Jiangsu 215009, China

ARTICLE INFO

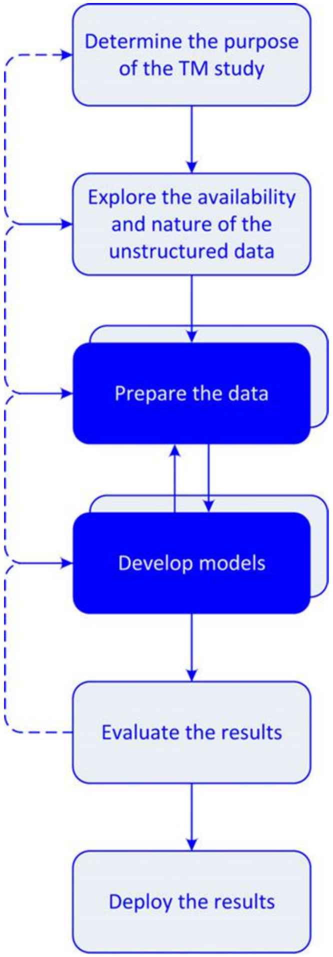
*Article history:*
Received 28 March 2012
Accepted 30 October 2012
Available online 15 November 2012

*Keywords:*
Biomedical text
Cancer
Systems biology
Text mining

ABSTRACT

Cancer is a malignant disease that has caused millions of human deaths. Its study has a long history of well over 100 years. There have been an enormous number of publications on cancer research. This integrated but unstructured biomedical text is of great value for cancer diagnostics, treatment, and prevention. The immense body and rapid growth of biomedical text on cancer has led to the appearance of a large number of text mining techniques aimed at extracting novel knowledge from scientific text. Biomedical text ...g on cancer research is computationally automatic and high-throughput in nature. However, it is err... due to the complexity of natural language processing. In this review, we introduce the basic c...pts underlying text mining and examine some frequently used algorithms, tools, and data sets, as well as assessing how much these algorithms have been utilized. We then discuss the current state-of-the-art text mining applications in cancer research and we also provide some resources for cancer text minir... ...h the development of systems biology, researchers tend to understand complex biomedical syste... a systems biology viewpoint. Thus, the full utilization of text mining to facilitate cancer systems ...gy research is fast becoming a major concern. To address this issue, we describe the general workflow of text mining in cancer systems biology and each phase of the workflow. We hope that this review can (i) ...e a useful overview of the current work of this field; (ii) help researchers to choose text mining t... ...d datasets; and (iii) highlight how to apply text mining to assist cancer systems biology research.

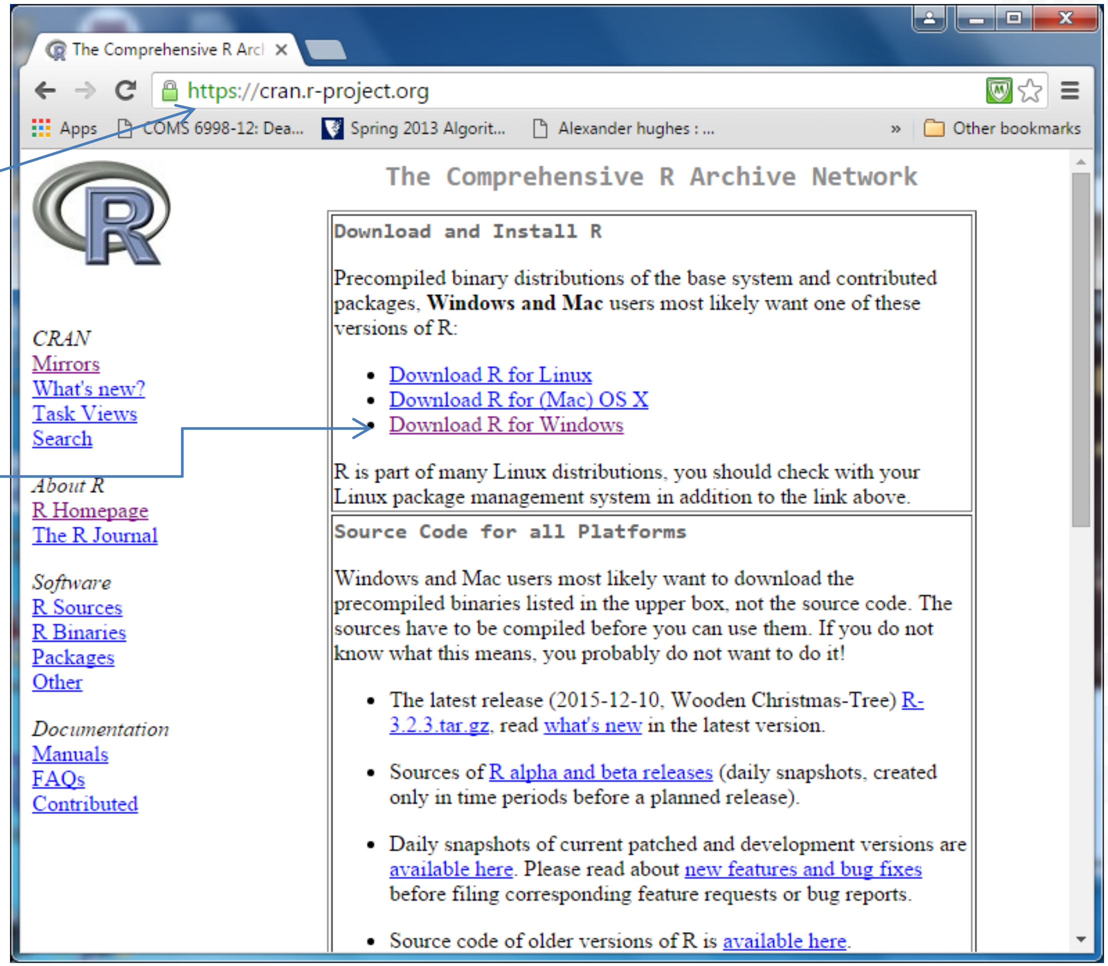
# Text Mining Process Flow (2)



# Download R

URL site

Win download

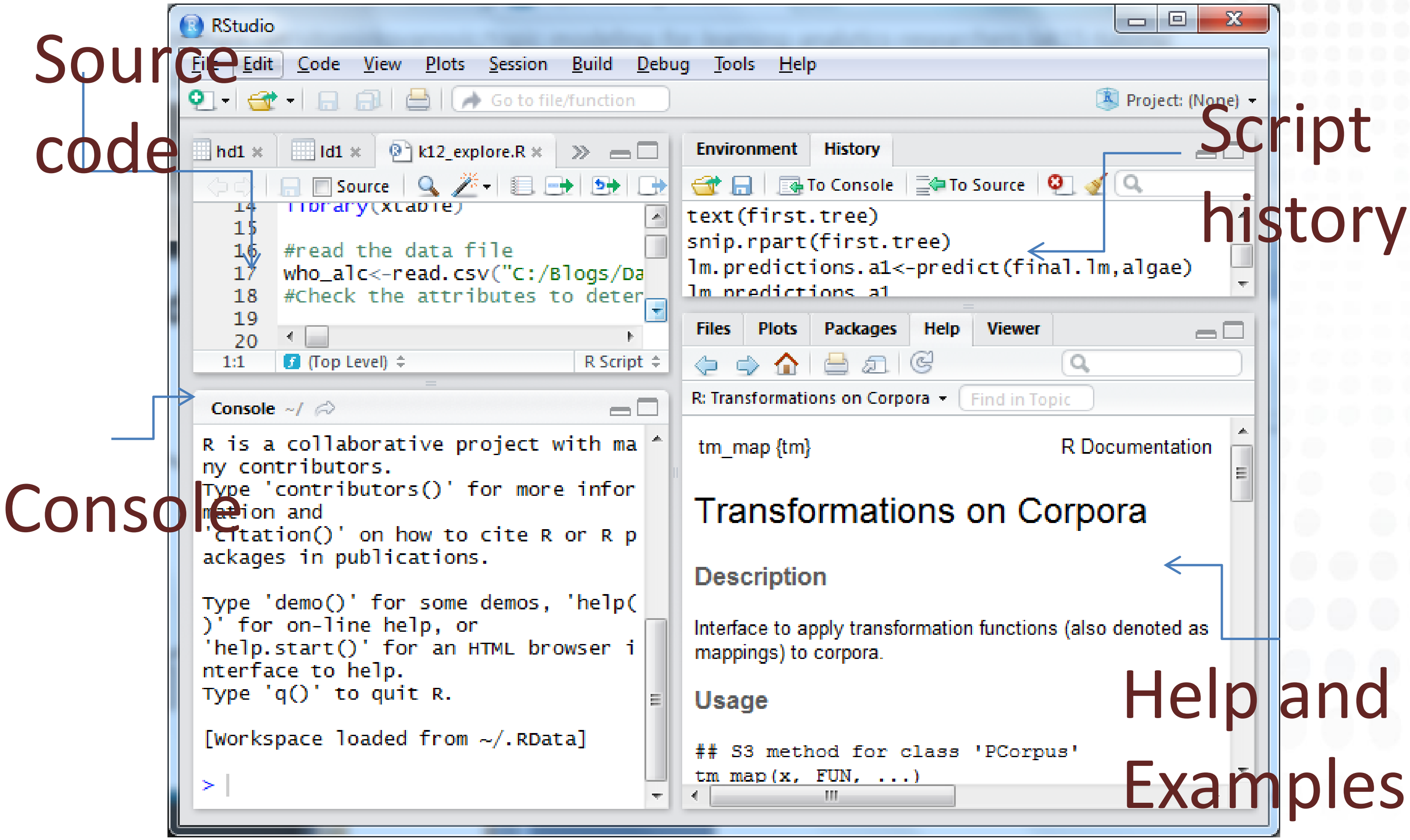


# RStudio

- IDE a powerful and user interface for R
- Open source and works for Windows, Linux, MAC
- Download from http://www.rstudio.com

2ND ANNUAL BIG DATA & ANALYTICS SUMMIT CANADA
Optimize your business value NOW!

# RStudio IDE



# R Basics

- To find the current wd:

  >getwd()
- To set to a specific directory:

  >setwd("C:/rfiles/")
- R makes extensive use of libraries
- Libraries are essentially packages designed to perform a collection of specific functions eg tm
- Library(), search(), install.library("package"), library("package")

# Scripts

- A text file with R commands
- R is an interpretive language and not a complied language.  It is also case-sensitive
- Can use any of the text editors for batch processing or with R studio
- To run a script containing commands from an external file from RStudio

  >studio("textmine.R")
- To direct the output of all commands from the console to the file

  >sink("textmine_out.R")

# R Graphics

- Packages: primarily ggplot2; historically:lattice
- Both basic graphs for exploratory data analysis: box plots, density plots, scatter plots
- Advanced graphs : customization, legends, axes, statistical plots like probability plots

# Commands

- can write any comments between #...#
- The entities that R creates and manipulates are known as objects: variables,arrays of numbers, character strings, functions
- >ls() shows the current objects in the workspace
- <rm(obj1,obj2) – removes objects obj1 and obj2

# R Libraries for Text Mining

- Tm: framework for text mining
- Ggplot2: new graphical package
- Wordcloud: generating the wordcloud
- SnowballC: Stemming of words
- Rgraphviz: plotting correlation
- Rcolorbrewer:provides color schemes for graphics

# Variables

- Variables and expressions

```
x <- c(2, 4.6, 5.3, 12, 22.5,32) #c: concatenation
 > x [1] 2.0 4.6 5.3 12.0 22.5 32.0
```

Variable names can also be stated as a.b eg

```
> one.twenty<-seq(from=1,to=20)
>one.twenty
 [1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
   19 20 > one.twenty[5]
 [1] 5
```

# Different Vectors (5)

- Vectors  are contiguous cells containing data. Cells are accessed through indexing operations such as x[2] – indexing starts from 1
- Lists: elements do not have to be of the same type
- Arrays: vectors plus the dim attribute , matrices are arrays with a dim attribute of length 2.
- Factors: handle nominal and ordered categorical data
- Factors : describe items that can have a finite number of values (gender, social class, etc.).

# Data Frames

- Representation of data in a table format
- Matrix like structures with rows and columns in which columns can be different types

```
>b.boolean=c(TRUE,FALSE,TRUE)
>s.names=c("Mary","Bob","Jill")
> df=data.frame(one.three,b.boolean,s.names)
> df
```

one.three b.boolean s.names 1 1 TRUE Mary 2 2 FALSE Bob 3 3 TRUE Jill

# Data Frames

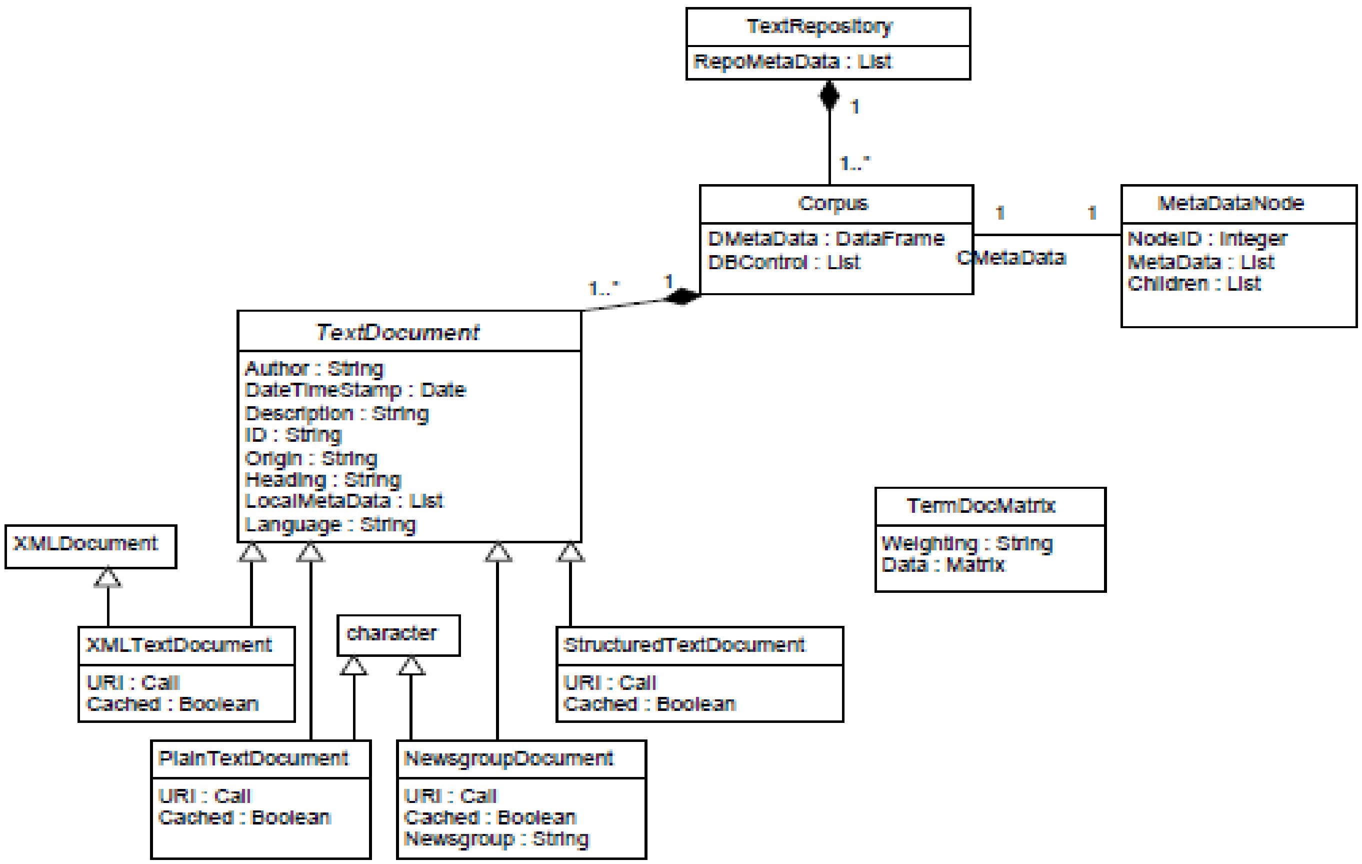
- To find the 1st row of all the columns for the in-house data frame mtcars

>mtcars[1,]

mpg cyl disp hp drat wt qsec vs am gear carb
Mazda RX4 21 6 160 110 3.9 2.62 16.46 0 1 4 4

2ND ANNUAL
BIG DATA & ANALYTICS
SUMMIT CANADA
Optimize your business value NOW!

# Tm package (Class diagram)(6)



# Term Document Collection

- Also known as Corpus
- Electronic collection of text documents.
- Holds both the actual text and the metadata.
- The tm package supports different formats including : PDF, DOC, XML, TXT

# Corpus Readers

Tm supports different formats of readers

>getReaders()

[1] "readDOC" "readPDF" [3] "readPlain" "readRCV1" [5] "readRCV1asPlain" "readReut21578XML" [7] "readReut21578XMLasPlain" "readTabular" [9] "readXML”

# Readers available

| Name of Reader | Description |
|---|---|
| readDoc | Read in MS Word documents |
| readPDF | Read in Adobe PDF documents |
| readPlain | Read in plain text ignoring metadata |
| readRCV1 | Read in Reuters Corpus Volume 1 XML format |
| readRCV1asPlain | Read in a Reuters Corpus Volume 1 XML document |
| readReut21578XML | Read in Reuters-21578 XML format |
| readReut21578XMLasPlain | Read in Reuters-21578 XML format |
| readTabular | Read in a text document from a tabular data structure (like a data frame or a list matrix) |
| readXML | Read in XML documents |

# Corpus Sources

Sources supported by the tm package

```
>getSources()
[1] "DataframeSource" "DirSource" "URISource"
[4] "VectorSource" "XMLSource"
```

# Exploring the Corpus

Inspect that the data has been loaded

```
>inspect(docs[1])
```

# Pre-Processing

- Once data is loaded in the Corpus, pre-processing  or cleaning up the data is necessary to ready the data for analysis
- This is done by employing the getTransformations() function

```
>getTransformations()
[1] "removeNumbers" "removePunctuation" [3] "removeWords" "stemDocument" [5] "stripWhitespace"
```

# Transformations

- Use tm_map() function for transformation
- *#clean the text -- transformation* docs
- Docs <- tm_map(docs, removePunctuation) *# Removing punctuation#*
- docs <- tm_map(docs, removeNumbers) *# Removing numbers#*
- docs <- tm_map(docs, tolower) *# Converting to lowercase #*

# Transformations

- *docs* <- tm_map(docs, removeWords, stopwords("english"))
- docs <- tm_map(docs, stemDocument) *#Removing common word endings (e.g., "ing", "es") #*
- docs <- tm_map(docs, stripWhitespace) # *Stripping whitespace #*
- docs <- tm_map(docs, PlainTextDocument)

# Stopwords

- Stopwords are words that have low information value ie entropy is low
- Removing stopwords reduces dimensionality

```
>length(stopwords("english"))
[1] 174
```

- Some of the stopwords are

```
>stopwords("english")
 [1] "i" "me" "my" "myself“
```

# Stemming

- Normalizes variations of the word eg talking, talked, talks.  Equivalent would be talk
- Removes suffixes from words eg. “er”, “es”, “ed”
- Purpose is to remove complexity
- Infectional stemming:

  Remove plurals, normalize verb tenses

# Stemming(continued)

- Use wordstem() function in the SnowballC package for stemming

> getStemLanguages()

[1] "danish" "dutch" "english" "finnish"

[5] "french" "german" "hungarian" "italian“

[9] "norwegian" "porter" "portuguese " " romanian “

[13] "russian" "spanish" "swedish" "turkish“

# Specific Transformations

- Use content_transformer() to build transformational functions specific to the requirements

```
>toString <- content_transformer(function(x, from, to) gsub(from, to, x))
>mydocs1<-tm_map(mydocs1,toSpace,"'")
>mydocs1<-tm_map(mydocs1,toSpace,"-")
```

# Count based Evaluation

- Find terms with highest frequencies
- Use Document- Term matrix (DTM)
- Rows contain documents and columns contain the terms
- Count of the frequency of the words as cells in the matrix
- Transpose of the DTM is known as Term Document Matrix(TDM)

# Document Term Matrix

# build the Document Term Matrix and/or the Term Document Matrix

```
>dtm <- DocumentTermMatrix(docs)
>tdm <- TermDocumentMatrix(docs)
```

# Download Data Sources

- Data sources to be used are:
- Chapter 1 – ch01.txt
- Chapter 2 – ch02. txt
- These are chapters from my book: Pro SQL Server Replication 2005

# Initialize libraries

- #initialize the library
- libs<-c("tm","plyr","class","wordcloud",
 "SnowballC","Rgraphviz","ggplot2")
- lapply(libs,require,character.only=TRUE)

# Corpus Loading

```
#loading the corpus and summary the mydocs1
mydocs1 <-Corpus(DirSource("C:/conf"))
#specify the source to be character vectors
mydocs1<-Corpus(VectorSource(mydocs1))

mydocs1
<<VCorpus (documents: 2, metadata
  (corpus/indexed): 0/0)>>
```

# Corpus Summary

summary(mydocs1)

Head(mydocs1)

<<VCorpus (documents: 2, metadata (corpus/indexed): 0/0)>>

head(mydocs1[[2]])

$content [1] "CHAPTER 2" [2] "Replication Basics" [3] "In the previous chapter, I introduced replication as a method of distributing data.. ......

# Pre-processing Corpus

```
# Remove unwanted characters using gsub function and the tm_map function
toSpace<-content_transformer(function(x,pattern)
{return (gsub(pattern," ",x))})
mydocs1<-tm_map(mydocs1,toSpace,""")
mydocs1<-tm_map(mydocs1,toSpace,"-")
mydocs1<-tm_map(mydocs1,toSpace,"_")
mydocs1<-tm_map(mydocs1,toSpace,":")
```

# Transformation

```
mydocs1 <- tm_map(mydocs1, removePunctuation)
mydocs1 <- tm_map(mydocs1, removeNumbers)
mydocs1 <- tm_map(mydocs1, tolower)
mydocs1 <- tm_map(mydocs1, removeWords, stopwords("english"))
mydocs1 <- tm_map(mydocs1, stemDocument) # *Removing common word endings* (e.g., "ing", "es")
```

# Further Transformation

```
mydocs1 <- tm_map(mydocs1, stripWhitespace)
mydocs1 <- tm_map(mydocs1,
   PlainTextDocument)
#remove stop words like note
mydocs1 <- tm_map(mydocs1, removeWords,
   c("chapter",
   "figure","note","tip","caution","table"))
##run inspect(mydocs1) again to check how the
   corpus looks
inspect(mydocs1)
```

# Document Term Matrix

```
dtm <- DocumentTermMatrix(mydocs1)

tdm <- TermDocumentMatrix(mydocs1)

#run dtm to check sparsity

Dtm

<<DocumentTermMatrix (documents: 2, terms: 1494)>> Non-/sparse entries: 1858/1130
Sparsity : 38% Maximal term length: 40
Weighting : term frequency (tf)
```

# Exploring the DTM

```
#explore the frequencies of the words
freq <- colSums(as.matrix(dtm))
  #convert the DTM into matrix and
  then sum the column
length(freq)
[1] 1494
#find the most frequencies of the words
freq_words<-sort(freq,decreasing=TRUE)
```

2ND ANNUAL BIG DATA & ANALYTICS SUMMIT CANADA
Optimize your business value NOW!

# Investigating DTM

head(table(freq_words),20)

server replication sql publisher data database

329 188 135 131 129 119

**can** distributor distribution will agent distributed

**117** 114 111 91 90 64

set servers transaction shown subscriber also

62 61 60 58 57 52

name different 52 49

# Distribution of Term Frequencies

```
#find the least frequency words
ord <-order(freq)
freq[tail(ord)]
database data publisher sql replication server
119 129 131 135 188 329
```

# Change the DTM

```
#lets include words that are more than 3
   characters and less than 15
dtm <-DocumentTermMatrix(mydocs1,
   control=list(wordLengths=c(4, 15)))
#Run the frequencies again
freq <- colSums(as.matrix(dtm))
length(freq)
[1] 1388
```

2ND ANNUAL BIG DATA & ANALYTICS SUMMIT CANADA Optimize your business value NOW!

# Frequency of Words

```
#order the frequency to find the least frequency
ord <-order(freq)
freq[tail(ord)]
distributor database data publisher replication
   server
 114 119 129 131 188 329
#find the most frequencies of the words
freq_words<-sort(freq,decreasing=TRUE)
```

# Frequency of Words(new)

head(freq_words,20)

server replication publisher data database

329 188 131 129 119

distributor distribution will agent distributed

114 111 91 90 64

servers transaction shown subscriber also

61 60 58 57 52

name different databases model subscriptions

52 49 46 45 43

# Wordclouds

```
#find frequent terms
freq.terms<-findFreqTerms(dtm, lowfreq=50)
#plot the wordcloud
dark2 <- brewer.pal(6, "Dark2")
#plot the 50 most frequently used words
wordcloud(names(freq), freq, max.words=50,
  rot.per=0.2, colors=dark2)
```

# Purpose of WordClouds

"The advantage of word clouds is that this visualization is not biased by the use of a predefined set of concepts or an ontology, but is driven by the raw content of the text. As such they can provide new ideas and insights on a particular concept and can function as a starting point for more specific searches"(7)

# WordClouds >50

# Wordclouds

```
#plot words that occur at least 50 times
wordcloud(names(freq), freq, min.freq=50,
   rot.per=0.2, colors=dark2)
```

2ND ANNUAL
BIG DATA & ANALYTICS
SUMMIT CANADA
Optimize your business value NOW!

# Wordclouds (50 times)

# Finding Associations

#find associations of word merge with correlation of 0.95

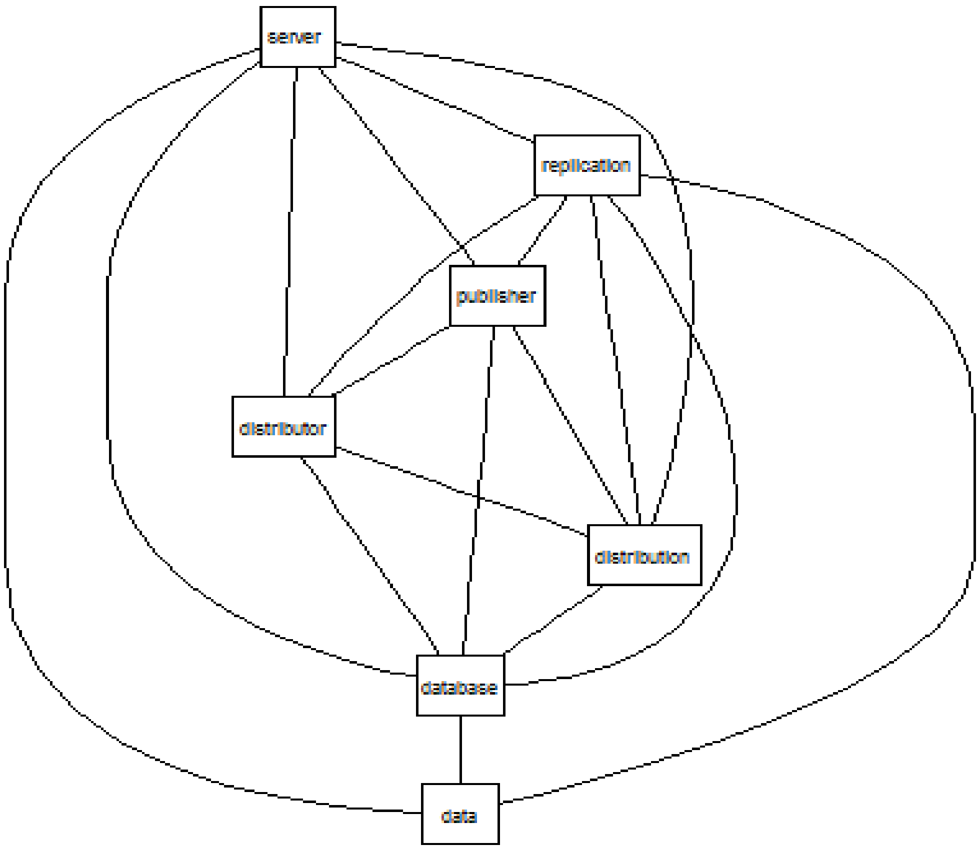
findAssocs(dtm,”merge",corlimit=0.95)

merge able 1.00 acts 1.00 book 1.00 called 1.00 check 1.00 complete 1.00 configuration 1.00 depending 1.00 discussed 1.00 done 1.00 executed 1.00 file 1.00 however 1.00 local 1.00 multiple 1.00 regional 1.00 service 1.00 shops 1.00 using 1.00 account 0.99 also 0.99 cases 0.99 chapters 0.99 compon 0.99 consider 0.99

freqTerms= findFreqTerms(dtm,lowfreq=100)#frequent words

plot(dtm,freqTerms,corThreshold=0.5)# plot graph

# Correlation plot



# Plot Word Frequency

```
term.freq<-
  sort(colSums(as.matrix(dtm)),decreasing=TRUE)
head(term.freq,14)
```

server replication publisher data database distributor

329 188 131 129 119 114

distribution will agent distributed servers transaction

111 91 90 64 61 60

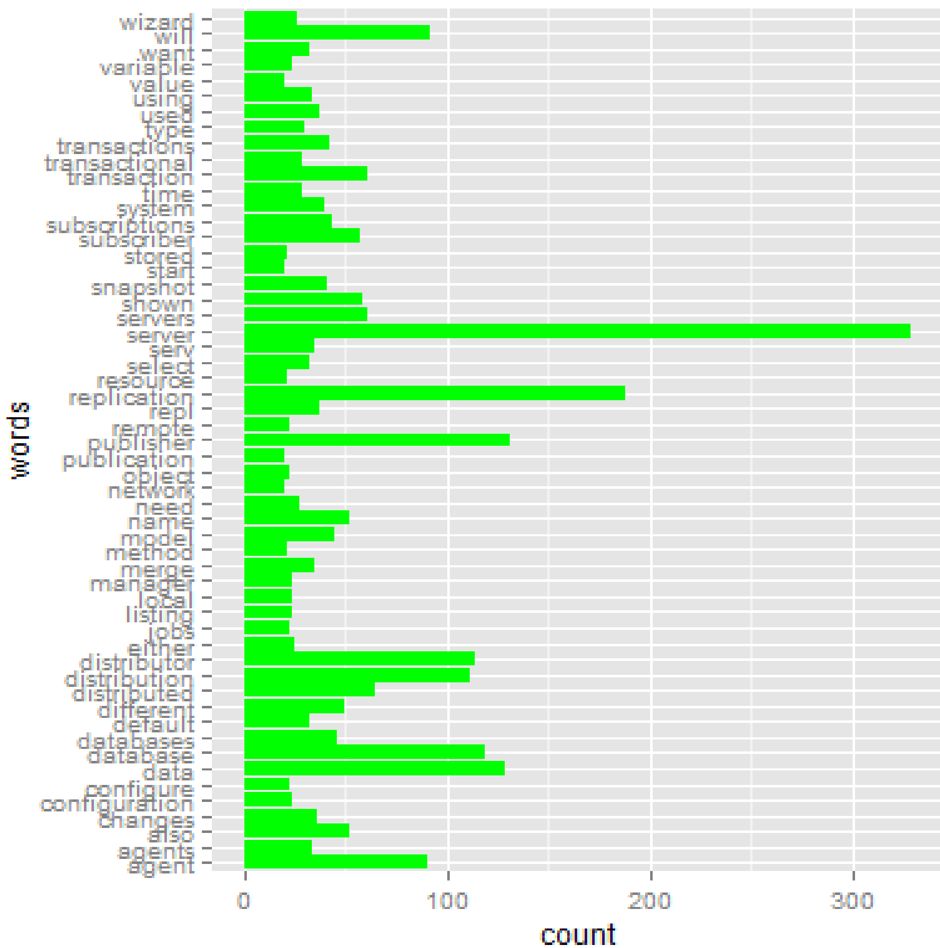
shown subscriber

58 57

# Plot Word Frequency

```
term.freq<-subset(term.freq,term.freq>=50)
word.freq<-data.frame(word=names(term.freq),freq=term.freq)
#plot the word frequency
ggplot(word.freq,aes(x=word,y=freq)) +
  geom_bar(stat="identity",fill="green")+
 xlab("words")+ylab("count")+coord_flip()
```

# Word Frequency Graph



2ND ANNUAL BIG DATA & ANALYTICS SUMMIT CANADA
Optimize your business value NOW!

# References


1. Wikipedia, https://en.wikipedia.org/wiki/Text_mining
2. Practical Text Mining and Statistical Analysis for Unstructured Text Data Applications, Gary Miner , John Elder, Andrew Fast, Thomas Hill, Academic Press, Jan 2012
3. http://siliconangle.com/blog/2012/05/21/when-will-the-world-reach-8-zetabytes-of-stored-data-infographic
4. http://blogs.gartner.com/darin-stewart/2013/05/01/big-content-the-unstructured-side-of-big-data
5. R Language Definition, https://cran.r-project.org/doc/manuals/r-release/R-lang.html

# References

6. Text Mining Infrastructure in R, Journal of statistical software, March 2008, vol25, Issue 5. Feinerer,I., Hornik,K., Meyer,D
7. Application of Text Mining in the Biomedical Domain, Methods, vol 74, 1 March 2015, pgs 97-106 http://www.sciencedirect.com/science/article/pii/S1046202315000274