Name: Lee Zhen Xuan

Student ID: 31860532

Unit Code: FIT3152

Q1: Collect a set of (machine-readable text) documents from an area of interest.

The area of interest collected are reviews of instruments. The documents contain different types of instruments for example acoustic guitar, electric guitar, bass guitar, acoustic piano, and keyboards. The documents are listed in a csv file, where each row is a document. The reviews collected are mostly from the website https://www.rogerebert.com/reviews to maintain the consistency of the writing of the documents. There are a total of 20 documents collected.

Q2: Creating corpus.

The corpus is built from the text data file "Instruments.csv". The reviews from the website are copied and pasted into rows of the csv file "Instruments.csv", and therefore each of the rows contain individual documents. The names of the instruments are used as identifiers in each of the rows of the "Instruments.csv" file, however only the first 20 letters are selected as identifiers for easier identification of each of the documents to plot the clustering nicely.

The added frisson in the case of the aforementioned 000 14-fretters was t	martin-000-28-modern-deluxe
Its search for sustainable alternatives to the old favourites have given us U	taylor-514ce-urban-ironbark
At first there's surprise, maybe a little horror, too. The shock of the new is	fender-acoustasonic-player-telecaster
The Player Series takes Fender's classic electric guitar designs, offering a g	fender-player-stratocaster
While Fender splits its guitars into series and periodically updates or refres	roland-fp-10-digital-piano-review
Roland's FP line of portable digital pianos features a range of models to su	gibson-les-paul-classic-2019
The company's most affordable entry-level upright, Yamaha's b1 is one of	yamaha-b1-acoustic-piano-review
Yamaha's Arius range of stylish, budget-friendly digital home console-style	yamaha-arius-ydp-s55-review
Offering an all-solid wood build of sexy Sitka spruce on top and okoume o	cort-gold-oc6
Taylor Academy's proposition is proper guitars for - in Taylor terms - not a	taylor-academy-12e-n
We have seen what the Fender Player Plus series has done for the electric	fender-player-plus-jazz-bass
Fender's American Ultra Series marks a new era for the Californian guitar	gibson-les-paul-junior-tribute-doublecut-bass-review
This year is Vox's 60th birthday, a rare event for any manufacturer and on	vox-ac30s1-combo
Fender has revealed the latest addition to its Tone Master range of guitar	fender-tone-master-princeton-chorus-amp
Casio's iconic Casiotone brand of portable electronic keyboards first brou	casio-casiotone-lk-s250-keyboard
Perched on the northwestern shoulder of North America in Bend, Oregon,	breedlove-eco-pursuit-ex-s-concert-sweetgrass
As befits one of the biggest-selling artists of recent times, Ed Sheeran does	sheeran-by-lowden-equals-edition
Weighing in at 12.8kg (28lbs 4oz), the Roland RD64 digital piano is surprisir	roland-rd-64-digital-piano-575154
Signature bass guitars tend to attract a little more attention. It's only natu	manson-john-paul-jones-signature-e-bass-and-manson-standard-e-bass
For any budding young shredder looking for their first guitar the CL-22M is	black-knight-cl22m-electric-guitar-567540

The first column contains the reviews, and the second column contains the instrument name, which is used as the identifier.

Q3: Creating Document-Term Matrix (DTM)

Some basic pre-processing has been done which are:

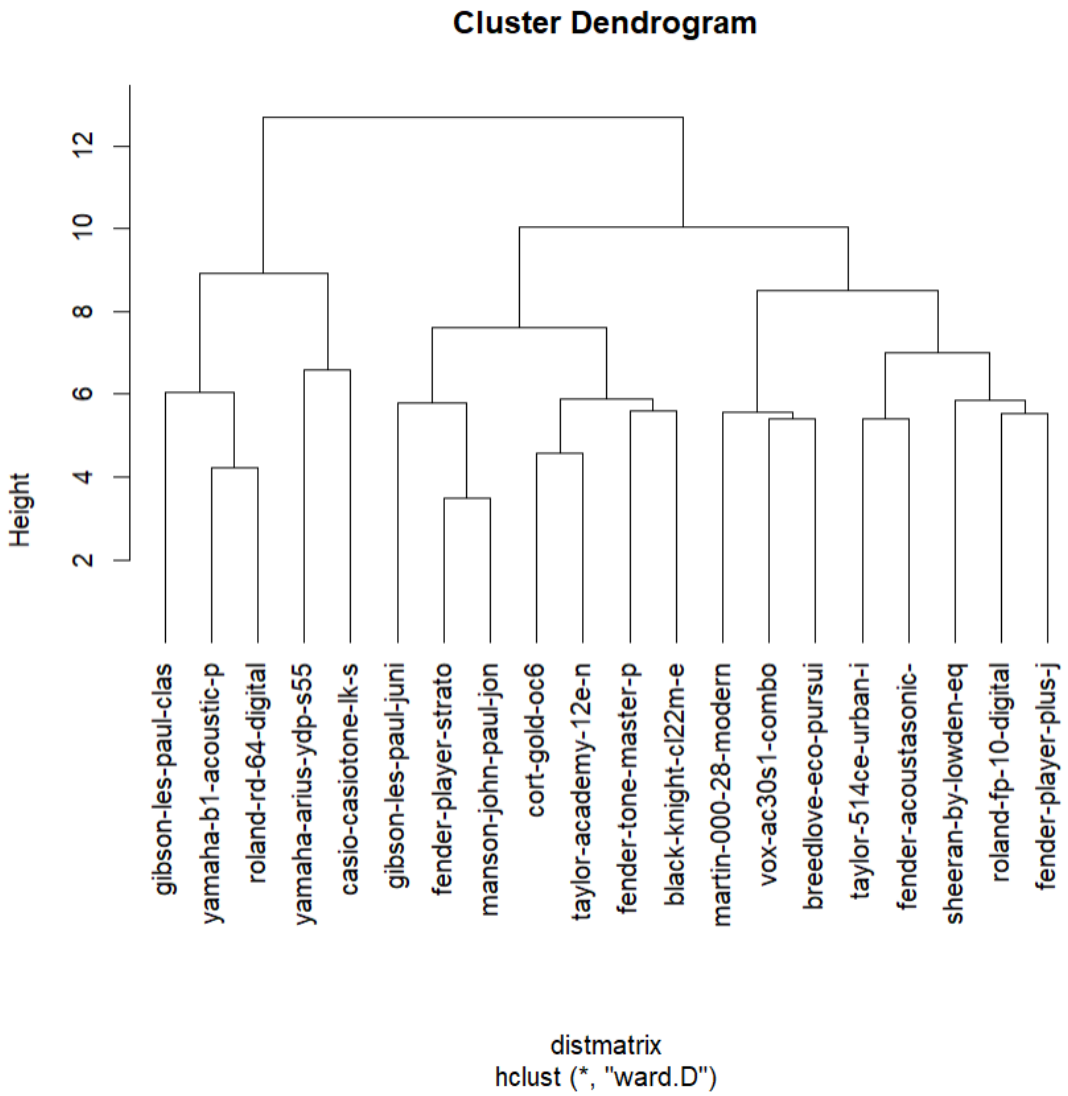
- Removing numbers as they are considered noise in text analysis.
- Removing punctuations as they are usually not relevant in text analysis.
- Converting the documents to lowercase to standardize the documents.
- Removing stop words as they do not carry much meaning and will affect the highest frequency tokens.

- Stemming as English language to reduce the words to their base/root form, where similar words are considered the same.

The sparse terms has also been removed from distance matrix. Before removing the sparse terms, there are a total of 2639 terms. After removing the terms, there are 22 terms.

Q4: Creating hierarchical clustering for corpus.

One of the processes of creating of the hierarchical clustering is by calculating the distance matrix using the cosine distance measure. The Hierarchical clustering is performed by minimizing the total within-cluster variance. The dendrogram is cut into 20 clusters and assigned a cluster label to each of the documents.



The quantitative measure of the quality of the clustering is by first creating the list of topics/types of instruments that corresponds to the corpus, plotting the cluster table then calculating the accuracy.

	Clusters						
GroupNames	1	2	3	4	5	6	7
ag	2	1	0	1	0	0	1
amp	1	0	0	0	0	0	1
ap	0	0	0	0	1	0	0
bass	0	0	2	1	0	0	1
cg	0	0	0	0	0	0	1
dp	0	0	0	1	1	2	0
eg	0	1	1	0	1	0	0

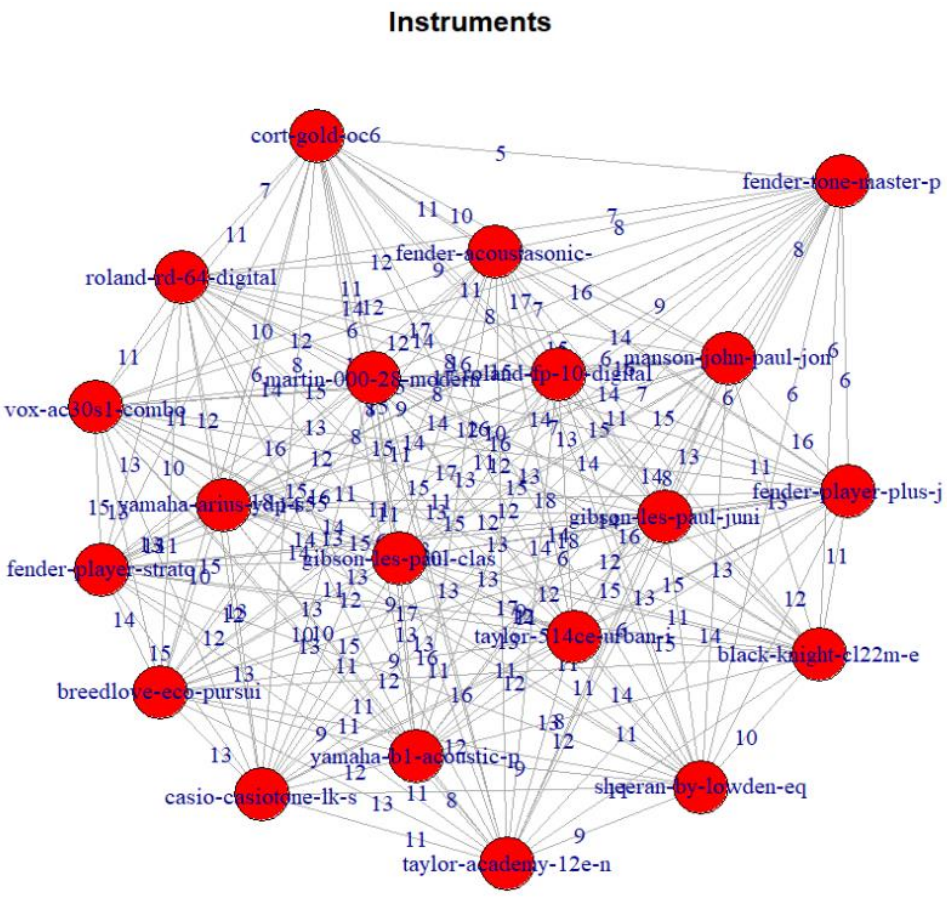
The cluster table only reported an accuracy of 0.25, which means that only 25% of the assignments in the cluster table matches the true labels.

Q5: Single-mode network showing the connections between the documents

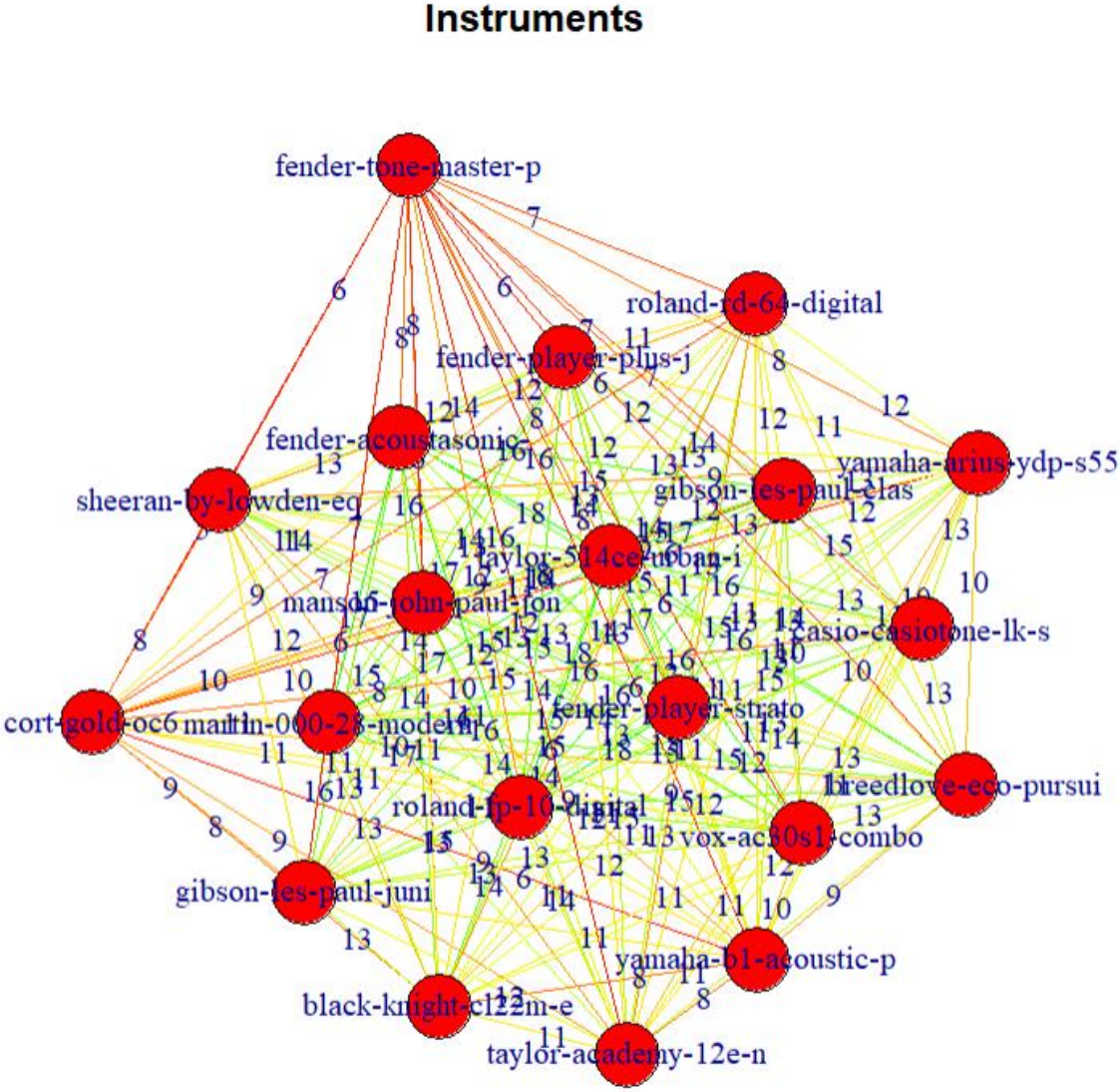
The steps of creating the single-mode network are shown as below:

1. Converting the distance matrix to binary matrix.
2. Multiply transpose binary matrix by binary matrix.
3. Make leading diagonal zero.
4. Strength of documents are created.
5. Plotting the graph with edge weights

The single-mode network plot can be shown below:



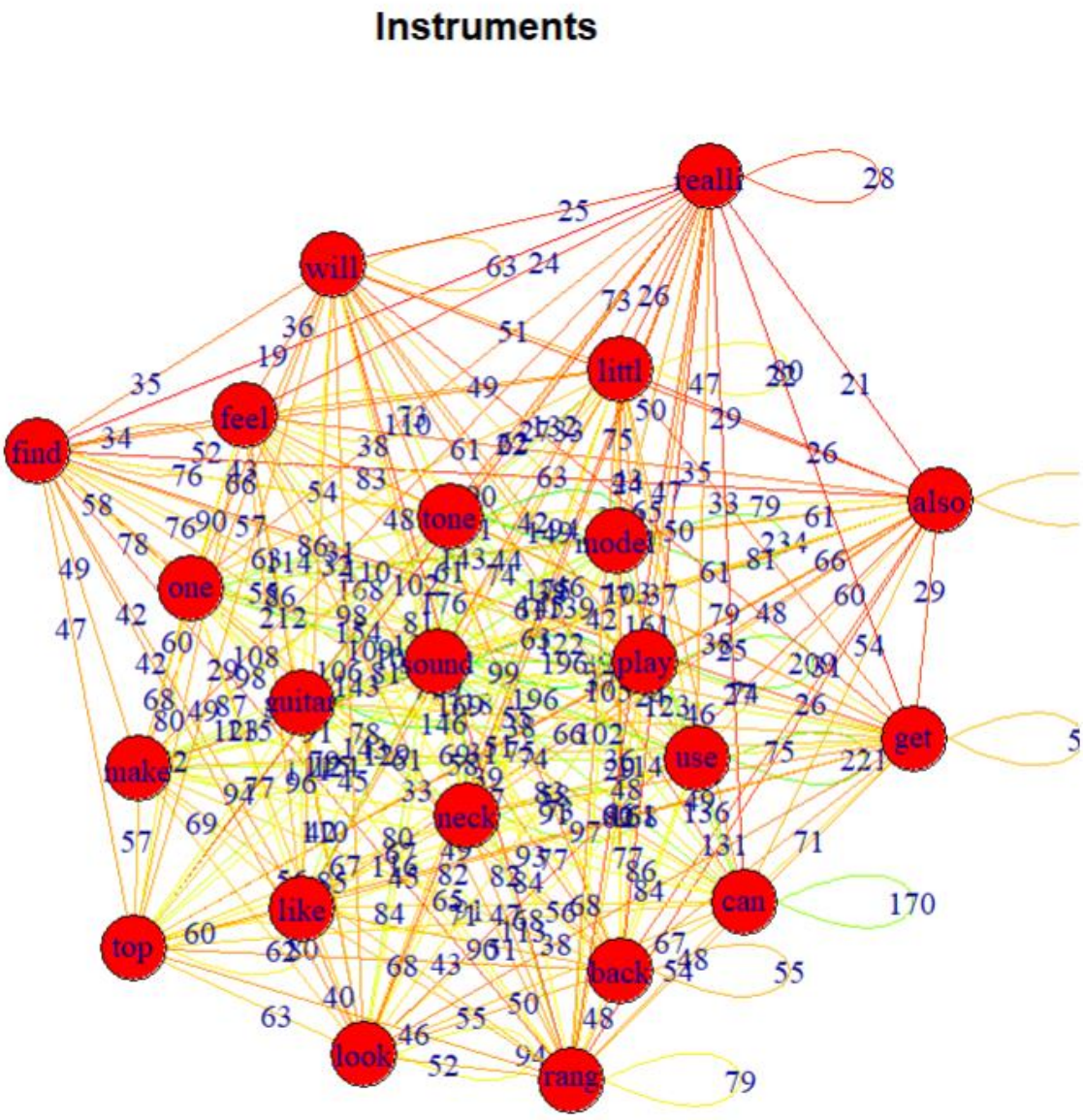
It is hard to identify the important relationships between the documents with the plot above. Therefore, an improvement to the visualisation is by assigning colours to the edges based on their weights.



As we can see from the plot above, the stronger weights between the nodes are coloured in green, followed by yellow, and the weakest weights are coloured red. By assigning colours to the edges, we can clearly see that there is an obvious group in the middle of the network, but most importantly we can also identify the outliers which are located on the outside of the network where the edges are coloured red. For example, the “fender-tone-master-p” node which lies at the far top of the network has weak weighted edges and is located further away from the network which indicates that the node does not have a strong relationship with the other documents. On the other hand, most of the edges in the middle of the network are mostly green, which indicates that these documents are closely related to each other in terms of commonly used words for instruments reviews.

Q6: Single-mode network showing the connections between the words(tokens)

Based on the single-mode network for the documents above, we can perform the same analysis for the words. The network plot is shown below:



Based on the network plot, we can also identify the nodes which are important by looking at the colour of the edges. Overall, by manually looking at the meaning of the nodes, the network does quite well in predicting the important words used in the review. For example, commonly used words for instruments such as “sound”, “tone”, “model”, “play” are the important nodes which lies in the middle of the network and some high weighted edges.

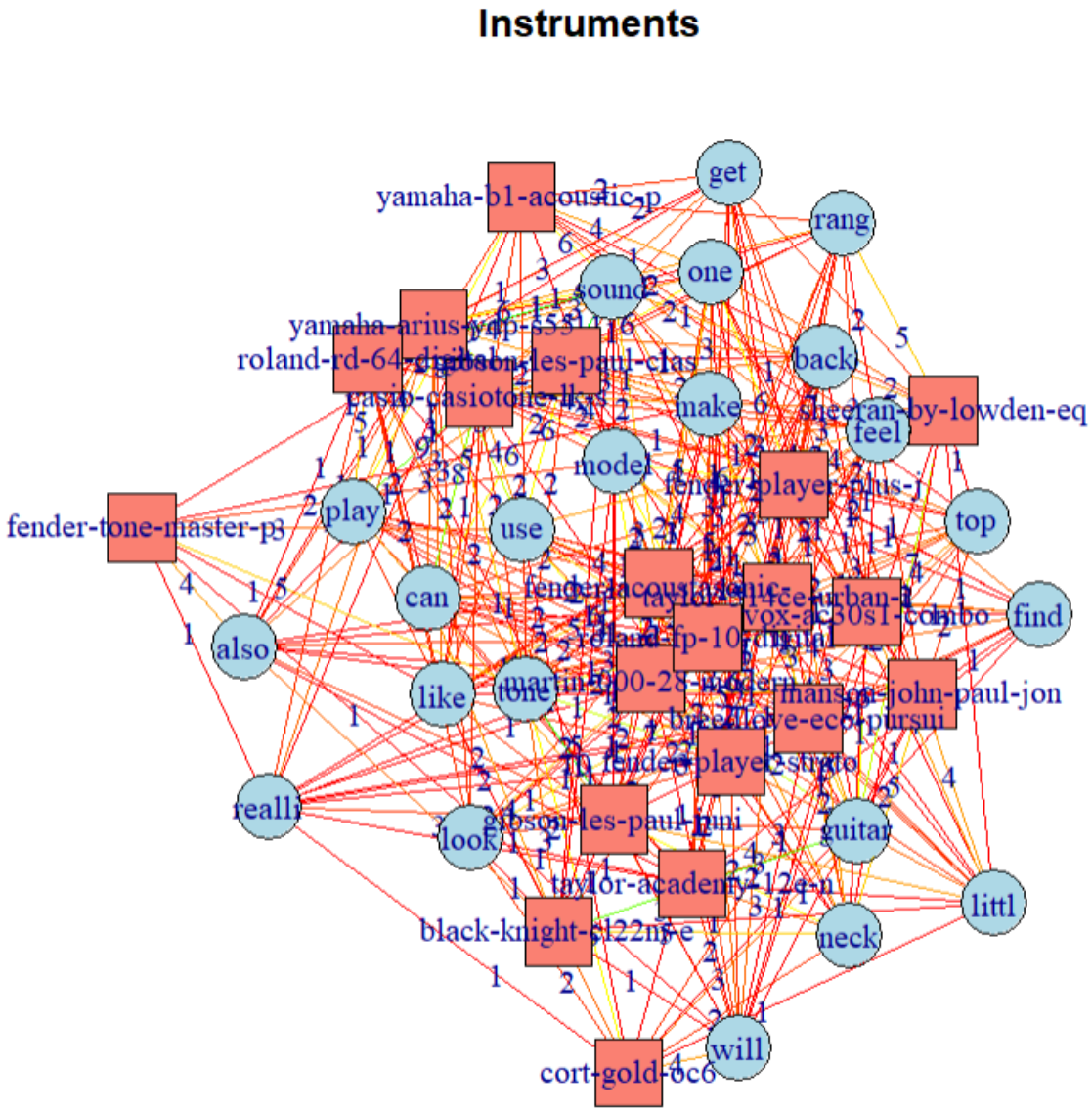
Q7: Bipartite (two-mode) network for document and tokens

A bipartite network can be plotted, with one type of nodes as the document names, and the other type which are the nodes to identify their relationships between the words and documents as well.

First, a table is plotted with the document names, tokens and their respective weights. The table is shown below:

	abs	token	weight
1	martin-000-28-modern	also	1
2	martin-000-28-modern	back	2
3	martin-000-28-modern	can	2
4	martin-000-28-modern	find	3
5	martin-000-28-modern	guitar	7
6	martin-000-28-modern	littl	2
7	martin-000-28-modern	look	5
8	martin-000-28-modern	make	1
9	martin-000-28-modern	model	6
10	martin-000-28-modern	neck	1
11	martin-000-28-modern	one	3
12	martin-000-28-modern	play	2
13	martin-000-28-modern	rang	3
14	martin-000-28-modern	realli	1
15	martin-000-28-modern	sound	2
16	martin-000-28-modern	tone	2
17	martin-000-28-modern	top	2
18	martin-000-28-modern	use	6
19	martin-000-28-modern	will	1
23	taylor-514ce-urban-i	also	1
24	taylor-514ce-urban-i	back	2
25	taylor-514ce-urban-i	can	4
26	taylor-514ce-urban-i	find	3
27	taylor-514ce-urban-i	guitar	4

By using this table, we can then assign different colours and shape to the types of nodes in the bipartite network:



As we can see from the plot above, the red squared nodes are the documents, where the blue circles are the words. The weights of the edges as well as the colours assignments are also assigned to the edges. The weights of the edges tell us how strongly they are related to each other.

Based on the graph above, we can see that there are clear groups of the documents that form a cluster. For example, on the top left of the graph, we can see that rolan-64, Yamaha-arius, casio-casiotone and Gibson-les-paul-class form a cluster (we call this “group” 1) as they are closely placed to each other, and Yamaha-b1-acoustic-p is also close to the cluster as well, while on the other hand there appears to be another cluster around the middle of the network which contains the documents fender-cousticsonic, roland-fp-10, martin-000-28,taylo-514ce and more (we call this “group 2). By looking at the graph manually and knowing the type of instruments they belong to, the graph does quite well in forming clusters of the instruments in the same group and the related words as well. For example, “group” 1 mostly consists of pianos, and “group” 2 mostly consists of guitars. The words that are closely related to the 2 groups are commonly used words for all instruments such as “play”, “use”, “model”, “make” and more. While on the other hand, the words that are only closely

related to “group” 2 are commonly used words for guitars only such as “guitar” “neck”, “top” and more. The graph is also improved by adding the weights as the labels and the colours to indicate the strength of the weights.

In conclusion, text analysis can be used to analyse the relationships between individual text based documents to find important similarities and differences between the documents overall, and the individual words, and how closely related they are to each other.

Appendix

Websites for the reviews:

https://www.musicradar.com/reviews/martin-000-28-modern-deluxe

https://www.musicradar.com/reviews/taylor-514ce-urban-ironbark

https://www.musicradar.com/reviews/fender-acoustasonic-player-telecaster

https://www.musicradar.com/reviews/fender-player-stratocaster

https://www.musicradar.com/reviews/roland-fp-10-digital-piano-review

https://www.musicradar.com/reviews/gibson-les-paul-classic-2019

https://www.musicradar.com/reviews/yamaha-b1-acoustic-piano-review

https://www.musicradar.com/reviews/yamaha-arius-ydp-s55-review

https://www.musicradar.com/reviews/cort-gold-oc6

https://www.musicradar.com/reviews/taylor-academy-12e-n

https://www.musicradar.com/reviews/fender-player-plus-jazz-bass

https://www.musicradar.com/reviews/gibson-les-paul-junior-tribute-doublecut-bass-review

https://www.musicradar.com/reviews/fender-american-ultra-precision-and-jazz-bass

https://www.musicradar.com/reviews/vox-ac30s1-combo

https://www.musicradar.com/news/fender-tone-master-princeton-chorus-amp

https://www.musicradar.com/reviews/casio-casiotone-lk-s250-keyboard

https://www.musicradar.com/reviews/tech/roland-rd-64-digital-piano-575154

https://www.musicradar.com/reviews/breedlove-eco-pursuit-ex-s-concert-sweetgrass

https://www.musicradar.com/reviews/manson-john-paul-jones-signature-e-bass-and-manson-standard-e-bass

https://www.musicradar.com/reviews/guitars/black-knight-cl22m-electric-guitar-567540

https://www.musicradar.com/reviews/sheeran-by-lowden-equals

R code:

```
setwd("C:/Monash/FIT3152/Assignment3/asgn3")

rm(list = ls())

library(slam)

library(tm)

library(SnowballC)

library(igraph)

library(ggplot2)

library(slam)



set.seed(31860532)

# Q2

# Build corpus from the text data file

inst <- read.csv("Instruments.csv", header = FALSE)

# Extract first 10 letters from the second column

doc_id <- substr(as.character(inst[, 2]), 1, 20)

doc_id

inst <- data.frame(doc_id = doc_id, text = inst[1])


colnames(inst) <- c('doc_id', 'text')


docs <- Corpus(DataframeSource(inst))


print(summary(docs))


# Tokenize

# Hyphen to space, ref Williams

toSpace <- content_transformer(function(x, pattern) gsub(pattern, " ", x))
```

```
docs <- tm_map(docs, toSpace, "–")
docs <- tm_map(docs, toSpace, "’s")
docs <- tm_map(docs, removeNumbers)
docs <- tm_map(docs, removePunctuation)
docs <- tm_map(docs, content_transformer(tolower))

# Filter Words
# Remove stop words and white space
docs <- tm_map(docs, removeWords, stopwords("english"))
docs <- tm_map(docs, stripWhitespace)
# Stem
docs <- tm_map(docs, stemDocument, language = "english")

# Q3
# Create document term matrix
dtm <- DocumentTermMatrix(docs)
dtm

tokens <- colnames(dtm)
num_tokens <- length(tokens)
num_tokens

# Check word frequencies, ref Williams
freq <- colSums(as.matrix(dtm))
length(freq)
ord <- order(freq)
freq[head(ord)]
freq[tail(ord, 20)]
freq
```

```
#Removing sparse terms
dtm <- removeSparseTerms(dtm, sparse = 0.35)  # Adjust the 'sparse' parameter as needed (e.g., 0.95 means remove terms that occur in more than 95% of documents)



# Check word frequencies, ref Williams
freq <- colSums(as.matrix(dtm))
length(freq)
ord <- order(freq)
freq[head(ord)]
freq[tail(ord, 20)]
freq

#modified matrix
dtm



# Frequency of frequencies, ref Williams
head(table(freq), 10)
tail(table(freq), 10)
dim(dtm)
dtms <- removeSparseTerms(dtm, 0.9)
dim(dtms)

# inspect(dtms)
findFreqTerms(dtm, lowfreq = 10)
dtms <- as.matrix(dtms)
dtms
write.csv(dtms, "Inst.csv")
```

```
# Q4

# Cluster


distmatrix <- dist(scale(dtms))

distmatrix

fit <- hclust(distmatrix, method = "ward.D")

cutfit <- cutree(fit, k = 7)

plot(fit)

plot(fit, hang = -1)

cutfit



topics = c("ag", "ag", "eg", "eg", "dp", "eg", "ap", "dp","ag", "cg", "bass", "bass", "amp", "amp", "dp",
"ag","ag", "dp", "bass", "bass")


groups = cutree(fit, k=7)

cluster_table <- table(GroupNames = topics, Clusters = groups)

cluster_table



# Calculate the accuracy

accuracy <- sum(diag(cluster_table)) / sum(cluster_table)


accuracy



#install.packages("cluster")

library(cluster)



# Check word frequencies, ref Williams
```

```
freq <- as.data.frame(as.table(freq))
nfreq <- freq[order(-freq$Freq),]
nfreq <- nfreq[1:100,]
# Plot column graph of frequent words
nfreq$Var1 <- factor(nfreq$Var1, levels = nfreq$Var1[order(-nfreq$Freq)])
ggplot(data = nfreq, aes(x = Var1, y = Freq)) + geom_bar(stat = "identity") + theme_minimal()

print(nfreq$Var1)



#Q5
#convert to binary matrix
dtmsx <- as.matrix((dtms > 0) + 0)
#multiply transpose binary matrix by binary matrix
ByAbsMatrix <- dtmsx %*% t(dtmsx)
#make leading diagonal zero
diag(ByAbsMatrix) <- 0
ByAbsMatrix
g1 <- graph_from_adjacency_matrix(ByAbsMatrix, mode = "undirected", weighted = TRUE)

# Get the edge weights
edge_weights <- E(g1)$weight

# Plot the graph with edge weights
plot(g1, vertex.color = "red", main = "Instruments", edge.label = edge_weights)

# Define the color palette for the edges
color_palette <- colorRampPalette(c("red","yellow", "green"))

# Map the edge weights to the color palette
```

```
edge_colors <- color_palette(length(edge_weights))[as.numeric(cut(edge_weights, breaks =
length(edge_weights)))]

# Plot the graph with adjusted edge color

plot(g1, vertex.color = "red", main = "Instruments", edge.color = edge_colors, edge.width = 1,
edge.label = edge_weights)



set.seed(31860532)

# Calculate the co-occurrence matrix

token_cooc <- t(dtms) %*% dtms

# Convert co-occurrence matrix to a graph

g2 <- graph_from_adjacency_matrix(token_cooc, mode = "undirected", weighted = TRUE)

edge_weights2 <- E(g2)$weight

# Calculate the logarithm of edge weights to handle outliers

log_weights <- log(edge_weights2)

# Define the color palette for the edges based on the log weights

color_palette <- colorRampPalette(c("red","yellow","green"))(length(log_weights))

# Normalize the log weights to [0,1]

normalized_weights <- (log_weights - min(log_weights)) / (max(log_weights) - min(log_weights))

# Map the normalized log weights to colors from the color palette

edge_colors <- color_palette[as.integer(cut(normalized_weights, breaks = length(log_weights)))]

# Plot the graph with adjusted edge color

plot(g2, vertex.color = "red", main = "Instruments", edge.color = edge_colors, edge.width = 1,
edge.label = edge_weights)
```

```
# Q7
# Create bipartite graph
dtmsa <- as.data.frame(dtms)  # clone dtms
dtmsa$ABS <- rownames(dtmsa)  # add row names
dtmsb <- data.frame()

for (i in 1:nrow(dtmsa)) {
  for (j in 1:(ncol(dtmsa) - 1)) {
    touse <- cbind(dtmsa[i, j], dtmsa[i, ncol(dtmsa)], colnames(dtmsa)[j])
    dtmsb <- rbind(dtmsb, touse)
  }
}

colnames(dtmsb) <- c("weight", "abs", "token")
dtmsc <- dtmsb[dtmsb$weight != 0,]  # delete 0 weights
dtmsc = dtmsc[,c(2,3,1)]
dtmsc

# Create graph object and declare bipartite
g <- graph.data.frame(dtmsc, directed = FALSE)
bipartite.mapping(g)
bipartite.mapping(g)$type
V(g)$type <- bipartite_mapping(g)$type
V(g)$color <- ifelse(V(g)$type, "lightblue", "salmon")
V(g)$shape <- ifelse(V(g)$type, "circle", "square")
```

```
E(g)$color <- "lightgray"

plot(g, layout = layout_with_fr(g), edge.curved = FALSE, vertex.label.dist = 1.5, vertex.label.cex = 0.8, margin = 1, asp = 0)


edge_weights3 <- as.numeric(E(g)$weight)



# Define the color palette for the edges

color_palette <- colorRampPalette(c("red","yellow", "green"))


# Map the edge weights to the color palette

edge_colors <- color_palette(length(edge_weights3))[as.numeric(cut(edge_weights3, breaks = length(edge_weights3)))]


# Plot the graph with adjusted edge color

plot(g, main = "Instruments", edge.color = edge_colors, edge.width = 1, edge.label = edge_weights3)
```