FIT3152 Data Analytics
Semester 1 2023
Assignment 3

Name: Tatiana Sutulova

Student ID: 30806151

1.1 Data collection

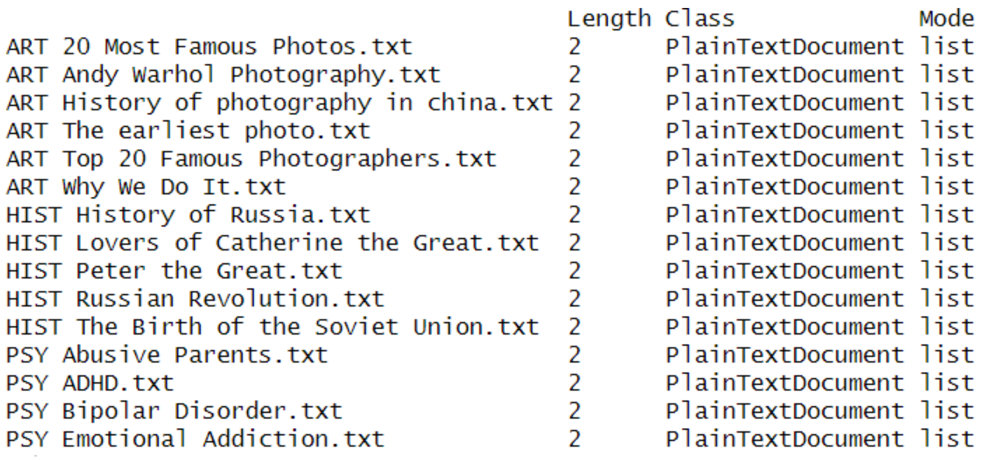
Table 1 provides a comprehensive compilation of the documents gathered for subsequent processing and analysis. In total, there are 15 documents encompassing three distinct areas of interest: Photography (6 documents), the History of Russia (5 documents), and Psychology (4 documents).

Photography	https://time.com/4839246/photographers-passion/ https://webneel.com/famous-photographers https://www.invaluable.com/blog/20-most-famous-photographs/ https://www.theguardian.com/culture/2022/sep/30/andy-warhols-star-studded-photographs-you-find-out-much-more-about-him-as-a-person https://quod.lib.umich.edu/t/tap/7977573.0004.208/--history-of-photography-in-china-chinese-photographers-1844?rgn=main;view=fulltext https://www.hrc.utexas.edu/niepce-heliograph/#:~:text=It%20is%20the%20earliest%20photograph,the%20Burgundy%20region%20of%20France.
History of Russia	https://www.advantour.com/russia/history.htm https://www.britannica.com/biography/Nikolay-Ivanovich-Bukharin https://guides.loc.gov/peter-the-great https://www.historyhit.com/the-most-important-lovers-of-catherine-the-great/ https://daily.jstor.org/the-birth-of-the-soviet-union-and-the-death-of-the-russian-revolution/
Psychology	https://marksteinberg.com/webpages/writings/emotional-addiction.jsp https://www.psychiatry.org/patients-families/bipolar-disorders/what-are-bipolar-disorders https://www.additudemag.com/what-is-adhd-symptoms-causes-treatments/ https://www.parentingforbrain.com/mentally-abusive-parents/

Table 1. Documents dataset

1.2 Corpus creation

To compile the corpus, all the documents referenced in section 1.1 were transformed into text files, which was done by manually copying and pasting the data while also fixing any obvious text mistakes. Each file is named using the format "Area of interest code + file name," such as "PSY Emotional Addiction.txt" or “HIST History of Russia.txt”. This naming convention facilitates identification during tasks like clustering or network graph analysis. All the files are organized within the “. /CorpusAbstracts/txt" directories. Figure 1 displays the summary of the created corpus object.



```
                                           Length Class            Mode
ART 20 Most Famous Photos.txt              2      PlainTextDocument list
ART Andy Warhol Photography.txt            2      PlainTextDocument list
ART History of photography in china.txt 2      PlainTextDocument list
ART The earliest photo.txt                 2      PlainTextDocument list
ART Top 20 Famous Photographers.txt        2      PlainTextDocument list
ART Why We Do It.txt                       2      PlainTextDocument list
HIST History of Russia.txt                 2      PlainTextDocument list
HIST Lovers of Catherine the Great.txt     2      PlainTextDocument list
HIST Peter the Great.txt                   2      PlainTextDocument list
HIST Russian Revolution.txt                2      PlainTextDocument list
HIST The Birth of the Soviet Union.txt     2      PlainTextDocument list
PSY Abusive Parents.txt                    2      PlainTextDocument list
PSY ADHD.txt                               2      PlainTextDocument list
PSY Bipolar Disorder.txt                   2      PlainTextDocument list
PSY Emotional Addiction.txt                2      PlainTextDocument list
```

Figure 1. Summary of the Corpus object

1.3 Document-Term Matrix (DTM) creation

In order to construct the Document-Term Matrix (DTM), a series of processing steps were implemented to eliminate undesired terms and artifacts from the text documents. Initially, punctuation marks, numbers, and default English stop words were excluded from the text as they do not contribute to the subsequent analysis. Subsequently, the texts underwent normalization by converting all characters to lowercase and standardizing the spacing. Through a process of trial and error, an additional list of stop words was identified and incorporated (Table 2). These words were also eliminated from the text as they do not provide useful insights for further analysis. Furthermore, stemming was applied to reduce words to their base or root form, thereby reducing redundancy in the dataset. With these preprocessing steps completed, the Document-Term Matrix was generated. To refine the matrix further, terms that appeared in less than 65% of the documents were removed, ensuring that only sufficiently common terms were retained. After this refinement, the final DTM (refer to Appendix 1) comprised 25 terms that capture the most relevant information from the text documents.

Additional Stop Words	can, one, many, new, often, people, work, may, also, even, life, time, two, way, make, first, use, still, take, change, live, made, ing, remain, day, however, like, form, much, known, will, see, another, person, without, last, towards, around, well, took, make, just, sen

Table 2. Manually determined Stop Words

1.4 Hierarchical clustering

To gain deeper insights into clustering outcomes and enhance comprehension, I conducted two types of clustering analyses: conventional one employing Euclidean distance and the other one utilizing Cosine distance. The outcomes are visualized in Figure 2 and Figure 3, presenting dendrograms respectively. Upon examining the figures, it becomes evident that both clustering algorithms perform reasonably well. Nonetheless, the Cosine distance clustering approach exhibits superior results by clearly revealing three distinct thematic groups, each containing the relevant documents. In contrast, the Euclidean distance-based algorithm forms four separate clusters, with three of them successfully capturing groups of interest. However, the fourth cluster comprises a mixture of documents from all three thematic groups, thereby diminishing the algorithm's accuracy.

After analyzing dendrograms presented in Figure 2 and Figure 3, confusion matrices for both clustering methods are constructed and depicted in Figure 4. Subsequently, the accuracy of each algorithm is computed: Accuracy (Euc) = (4+3+3)/15 = 67% and Accuracy (Cos) = (6+4+5)/15 = 100%. These calculations provide quantitative evidence that the Cosine distance-based algorithm exhibits superior accuracy compared to the Euclidean-distance algorithm.

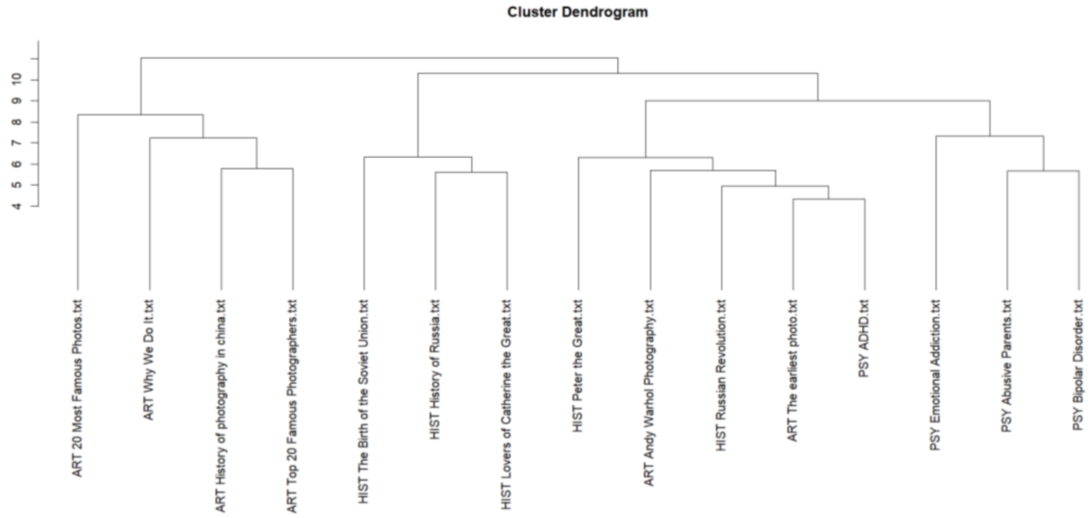


Figure 2. Euclidean distance clustering dendrogram

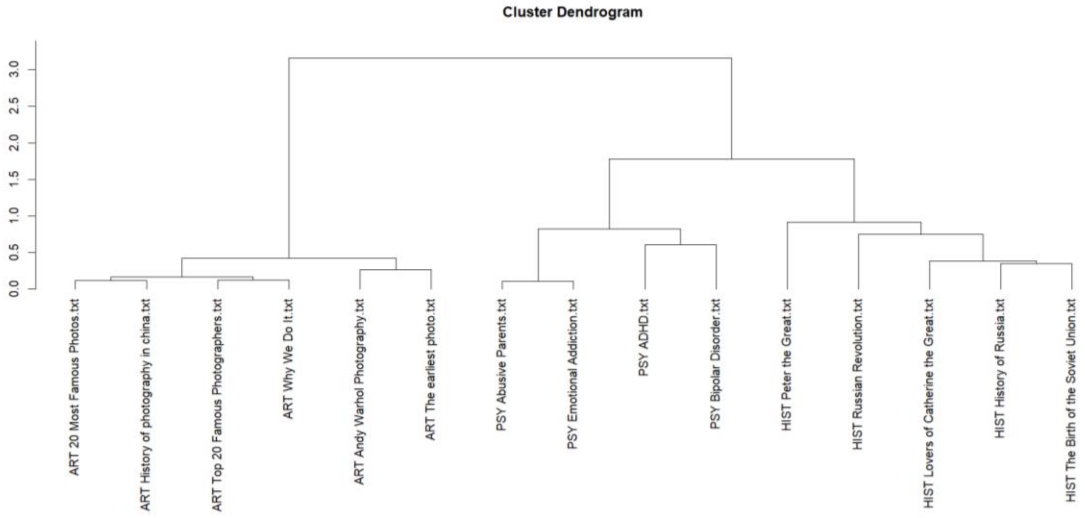


Figure 3. Cosine distance clustering dendrogram

	1	2	3	4
ART	4	0	2	0
HIST	0	3	2	0
PSY	0	0	1	3

Euclidean distance clustering

	1	2	3
ART	6	0	0
HIST	0	0	5
PSY	0	4	0

Cosine distance clustering

Figure 4. Confusion matrices

1.5 Documents single mode network

Figure 5 depicts the single-mode network, which visualizes the relationships between documents based on the shared terms they possess. Upon analyzing the graph, it becomes evident that there is no distinct clustering of data, and all thematic groups appear to be intertwined. Additionally, the graph highlights the significance of the document titled *"ART 20 Most Famous Photos.txt,"* as it occupies a central position within the network.

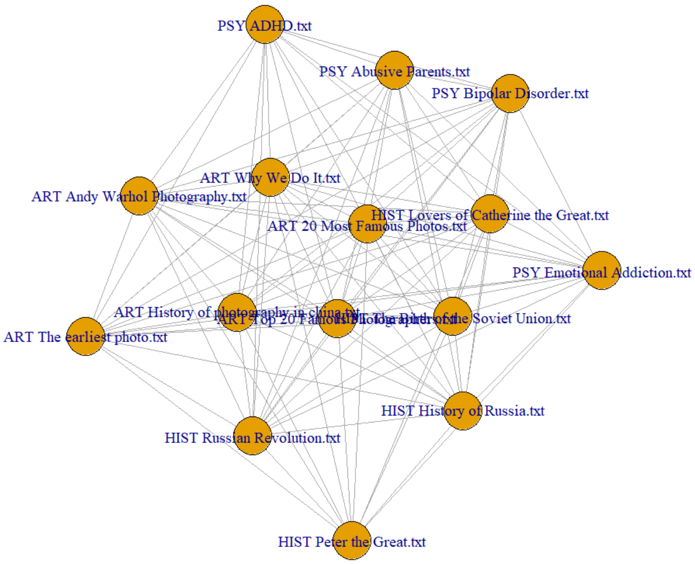
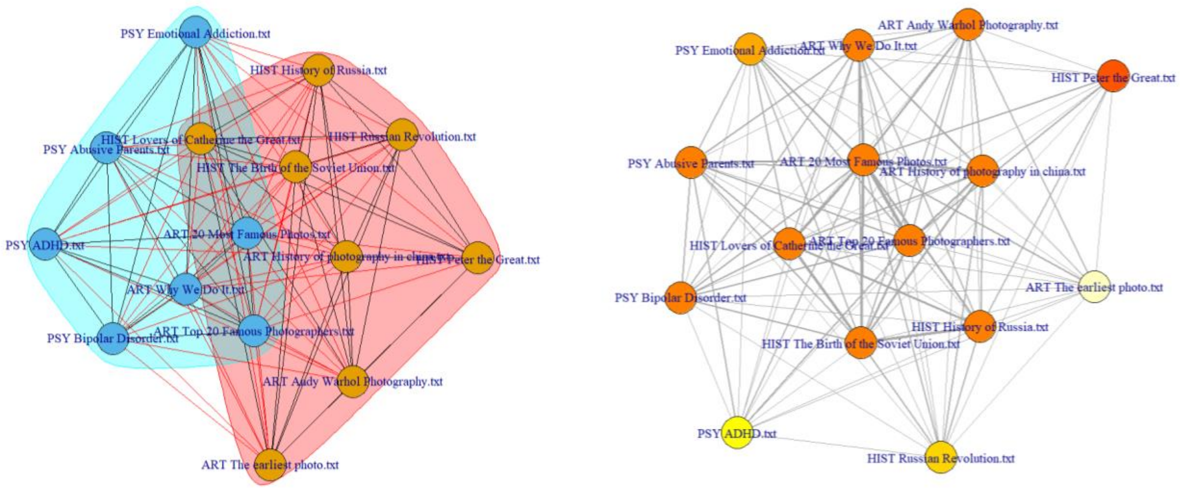

PSY ADHD.txt
PSY Abusive Parents.txt
PSY Bipolar Disorder.txt
ART Why We Do It.txt
ART Andy Warhol Photography.txt
HIST Lovers of Catherine the Great.txt
ART 20 Most Famous Photos.txt
PSY Emotional Addiction.txt
ART History of photography in china.txt
ART Top 20 Famous Photographers.txt
HIST The Birth of the Soviet Union.txt
ART The earliest photo.txt
HIST History of Russia.txt
HIST Russian Revolution.txt
HIST Peter the Great.txt

Figure 5. Single-mode network showing the connections between the documents.

Figure 6 showcases an enhanced version of the single-mode network, illustrating the connections between documents based on the shared terms. In Figure 6(a), the network's communities are depicted based on greedy modularity optimization. Two main communities are determined: one comprising HIST theme documents and the other consisting of PSY theme documents. While these communities exhibit some resemblance to the original three groupings, they still significantly differ from the original classifications. Figure 6(b) illustrates the strength of connections, represented by the width of the edges in the graph. Additionally, it portrays the relative importance of nodes, with lighter-colored nodes indicating higher significance. Node importance is determined by a combined centrality measure that incorporates degree, betweenness, closeness, and eigenvector values. According to the figure, the most vital nodes are "*ART The Earliest photo.txt*", followed by "*PSY ADHD.txt*", and then by "*HIST Russian Revolution.txt*" and "*PSY Emotional Addiction.txt*".


PSY Emotional Addiction.txt
HIST History of Russia.txt
HIST Lovers of Catherine the Great.txt
HIST Russian Revolution.txt
PSY Abusive Parents.txt
HIST The Birth of the Soviet Union.txt
ART 20 Most Famous Photos.txt
PSY ADHD.txt
ART History of photography in china.txt
HIST Peter the Great.txt
ART Why We Do It.txt
PSY Bipolar Disorder.txt
ART Top 20 Famous Photographers.txt
ART Andy Warhol Photography.txt
ART The earliest photo.txt
ART Andy Warhol Photography.txt
PSY Emotional Addiction.txt
ART Why We Do It.txt
HIST Peter the Great.txt
PSY Abusive Parents.txt
ART 20 Most Famous Photos.txt
ART History of photography in china.txt
HIST Lovers of Catherine the Great.txt
ART Top 20 Famous Photographers.txt
ART The earliest photo.txt
PSY Bipolar Disorder.txt
HIST History of Russia.txt
HIST The Birth of the Soviet Union.txt
PSY ADHD.txt
HIST Russian Revolution.txt

(a) (b)

Figure 6. Improved single-mode network showing the connections between the documents.

1.6 Tokens single mode network

Figure 7 illustrates the single-mode network, which provides a visual representation of the connections between tokens. Upon analyzing the graph, it becomes evident that certain tokens are grouped together based on their meaning. For instance, tokens such as "*influenc*," "*photogrpahi*," "*photograph*," "*public*," and "*imag*" are closely clustered, indicating their association with the topic of ART. However, the grouping is not entirely clear, and there are some centrally positioned terms that do not appear to belong to any specific group. These centrally placed terms, namely "*year*" and "*follow*," are considered the most significant ones in terms of their placement.

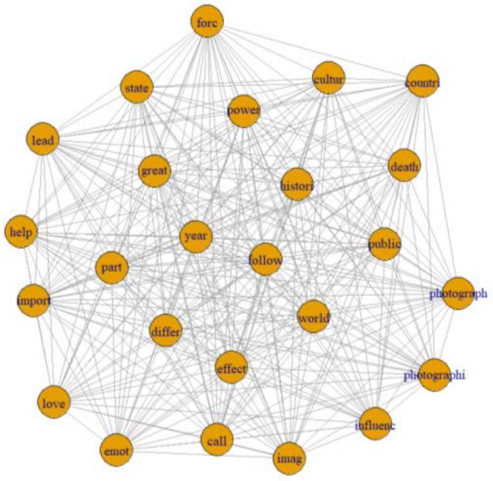
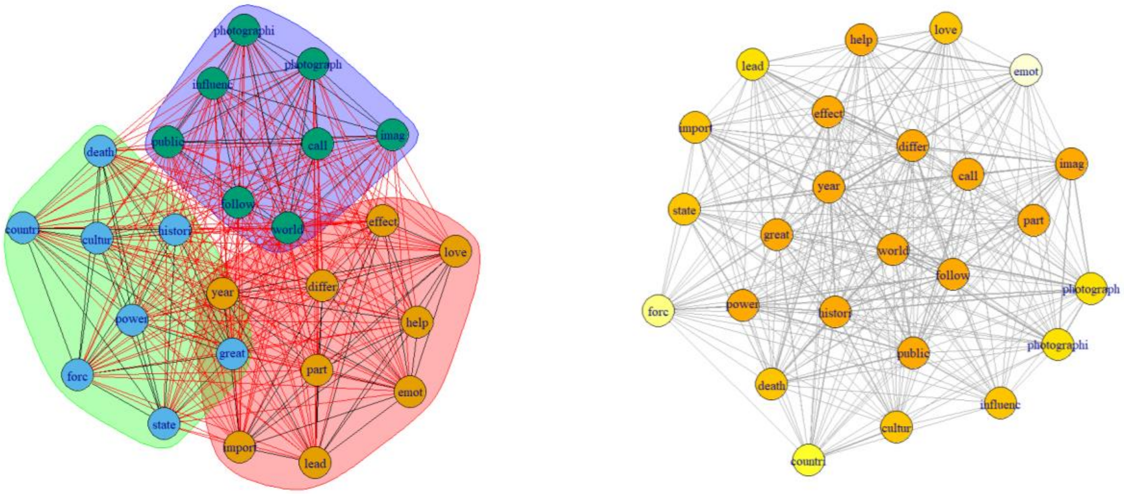


Figure 7. Single-mode network showing the connections between the tokens.

Figure 8 presents an improved version of the single-mode network, showcasing the relationships between tokens. In Figure 8(a), the network's communities are visualized through a greedy modularity optimization process. We identify three main communities and by examining the terms within each community, we can clearly discern three initial topics of interest: the purple community corresponds to ART, the green community to HIST, and the red community to PSY. Figure 8(b) demonstrates the strength of connections, represented by the width of the edges in the graph. Additionally, it illustrates the relative importance of nodes, with lighter-colored nodes indicating higher significance. Node importance is determined by a comprehensive centrality measure that incorporates degree, betweenness, closeness, and eigenvector values. According to the figure, the most crucial nodes are "*emot*", followed by "*forc*", and then "*lead*", "*countri*", "*photograph*", and "*photographic*".



(a) (b)

Figure 8. Improved single-mode network showing the connections between the tokens.

1.7 Bipartite (two-mode) network of the corpus

The data is transformed into an appropriate format, resulting in a new data frame that includes weights of edges between documents and tokens. Based on this data, a bipartite (two-mode) network of the corpus is constructed, where document names and tokens are represented as distinct types of nodes. The resulting network is visually depicted in Figure 9. Although the figure provides insights into the expected thematic groupings, it is evident that all the data remains closely interconnected. Consequently, the groups can only be differentiated by analyzing the meanings of the nodes, as visually they appear undistinguishable.

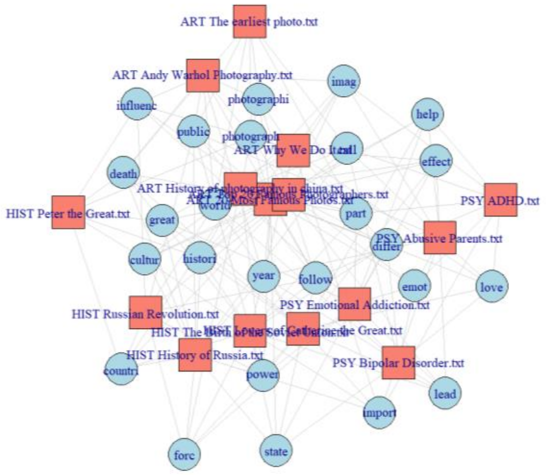
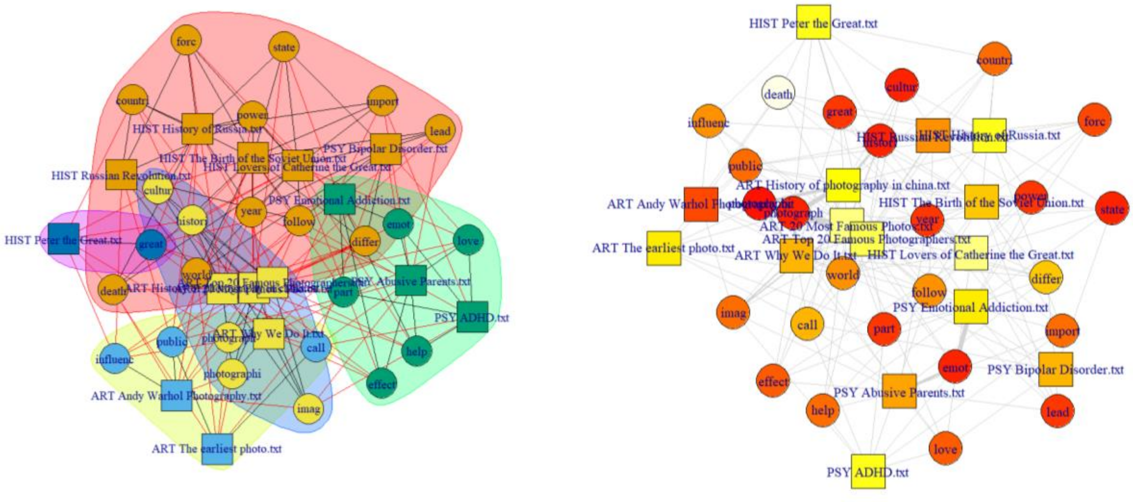

ART The earliest photo.txt
ART Andy Warhol Photography.txt
imag
influenc
photographi
help
public
photograph
ART Why We Do It.txt
call
effect
death
ART History of photography in china.txt
ART Top 20 Famous Photographers.txt
ART 20 Most Famous Photos.txt
world
PSY ADHD.txt
HIST Peter the Great.txt
great
part
PSY Abusive Parents.txt
differ
cultur
histori
year
follow
emot
love
PSY Emotional Addiction.txt
HIST Russian Revolution.txt
HIST The Birth of the Soviet Union.txt
HIST Lovers of Catherine the Great.txt
HIST History of Russia.txt
PSY Bipolar Disorder.txt
countri
power
lead
import
forc
state

Figure 9. Bipartite network of the corpus.

Figure 10 depicts an enhanced version of the two-mode network, which visualizes the associations between tokens and documents. In Figure 10(a), the network's communities are displayed using a modularity optimization process. Interestingly, the algorithm identifies more groups than anticipated. For instance, documents related to the ART category are divided into two separate communities instead of being grouped together. Similar to sections 1.5 and 1.6, Figure 10(b) showcases the relative significance of nodes with lighter colors and highlights stronger connections with thicker edges. Notably, the token "*death*" appears to be the most crucial, while among the documents, "*ART 20 Most Famous Photos.txt*" and "*HIST Lovers of Catherine the Great.txt*" stand out as the most important ones.


forc
state
countri
power
import
lead
HIST History of Russia.txt
HIST The Birth of the Soviet Union.txt
PSY Bipolar Disorder.txt
HIST Lovers of Catherine the Great.txt
HIST Russian Revolution.txt
cultur
PSY Emotional Addiction.txt
histori
year
follow
emot
love
HIST Peter the Great.txt
great
differ
world
death
PSY Abusive Parents.txt
part
PSY ADHD.txt
public
photograph
ART Why We Do It.txt
call
help
influenc
photographi
effect
ART Andy Warhol Photography.txt
imag
ART The earliest photo.txt
HIST Peter the Great.txt
countri
death
cultur
great
influenc
HIST Russian Revolution.txt
HIST History of Russia.txt
forc
public
ART History of photography in china.txt
ART Andy Warhol Photography.txt
photographi
photograph
HIST The Birth of the Soviet Union.txt
state
year
ART 20 Most Famous Photos.txt
ART Top 20 Famous Photographers.txt
ART The earliest photo.txt
ART Why We Do It.txt
HIST Lovers of Catherine the Great.txt
world
differ
follow
imag
call
PSY Emotional Addiction.txt
part
import
emot
PSY Bipolar Disorder.txt
effect
PSY Abusive Parents.txt
help
lead
love
PSY ADHD.txt

(a)

(b)

Figure 10. Improved bipartite network of the corpus.

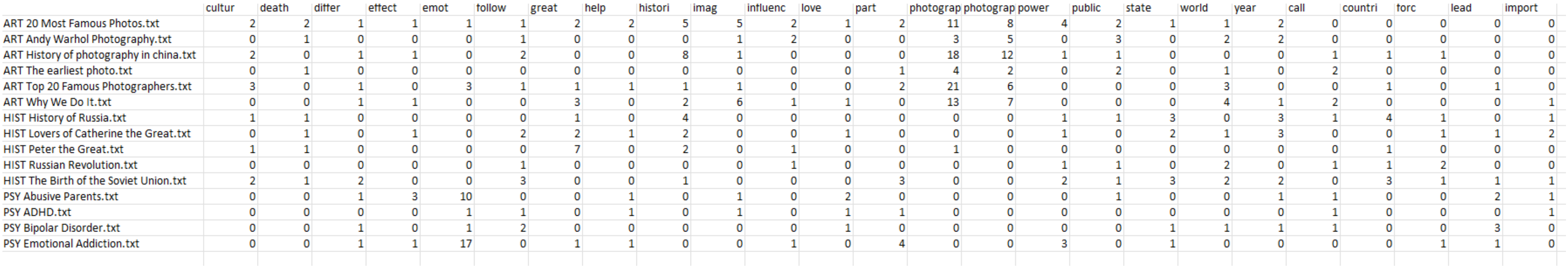
1.8 Summary

Analysing the findings presented in sections 1.1-1.4 reveals that the Cosine-distance based approach outperforms the Euclidean distance approach. The former demonstrates an accuracy of 100% in determining three groups of interest and appropriately classifying all the documents, while the latter achieves 67% accuracy only.

Examining the results from sections 1.5-1.7, it is determined that the single-mode network for tokens effectively distinguishes three distinct themes and accurately classifies the tokens within them. On the other hand, the single-mode network for documents only distinguishes two theme groups, and the bipartite network overcomplicates the clustering, resulting in five sub-groups.

Therefore, clustering proves to be a superior method for grouping documents based on different topics. However, Network Analysis provides additional valuable insights, such as identifying the most significant documents ("*ART The Earliest photo.txt*", "*PSY ADHD.txt*", "*HIST Russian Revolution.txt*" and "*PSY Emotional Addiction.txt*" as determined in 1.5) and determining the most significant tokens ("*death*" as determined in 1.7).

Appendix 1. Document-Term Matrix (DTM)



	cultur	death	differ	effect	emot	follow	great	help	histori	imag	influenc	love	part	photograp	photograp	power	public	state	world	year	call	countri	forc	lead	import
ART 20 Most Famous Photos.txt	2	2	1	1	1	1	2	2	5	5	2	1	2	11	8	4	2	1	1	2	0	0	0	0	0
ART Andy Warhol Photography.txt	0	1	0	0	0	1	0	0	0	1	2	0	0	3	5	0	3	0	2	2	0	0	0	0	0
ART History of photography in china.txt	2	0	1	1	0	2	0	0	8	1	0	0	0	18	12	1	1	0	0	0	1	1	1	0	0
ART The earliest photo.txt	0	1	0	0	0	0	0	0	0	0	0	0	1	4	2	0	2	0	1	0	2	0	0	0	0
ART Top 20 Famous Photographers.txt	3	0	1	0	3	1	1	1	1	1	0	0	2	21	6	0	0	0	3	0	0	1	0	1	0
ART Why We Do It.txt	0	0	1	1	0	0	3	0	2	6	1	1	0	13	7	0	0	0	4	1	2	0	0	0	1
HIST History of Russia.txt	1	1	0	0	0	0	1	0	4	0	0	0	0	0	0	1	1	3	0	3	1	4	1	0	1
HIST Lovers of Catherine the Great.txt	0	1	0	1	0	2	2	1	2	0	0	1	0	0	0	1	0	2	1	3	0	0	1	1	2
HIST Peter the Great.txt	1	1	0	0	0	0	7	0	2	0	1	0	0	1	0	0	0	0	0	0	0	1	0	0	0
HIST Russian Revolution.txt	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	1	0	2	0	1	1	2	0	0
HIST The Birth of the Soviet Union.txt	2	1	2	0	0	3	0	0	1	0	0	0	3	0	0	2	1	3	2	2	0	3	1	1	1
PSY Abusive Parents.txt	0	0	1	3	10	0	0	1	0	1	0	2	0	0	0	0	1	0	0	1	1	0	0	2	1
PSY ADHD.txt	0	0	0	0	1	1	0	1	0	1	0	1	1	0	0	0	0	0	0	0	1	0	0	0	1
PSY Bipolar Disorder.txt	0	0	1	0	1	2	0	0	0	0	0	1	0	0	0	0	0	1	1	1	1	0	0	3	0
PSY Emotional Addiction.txt	0	0	1	1	17	0	1	1	0	0	1	0	4	0	0	3	0	1	0	0	0	0	1	1	0

Appendix 2. R Script

```r
# Author: Sutulova Tatiana, 30806151
# Assignment 3
# The objective of this assignment is to create a corpus of documents and analyse
# the relationships between them, as well as the relationships between the important words used in these
documents.

rm(list = ls())

library(slam)
library(tm)
library(SnowballC)
library(igraph)

cname = file.path(".", "CorpusAbstracts", "txt")
docs = Corpus(DirSource(cname))
summary(docs)

# 1.3 Processing steps to create DTM
# Removing punctuation
docs <- tm_map(docs, removePunctuation)
toSpace <- content_transformer(function(x, pattern) gsub(pattern, " ", x))
docs <- tm_map(docs, toSpace, "‘")
docs <- tm_map(docs, toSpace, "’")
docs <- tm_map(docs, toSpace, '“')
docs <- tm_map(docs, toSpace, '”')
docs <- tm_map(docs, toSpace, "–")

# Removing numbers
docs <- tm_map(docs, removeNumbers)

# Converting everything to lower case
docs <- tm_map(docs, content_transformer(tolower))

# Removing stop words
docs <- tm_map(docs, removeWords, stopwords("english"))

# Normalizing spaces
docs <- tm_map(docs, stripWhitespace)

# Removing useless words
docs <- tm_map(docs, toSpace, "can")
docs <- tm_map(docs, toSpace, "one")
docs <- tm_map(docs, toSpace, "many")
docs <- tm_map(docs, toSpace, "new")
docs <- tm_map(docs, toSpace, "often")
docs <- tm_map(docs, toSpace, "people")
docs <- tm_map(docs, toSpace, "work")
docs <- tm_map(docs, toSpace, "may")
docs <- tm_map(docs, toSpace, "also")
docs <- tm_map(docs, toSpace, "even")
docs <- tm_map(docs, toSpace, "life")
docs <- tm_map(docs, toSpace, "time")
docs <- tm_map(docs, toSpace, "two")
docs <- tm_map(docs, toSpace, "way")
docs <- tm_map(docs, toSpace, "make")
docs <- tm_map(docs, toSpace, 'first')
docs <- tm_map(docs, toSpace, 'use')
docs <- tm_map(docs, toSpace, 'still')
docs <- tm_map(docs, toSpace, 'take')
docs <- tm_map(docs, toSpace, 'change')
docs <- tm_map(docs, toSpace, 'live')
docs <- tm_map(docs, toSpace, 'made')
docs <- tm_map(docs, toSpace, 'ing')
docs <- tm_map(docs, toSpace, 'remain')
docs <- tm_map(docs, toSpace, 'day')
docs <- tm_map(docs, toSpace, 'however')
docs <- tm_map(docs, toSpace, 'like')
docs <- tm_map(docs, toSpace, 'form')
docs <- tm_map(docs, toSpace, 'much')
docs <- tm_map(docs, toSpace, 'known')
docs <- tm_map(docs, toSpace, 'will')
docs <- tm_map(docs, toSpace, 'see')
docs <- tm_map(docs, toSpace, 'another')
docs <- tm_map(docs, toSpace, 'person')
docs <- tm_map(docs, toSpace, 'without')
docs <- tm_map(docs, toSpace, 'last')
docs <- tm_map(docs, toSpace, 'towards')
docs <- tm_map(docs, toSpace, 'around')
docs <- tm_map(docs, toSpace, 'well')
docs <- tm_map(docs, toSpace, 'took')
docs <- tm_map(docs, toSpace, 'make')
docs <- tm_map(docs, toSpace, 'just')
docs <- tm_map(docs, toSpace, 'sen')


# Stemming
docs <- tm_map(docs, stemDocument, language = "english")
docs <- tm_map(docs, toSpace, 'includ')

# Creating DTM
dtm <- DocumentTermMatrix(docs)

# Remove sparse terms
dtms <- removeSparseTerms(dtm, 0.65)
dim(dtms)
inspect(dtms)

write.csv(as.matrix(dtms), file = "dtms.csv", row.names = TRUE)

################################################################################
# 1.4 Hierarchical clustering
# Conventional clustering using Euclidean distance
distmatrix = dist(scale(as.matrix(dtms)))
fit = hclust(distmatrix, method = "ward.D")
plot(fit)
plot(fit, hang = -1)

# Clustering based on Cosine Distance.
# Converting DTM to a matrix
tf <- as.matrix(dtms)
# Calculating IDF values
idf <- log( ncol(tf) / ( 1 + rowSums(tf != 0) ) )
# Converting IDF values into a diagonal matrix
idf <- diag(idf)
# Calculating TF-IDF matrix
```

```
tf_idf <- crossprod(tf, idf)
# Assigning the row names of tf as the column names of tf_idf
colnames(tf_idf) <- rownames(tf)
# Calculating the cosine distance between each pair of documents in the tf_idf
cosine_dist = 1-crossprod(tf_idf) /(sqrt(colSums(tf_idf^2)%*%t(colSums(tf_idf^2))))
# Converting matrix into a distance object
cosine_dist <- as.dist(cosine_dist)
# Performing hierarchical clustering
cluster1 <- hclust(cosine_dist, method = "ward.D")
plot(cluster1)
plot(cluster1, hang = -1)

################################################################################
# 1.5 Documents single-mode network creation
# Starting with original document-term matrix
dtmsx = as.matrix(dtms)
# Convert to binary matrix
dtmsx = as.matrix((dtmsx>0)+0)
# Multiply binary matrix by its transpose
ByAbsMatrix = dtmsx%*%t(dtmsx)
#Make leading diagonal zero
diag(ByAbsMatrix) = 0
#create graph object (Basic plot)
ByAbs= graph_from_adjacency_matrix(ByAbsMatrix, mode = "undirected", weighted = TRUE)
plot(ByAbs)

# IMPROVING THE PLOT
#identify community groups/clusters using cluster_fast_greedy()
cfg = cluster_fast_greedy(as.undirected(ByAbs))
g_cfg = plot(cfg, as.undirected(ByAbs),vertex.label=V(ByAbs)$role,main="Fast Greedy")
# Set edge width based on weight:
E(ByAbs)$width <- E(ByAbs)$weight/3
# Calculate centrality measures
degree <- degree(ByAbs)
betweenness <- betweenness(ByAbs)
closeness <- closeness(ByAbs)
eigenvector <- eigen_centrality(ByAbs)$vector
# Combine centrality measures into a single score
combined_centrality <- (degree + betweenness + closeness + eigenvector) / 4
# Assign colors based on combined centrality score
node_colors <- heat.colors(max(combined_centrality) + 1)[combined_centrality + 1]
# Plot the graph with assigned colors
plot(ByAbs, vertex.color = node_colors)

################################################################################
# 1.6 Tokens single-mode network creation
# Multiply binary matrix by its transpose
ByTokenMatrix = t(dtmsx)%*%dtmsx
#Make leading diagonal zero
diag(ByTokenMatrix) = 0
#create graph object (Basic plot)
ByToken= graph_from_adjacency_matrix(ByTokenMatrix, mode = "undirected", weighted = TRUE)
plot(ByToken)

# IMPROVING THE PLOT
#identify community groups/clusters using cluster_fast_greedy()
cfg = cluster_fast_greedy(as.undirected(ByToken))
g_cfg = plot(cfg, as.undirected(ByToken),vertex.label=V(ByToken)$role,main="Fast Greedy")
# Set edge width based on weight:
E(ByToken)$width <- E(ByToken)$weight/3
```

```r
# Calculate centrality measures
degree <- degree(ByToken)
betweenness <- betweenness(ByToken)
closeness <- closeness(ByToken)
eigenvector <- eigen_centrality(ByToken)$vector
# Combine centrality measures into a single score
combined_centrality <- (degree + betweenness + closeness + eigenvector) / 4
# Assign colors based on combined centrality score
node_colors <- heat.colors(max(combined_centrality) + 1)[combined_centrality + 1]
# Plot the graph with assigned colors
plot(ByToken, vertex.color = node_colors)

################################################################################
# 1.7 Bipartite network creation
# Converting dtmsx to a data frame
dtmsa = as.data.frame(as.matrix(dtms))
# Create a new column with row names
dtmsa$ABS = rownames(dtmsa)
# Creating an empty data frame
dtmsb = data.frame()
# Iterating over the rows and columns of dtmsa
for (i in 1:nrow(dtmsa)){
  for (j in 1:(ncol(dtmsa)-1)){
    # Building the transformed data frame
    touse = cbind(dtmsa[i,j], dtmsa[i,ncol(dtmsa)], colnames(dtmsa[j]))
    dtmsb = rbind(dtmsb, touse)
  }
}
# Setting the proper column names
colnames(dtmsb)=c("weight", "abs", "token")
# Selecting only the rows where the "weight" column is not zero.
dtmsc = dtmsb[dtmsb$weight!=0,] #delete weights
# Reorder the columns
dtmsc = dtmsc[,c(2,3,1)]
dtmsc
dtmsc$weight <- as.numeric(dtmsc$weight)
# Create graph object and declare bipartite
g <-graph.data.frame(dtmsc, directed = FALSE)
bipartite.mapping(g) # perform bipartite mapping
# Assigning attributes
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)

# IMPROVING THE PLOT
#identify community groups/clusters using cluster_fast_greedy()
cfg = cluster_fast_greedy(as.undirected(g))
g_cfg = plot(cfg, as.undirected(g),vertex.label=V(g)$role, main="Fast Greedy")
# Set edge width based on weight:
E(g)$width <- E(g)$weight/3
# Calculate centrality measures
degree <- degree(g)
betweenness <- betweenness(g)
closeness <- closeness(g)
eigenvector <- eigen_centrality(g)$vector
# Combine centrality measures into a single score
combined_centrality <- (degree + betweenness + closeness + eigenvector) / 4
# Assign colors based on combined centrality score
```

```r
node_colors <- heat.colors(max(combined_centrality) + 1)[combined_centrality + 1]
# Plot the graph with assigned colors
plot(g, vertex.color = node_colors)
```