

Assignment 2

Smitesh Patil

2023-03-10

PART 1

Section 1: Set up the libraries and check the input data

```
#importing all the necessary libraries
library(igraph)
library(ggraph)
library(ggrepel)
library(kableExtra)
library(gt)
library(tidyr)
library(dplyr)

set.seed(200)
# reading the graph
g<- read_graph(file="./WordPairs.txt",format="pajek")
#removing directions
g<- as.undirected(g)
g<- simplify(g)

#reading the cue text file for setting the cue attribute for graph
cues <- read.table("./cue.txt", header = F, sep="\t", skip=4)

#setting the cue attribute
V(g)$cue<-cues[[1]]

#checking the diameter value
print(paste0("Diameter of the network: ", diameter(g, weights = NA)))
```

```
## [1] "Diameter of the network: 7"
```

```
check_cue_words <- function(target_node_name1, target_node_name2){
# test if the selected words are cue words

  if(V(g)[target_node_name1]$cue & V(g)[target_node_name2]$cue){
    cat("Both target words are cue words \n")
  }else{
```

```

    cat("Both target words are NOT cue words \n")
    cat(target_node_name1, "cue = ", as.logical(V(g)[target_node_name1]$cue ), "\n")
    cat(target_node_name2, "cue = ", as.logical(V(g)[target_node_name2]$cue ), "\n")
  }
}

```

Section 2 [Q1.1]: Create Random Walk and personalised page rank functions

Q1.1 Extract a word association sub-network around each pair of words by sampling the graph in the vicinity of each word. The sub-network for each pair of words should be between 100 and 200 words. The objective is to build a word association network around your pair of words that will capture all the semantic contexts your target words belong to – without bringing in irrelevant contexts.

```

#
# UTILITY FUNCTIONS for
# 1. creating word association network,
# 2. centralities
# 3. plot tables
# 4. plotting the graph
#in this cell

#-----Q1.1-----
#random walk function
random_walk_topic_network <- function(g,target_node_names, steps, walks, mode, topn){
  #initialising a empty vector
  vertices <- c()
  #looping on the two target names
  for (i in 1:2){
    #looping on the number of walks
    for (j in 1:walks){
      #running random walk and appending the vertices to the list
      vertices <- c(vertices, list(random_walk(g, target_node_names[i], steps, mode = mode)))
    }
  }

  #calculating the frequency of words appeared in the vertices vector and subsetting top n
  frequency_target <- head(sort(table(names(unlist(vertices)))), decreasing = TRUE), topn)
  #getting unique words
  unique_words <- names(frequency_target)

  #preparing and returning vertices in graph format
  vertices_in_word_association <- V(g)[name %in% unique_words]
  return(vertices_in_word_association)
}

#page rank function
page_rank_network <- function(g, target_node_names, topn, damping){
  #storing probabilities for all the vertices in the graph
  teleport_probs <- rep(0,vcount(g))
  teleport_probs[as.numeric(V(g)[target_node_names])] <- 1/length(target_node_names)
  #running page rank with parameters passed
  pr <- page_rank(g, directed = F, personalized=teleport_probs, damping = damping)$vector
  # getting top n vertices
  top_n_pr <- order(pr, decreasing=TRUE)[1:topn]
  top_n_pr<-V(g)[top_n_pr]
}

```

```

    #returning the vertices
    return(top_n_pr)
}

#-----Q1.2-----
#centrailites function for getting top vertices based on centrailites
centralities = function(word_association_network){
  #page rank centralities
  page_rank <- page_rank(word_association_network)$vector
  page_rank <- na.omit(page_rank[!names(page_rank) %in% c(target_word1, target_word2)])
  #getting five highest values
  page_rank <- sort(page_rank, decreasing = TRUE)[1:5]

  #betweenness centralities
  betweenness <- betweenness(word_association_network)
  betweenness <- betweenness[!names(betweenness) %in% c(target_word1, target_word2)]
  #getting five highest values
  betweenness <- sort(betweenness, decreasing = TRUE)[1:5]

  #eigen centralities
  eigen_centrality <- eigen_centrality(word_association_network)$vector
  eigen_centrality <- eigen_centrality[!names(eigen_centrality) %in% c(target_word1, target_word2)]
  #getting five highest values
  eigen_centrality <- sort(eigen_centrality, decreasing = TRUE)[1:5]

  #returning all the centralities in the list
  return(tibble("Page Rank" = names(page_rank) ,
                "Betweenness" = names(betweenness),
                "Eigen Centrality"= names(eigen_centrality)))
}

# plot the table dataframe
plot_table <- function(df, type){
  df %>%
  knitr::kable(
    format = "latex",
    align = "c",
    booktabs = TRUE,
    longtable = TRUE,
    linesep = "",
    caption = paste0("Centrality Score for three word pairs using ", type)
  ) %>%
  add_header_above(c("Index" = 1,
                    "Scores for wordpair Court & Lawyers" = 3,
                    "Scores for wordpair Children & Parents" = 3,
                    "Scores for wordpair King & Royal" = 3)) %>%
  kableExtra::kable_paper(
    position = "center",
    latex_options = c("striped", "repeat_header", "scale_down"),
    stripe_color = "gray!15",
    font_size = 8
  )
}

#-----Q1.3-----

```

```

#plotting the graph
plot_graph = function(association_network, vertex_size, label_size,
                      target_word1, target_word2, count, type){
  # getting top 25 nodes with high degree to label
  label_data <- names(sort(degree(association_network), decreasing = TRUE)[1:25])

  #setting vertex size
  vertex_size <- 2.5 + degree(association_network, mode = "all")/vertex_size
  cex_size <- 2 + degree(association_network, mode = "all")/label_size

  #graph creation
  ggraph(association_network, layout = "fr")+
  #creating edges
  geom_edge_link(start_cap = circle(2.5, "mm"),
                end_cap = circle(2.5, "mm"),
                edge_width = 0.2,
                alpha = 0.2)+
  #parameters for nodes
  geom_node_point(aes(size = vertex_size,
                    alpha = 0.8,
                    repel = TRUE,
                    # setting distinct color for word pair
                    colour = ifelse(V(association_network)$name %in%
                                   c(target_word1, target_word2), "yellow", "red"))+
  geom_node_text(
    # setting labels for word pair and top 25 nodes by degree
    aes(label = ifelse(V(association_network)$name %in% c(label_data, target_word1, target_word2),
                    name, NA)),
    fontface = "bold",
    position = "identity",
    size = cex_size,
    repel = TRUE

  ) +
  #title
  ggtitle(paste0("Figure No. ", count, " association network for words ", target_word1,
                " and ", target_word2, " using ", type))+
  #turning off legend
  guides(size = FALSE)+
  theme(plot.title = element_text(size = 10,
                                  hjust = 0.5))
}

```

Section 3: Simulate the function on three word pairs and justify the paramters

```

#target words
target_word1 <- "COURT"
target_word2 <- "LAWYER"

#checking the cue words
check_cue_words(target_word1, target_word2)

```

Both target words are cue words

```

#calling the function
out1_rand <- random_walk_topic_network(g, c(target_word1, target_word2), 3, 100, "all", 130)
out1_page <- page_rank_network(g, c(target_word1, target_word2), 140, 0.95)

#creating a subgraph with the vertices returned by random walk/ Personalised page rank
word_association_network1_rand <- induced.subgraph(g, out1_rand)
word_association_network1_page <- induced.subgraph(g, out1_page)

#target words
target_word1 <- "CHILDREN"
target_word2 <- "PARENTS"

#checking the cue words
check_cue_words(target_word1, target_word2)

```

```
## Both target words are cue words
```

```

#calling the function
out2_rand <- random_walk_topic_network(g, c(target_word1, target_word2), 3, 50, "all", 160)
out2_page <- page_rank_network(g, c(target_word1, target_word2), 140, 0.85)

#creating a subgraph with the vertices returned by random walk/ Personalised page rank
word_association_network2_rand <- induced.subgraph(g, out2_rand)
word_association_network2_page <- induced.subgraph(g, out2_page)

#target words
target_word1 <- "KING"
target_word2 <- "ROYAL"

#checking the cue words
check_cue_words(target_word1, target_word2)

```

```
## Both target words are cue words
```

```

#calling the function
out3_rand <- random_walk_topic_network(g, c(target_word1, target_word2), 3, 50, "all", 160)
out3_page <- page_rank_network(g, c(target_word1, target_word2), 150, 0.95)

#creating a subgraph with the vertices returned by random walk/ Personalised page rank
word_association_network3_rand <- induced.subgraph(g, out3_rand)
word_association_network3_page <- induced.subgraph(g, out3_page)

#calling the centralities function
centrality1_rand = centralities(word_association_network1_rand)
centrality2_rand = centralities(word_association_network2_rand)
centrality3_rand = centralities(word_association_network3_rand)

```

```

centrality1_page = centralities(word_association_network1_page)
centrality2_page = centralities(word_association_network2_page)
centrality3_page = centralities(word_association_network3_page)

#plot dataframe
centrality_random <- tibble(tibble("No."=1:5), centrality1_rand,
                                centrality2_rand,
                                centrality3_rand,
                                .name_repair = "minimal")

centrality_page <- tibble(tibble("No."=1:5), centrality1_page,
                              centrality2_page,
                              centrality3_page,
                              .name_repair = "minimal")
plot_table(df = centrality_random, type = "Random Walk")

```

Table 1: Centrality Score for three word pairs using Random Walk

Index	Scores for wordpair Court & Lawyers			Scores for wordpair Children & Parents			Scores for wordpair King & Royal		
No.	Page Rank	Betweenness	Eigen Centrality	Page Rank	Betweenness	Eigen Centrality	Page Rank	Betweenness	Eigen Centrality
1	COURT	COURT	BAD	CHILDREN	PARENTS	MOM	BLUE	POWER	QUEEN
2	LAWYER	LAWYER	GOOD	PARENTS	CHILDREN	DAD	QUEEN	FAMILY	CROWN
3	LAW	LAW	CORRUPT	BABY	MOTHER	PARENTS	MONEY	WEALTH	EMPEROR
4	MONEY	JUSTIFY	TROUBLE	CHILD	PAIN	MOTHER	HOME	MONEY	MONARCH
5	BAD	CLAIM	CRIME	ADULTS	AUTHORITY	PARENT	PRINCE	MASTER	PRINCE

```

plot_table(df = centrality_page, type = "Personalised Page Rank")

```

Table 2: Centrality Score for three word pairs using Personalised Page Rank

Index	Scores for wordpair Court & Lawyers			Scores for wordpair Children & Parents			Scores for wordpair King & Royal		
No.	Page Rank	Betweenness	Eigen Centrality	Page Rank	Betweenness	Eigen Centrality	Page Rank	Betweenness	Eigen Centrality
1	COURT	GOOD	BAD	CHILDREN	PARENTS	DAD	QUEEN	POWER	QUEEN
2	LAWYER	BAD	GOOD	CHILD	CHILDREN	MOM	BLUE	HOUSE	THRONE
3	LAW	HOUSE	WRONG	OLD	KIDS	MOTHER	HOUSE	LOVE	ROYALTY
4	JUDGE	SCHOOL	CRIME	PARENTS	SCHOOL	FATHER	RICH	GIRL	CROWN
5	BAD	JAIL	COURT	BAD	FRIEND	OLD	DOG	EMPIRE	EMPEROR

Section 3.1 [Q1.2]: Analysis of network and effects of Personalised Page Rank and Random walk on association network creation.

Three word pairs choose to create word network, they are “Court and Lawyer”, “Children and Parents”, and “King and Royal”. For creating the word association network, random walk algorithm was used for “Court and Lawyer” pair, and personalised pagerank algorithm was used for the other two pairs, The reason for choosing these specific algorithms were that they were optimal in creating word association network that would generate communities that were relatively coherent compared to others.

For Random Walk it was observed that as the number of walks and size of vertices increases, it increases the size of communities formed later during community detection and hence the number of walks for the one pair is restricted to 40 walks and 160 vertices.

For Personalised Page Rank every time nodes are ranked randomly some nodes are given more weight so that there is a chance for other nodes to be more relevant based on the damping factor, it was found that a high damping factor yields better results on community detection, i.e more communities of ideal size and coherent nodes.

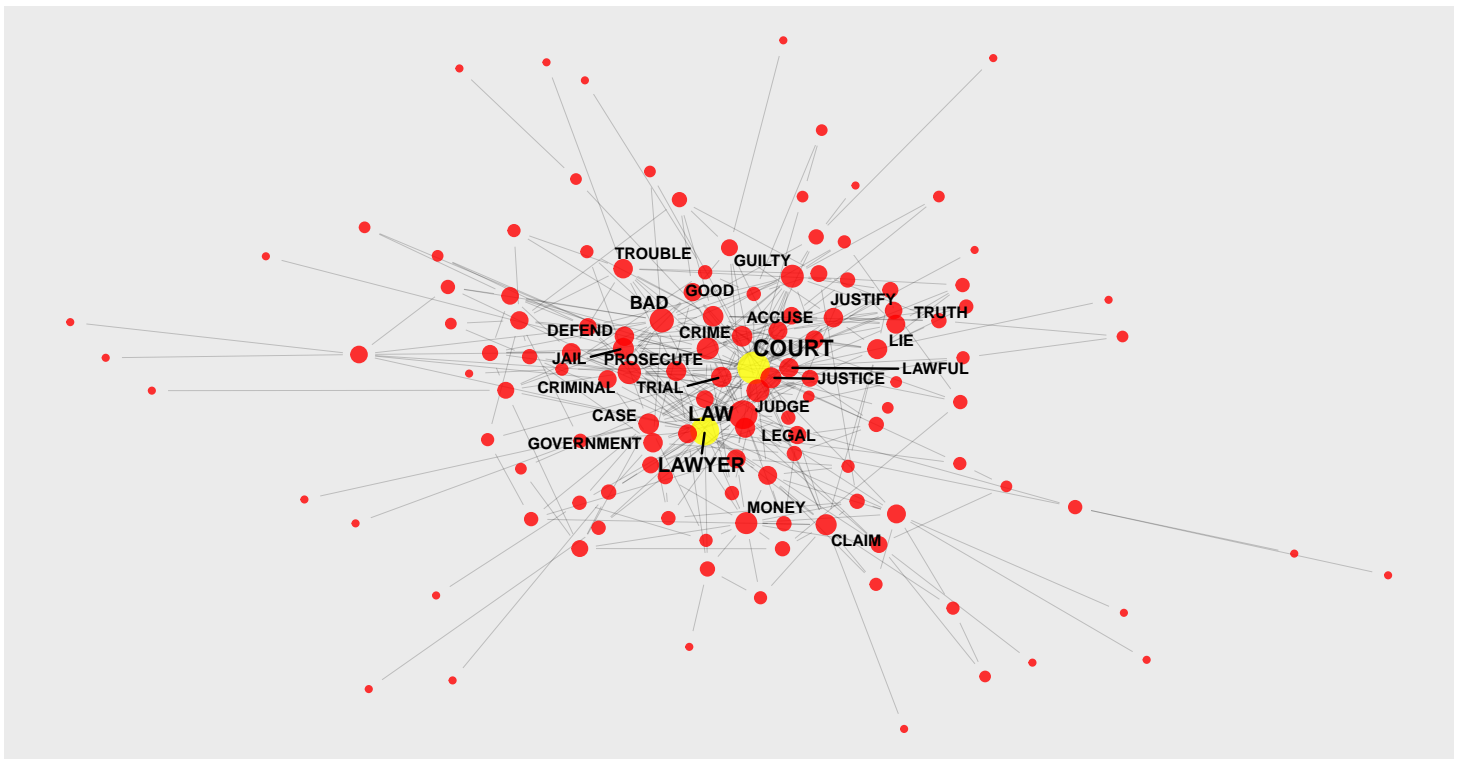
Section 3.2 [Q1.2]: Analysis of Centrality Measures on the word association networks.

Words of similar contexts seem to be appearing in the subgraph (association networks) created using both random walk algorithm and personalised page rank algorithm. The words from the word pairs seem to be top words based on centrality and authority measures as they are the root words used to generate the association networks this virtue of the networks makes sense. For example, for the first network generated using words “Court and Lawyers” seems to have both court and lawyer appear high on all of the centrality measures. Whereas, for the other two networks the words do appear in the list but they don’t seem to have the same level of centrality attribute as the first word network. This maybe because, the first network was created using random walk whereas the later two where created using personalised page rank.

Section 4 [Q1.3]:Produce a visualisation of the association network

```
#plotting network  
plot_graph(word_association_network1_rand, 10, 30, "COURT", "LAWYER", 1,  
           "Random Walk")
```

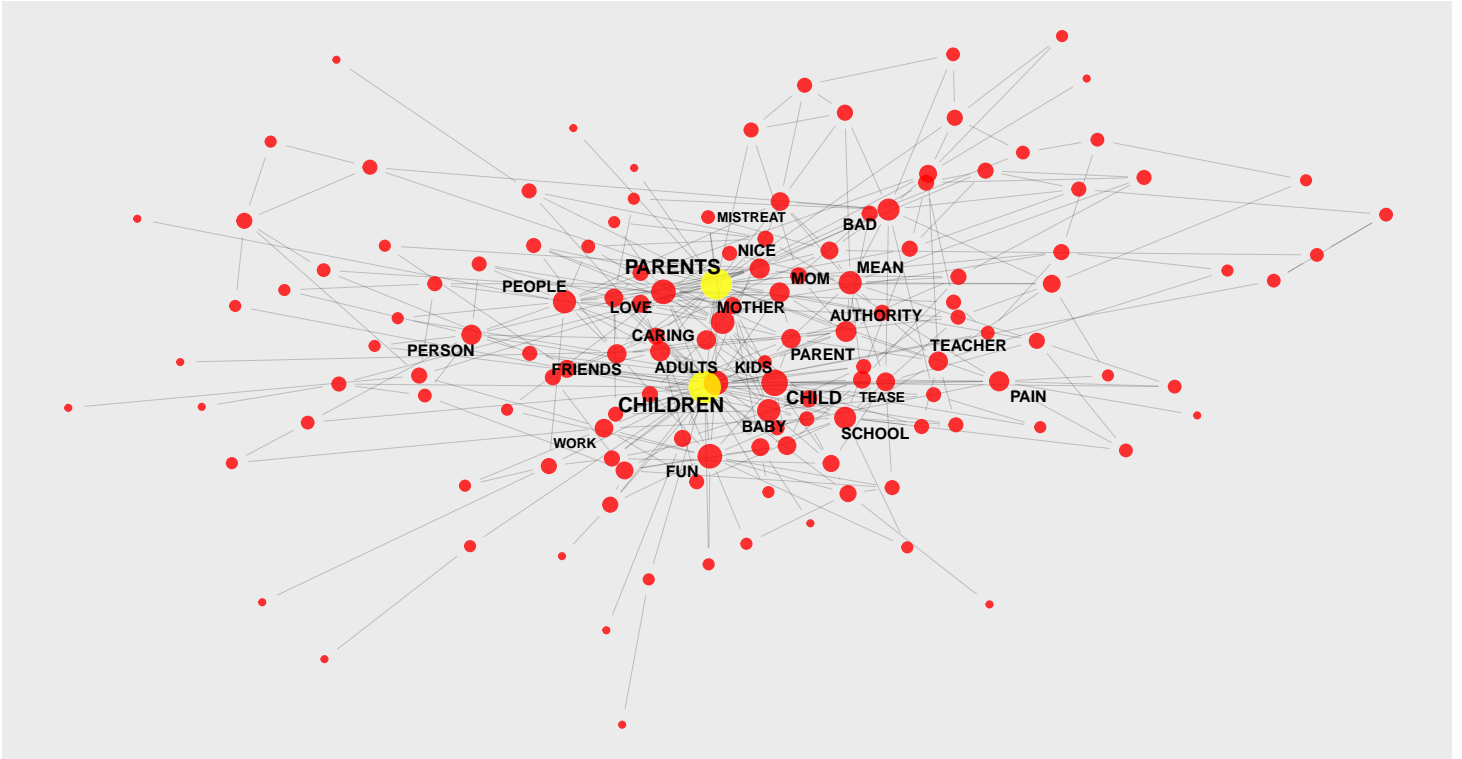
Figure No. 1 association network for words COURT and LAWYER using Random Walk



```
plot_graph(word_association_network1_page, 10, 30, "COURT", "LAWYER", 2,  
           "Personalised Page Rank")
```

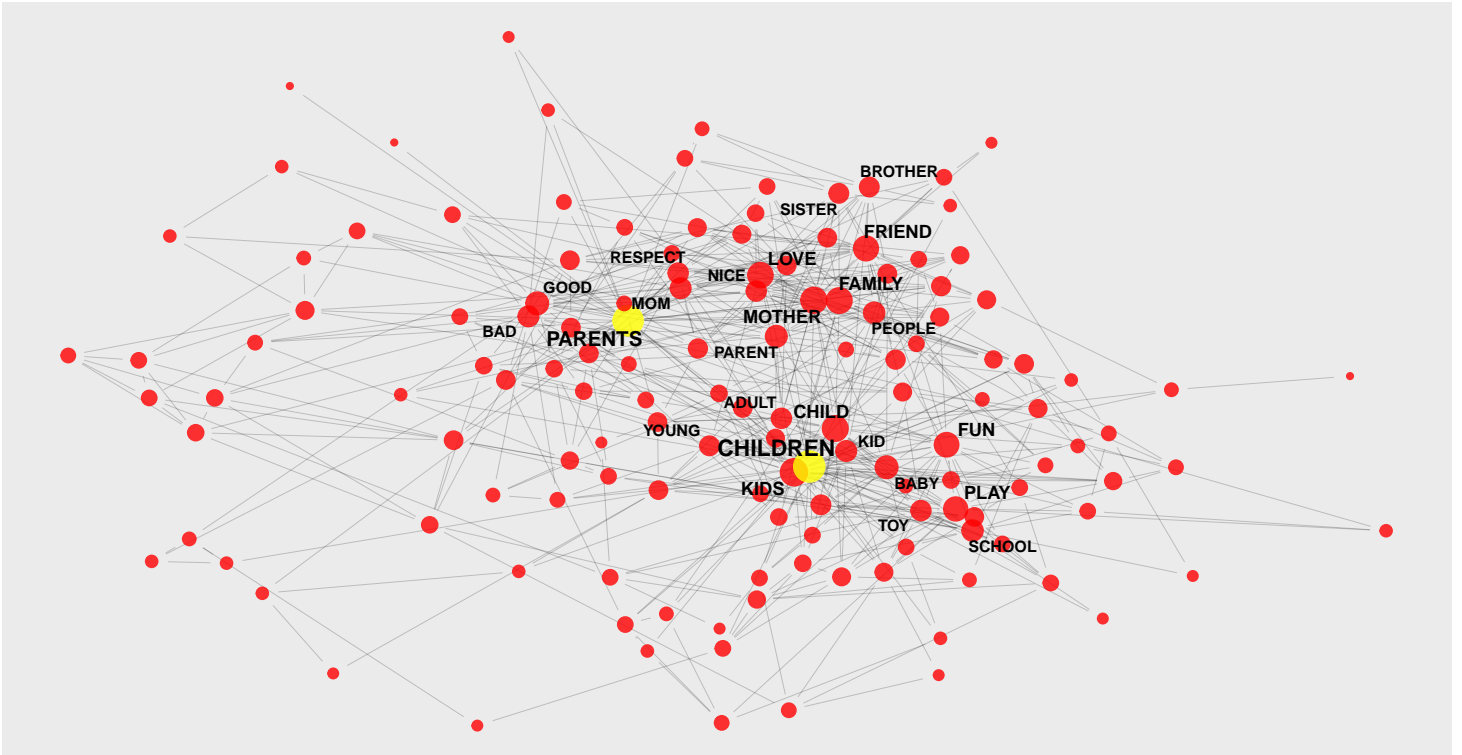
```
plot_graph(word_association_network2_rand, 10, 30, "CHILDREN", "PARENTS", 3,
           "Random Walk")
```


Figure No. 3 association network for words CHILDREN and PARENTS using Random Walk



```
plot_graph(word_association_network2_page, 10, 30, "CHILDREN", "PARENTS", 4,
"Personalised Page Rank")
```

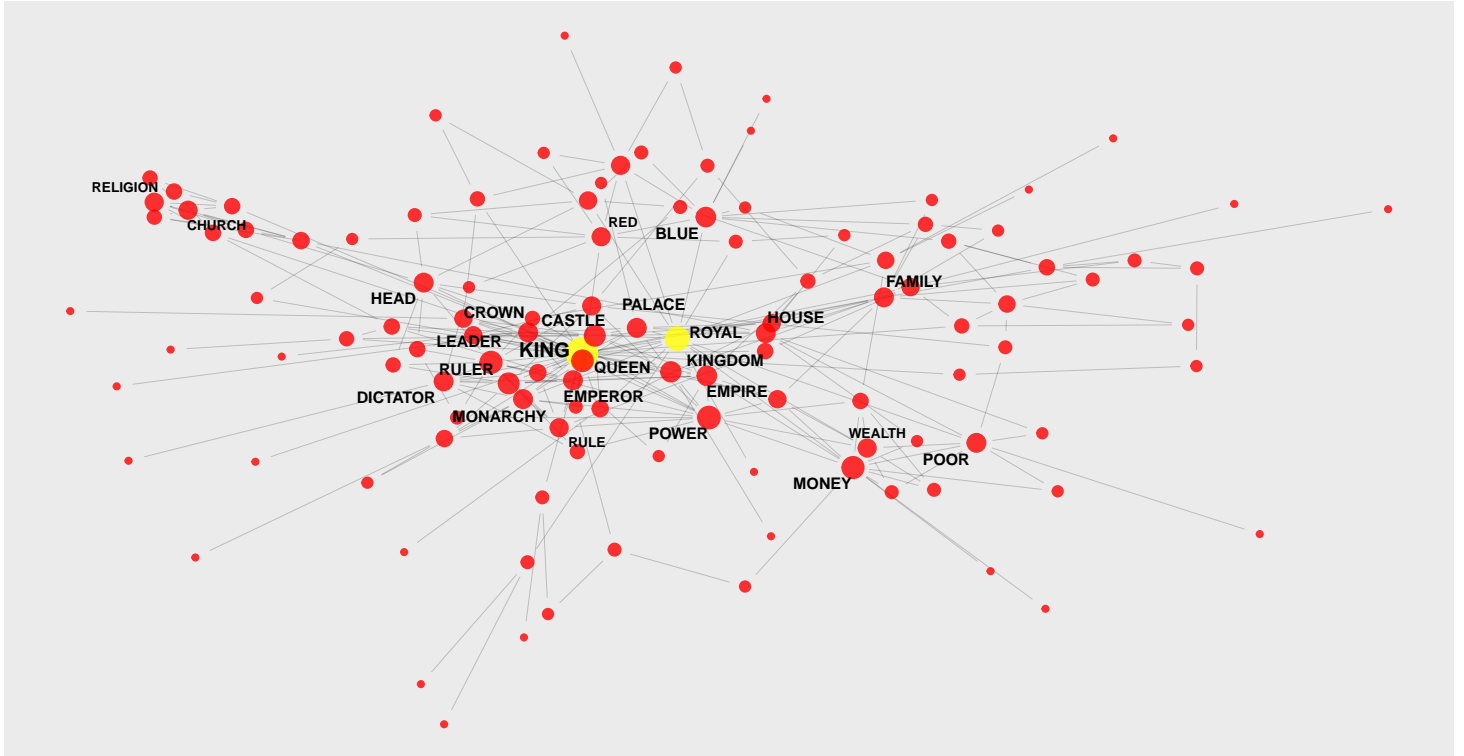
Figure No. 4 association network for words CHILDREN and PARENTS using Personalised Page Rank



```
#plotting the graph
```

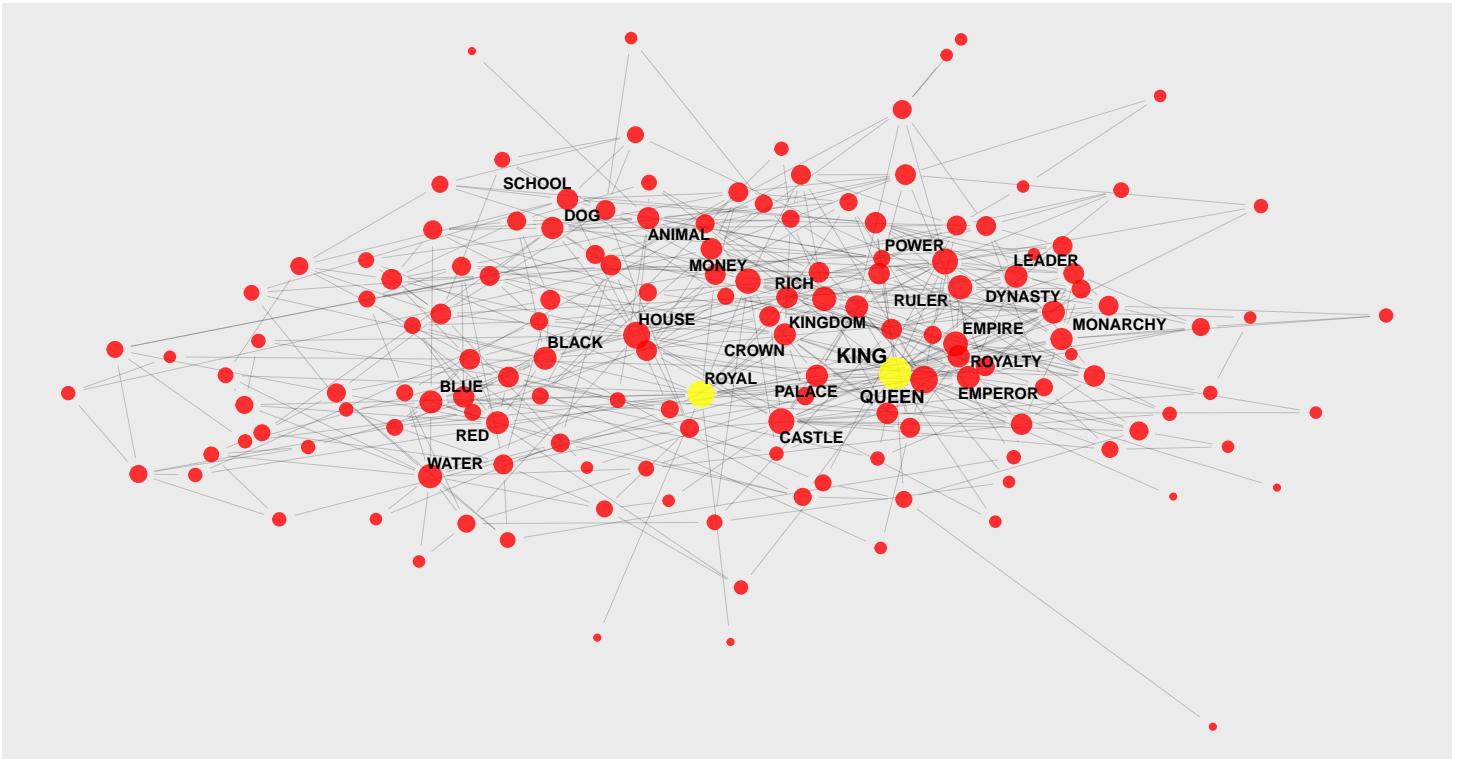
```
plot_graph(word_association_network3_rand, 10, 30, "KING", "ROYAL", 5,  
           "Random Walk")
```

Figure No. 5 association network for words KING and ROYAL using Random Walk



```
plot_graph(word_association_network3_page, 10, 30, "KING", "ROYAL", 6,  
           "Personalised Page Rank")
```

Figure No. 6 association network for words KING and ROYAL using Personalised Page Rank



PART 2

Section 5: Detect and display community tables

Q2.1 Using an appropriate community detection algorithm, determine the communities in each of the three word association networks you created in Part 1. You are expected to evaluate different community detection algorithms and you should justify the one you have selected.

Q2.2 Produce a table listing the words in each community. You should do this using a table formatting package such as Kable.

```
#creating optimal communities for word association network based on trial and error

cluster1_rand <- cluster_fluid_communities(word_association_network1_rand, no.of.communities = 13)

cluster2_rand <- cluster_walktrap(word_association_network2_rand)

cluster3_rand <- cluster_walktrap(word_association_network3_rand)

cluster1_page <- cluster_louvain(word_association_network1_page)

cluster2_page <- cluster_label_prop(word_association_network2_page)

cluster3_page <- cluster_edge_betweenness(word_association_network3_page)

#### UTILITY FUNCTIONS FOR TASK 2
# create community table

#create community table with size
create_community_table <- function(clustering_data){
  strings <- c()
```

```

lengths <- c()
#loop over each community
for(i in 1:length(clustering_data)){
  #append the clustering data into a string
  string = ""
  for(word in clustering_data[[i]]){
    string = paste0(string, word, " , ")
  }

  #store the string and the size
  strings <- append(strings, string)
  lengths <- append(lengths, length(clustering_data[[i]]))
}
#create a tibble
df <- tibble(strings, lengths) %>%
  filter(lengths > 1 )
colnames(df) <- c("Cluster", "Size")

#return the tibble
return(df)
}

#create and print the tables for the task
community_table1_rand <- create_community_table(cluster1_rand)

community_table1_page <- create_community_table(cluster1_page)

community_table2_rand <- create_community_table(cluster2_rand)

community_table2_page <- create_community_table(cluster2_page)

community_table3_rand <- create_community_table(cluster3_rand)

community_table3_page <- create_community_table(cluster3_page)

```

Section 5.1 [Q2.1, Q2.2] :

Various community detection algorithms were simulated on the association networks created on part 1.

Like the fast-greedy algorithm that joins pairs of granular communities if the move increases the combined community modularity in an iterative process.

Label Propagation algorithm, where every nodes are initially labelled and then iteratively each node adopts the label that the maximum number of its neighbors belong to, thus communities are formed in densely connected regions in the graph.

Louvain Algorithm which follows a bottom-up approach, where initially there are smaller set of communities which are converged using agglomerative clustering in two phases Identification of communities (based on modularity maximization) and folding of those communities.

Walktrap Algorithm, where the probability of distribution of a random walker reaching other nodes in the graph is considered while deciding whether two nodes should belong to same community or not.

Other than these algorithms based on module, fluid communities algorithm[1] was also tried out on word pair networks. Idea behind this approach is to think of nodes as fluids interacting in the network. Also, it allows us to identify an arbitrary number of communities as a parameter to our algorithm.

After a lot of trial and error it was found that the clustering algorithms set on word pair association networks above give the better results. Modularity of the clustering network and context of the communities found were used to quantify if the community detection results were good.

References:

[1] Parés F, Gasulla DG, et. al. (2018) Fluid Communities: A Competitive, Scalable and Diverse Community Detection Algorithm. In: Complex Networks & Their Applications VI: Proceedings of Complex Networks 2017 (The Sixth International Conference on Complex Networks and Their Applications), Springer, vol 689, p 229, doi: 10.1007/978-3-319-72150-7_19

```
#### UTILITY FUNCTIONS
# 1. create tibble with community label
# 2. plot the tibble with community labels

# function to create df with communities with label
define_community_labels <- function(cluster, dataframe){
  community_label = c()
  #looping on clusters
  for (i in 1:length(cluster)){
    # condition on length of cluster
    if (length(cluster[[i]]) > 1){
      #subsetting vertices
      vertices_in_community <- V(g)[name %in% cluster[[i]]]
      #create subgraph
      community_graph <- induced.subgraph(g, vertices_in_community)
      #calculating centrality
      page_rank <- page_rank(community_graph)$vector
      #highest ranked node as label
      label = sort(page_rank, decreasing = TRUE)[1]
      community_label <- c(community_label, label)
    }
  }

  #adding the community label by mutating
  dataframe<- dataframe %>%
    #adding the labels to the dataframe
    mutate(community_label = names(community_label)) %>%
    #arranging by order
    arrange(desc(Size)) %>%
    # adding the index row
    mutate(No = 1:nrow(dataframe)) %>%
    #discarding small clusters
    filter(Size > 3) %>%
    #setting order
    select(No, Cluster, community_label, Size) %>%

  return(dataframe)
}

# function to plot dataframe
plot_table = function(labelled_data, target_word1, target_word2, cluster, type){
  #plotting table
  labelled_data %>%
    #table printing function
    knitr::kable(
      format = "latex",
      align = "l",
      booktabs = TRUE,
      longtable = TRUE,
      linesep = "",

```

```

#printing the modularilty and word pairs
caption = paste0("Communities detected for word pair ", target_word1, " and ",
                target_word2, " generated from ",type,
                " approach"," Modularity ",
                round(modularity(cluster), 2))

) %>%
#column spec 6in
column_spec(column = 2, width = "6in") %>%
kableExtra::kable_paper(
  position = "left",
  latex_options = c("striped", "repeat_header, scale_down"),
  stripe_color = "gray!15"
)
}

#calling functions and plotting tables

clus1_comm_label_rand<-define_community_labels(cluster1_rand, community_table1_rand)
plot_table(clus1_comm_label_rand, "Court", "Lawyer", cluster1_rand,
           "Random Walk")

```

Table 3: Communities detected for word pair Court and Lawyer generated from Random Walk approach Modularity 0.31

No	Cluster	community_label	Size
1	TAKE , MONEY , DEGREE , FEDERAL , BILL , APPEAL , LAW , CHECK , ASSOCIATE , RELATE , SUE , CLAIMS , COMPLAINTS , ADJUSTER , LAWSUIT , LIBEL ,	MONEY	16
2	ACCUSE , GUILTY , PROSECUTE , COURT , TRIAL , PRESUME , HEARING , WITNESS , INNOCENT , EVIDENCE , PEOPLE'S , JUDGMENT , LOSS , DEFENDANT ,	COURT	14
3	BAD , CRIME , CRIMINAL , GUN , JAIL , COP , HOOK , ROBBER , CROOK , CROOKED , CORRUPT , CONVENT , GANGSTER ,	CRIMINAL	13
4	LAWYER , FIRM , RICH , LEGS , ATTORNEY , BROKEN , CURVED , BULL , EXTRAVAGANT , LAWS , LEGISLATURE ,	LAWYER	11
5	HELP , COUNSEL , ADVISE , CAUSE , GOVERNMENT , DOCTOR , ANATOMY , CLERK , JUST , UNJUST ,	HELP	10
6	JUDGE , ADJOURN , MARRIAGE , DIVORCE , CONTRACT , DISEASE , BRIBE , LEGAL , GAVEL , LEASE ,	JUDGE	10
7	DEFEND , CLAIM , OWN , ARMY , DECLARE , BATTLE , GOLD , RIGHTS , CONSTITUTION , DEFENSE ,	RIGHTS	10
8	BOOK , DATE , CASE , BRIEFCASE , BUSINESSMAN , WALLET , FILE , CLOSED , DO IT ,	BRIEFCASE	9
9	GOOD , CORRECT , BEER , ILLEGAL , PROVE , INNOCENCE , LAWFUL , LAW ABIDING ,	GOOD	8
10	LIE , CONFESS , TRUTH , ORDER , CONSEQUENCE , CONFESSION , FACT , TESTIFY ,	TRUTH	8
11	EXCUSE , DISMISS , INSURANCE , BUREAU , FAULT , JUSTICE , JUSTIFY ,	EXCUSE	7
12	FAST , ACE , KING , TENNIS , BASKETBALL , HOOP , RACQUETBALL ,	TENNIS	7
13	FREEDOM , TROUBLE , GULLIBLE , LIBERTY , FLUNK , INSULT , CRISIS ,	TROUBLE	7

```

clus1_comm_label_page<-define_community_labels(cluster1_page, community_table1_page)
plot_table(clus1_comm_label_page, "Court", "Lawyer", cluster1_page,
           "Personalised Page Rank")

```

Table 4: Communities detected for word pair Court and Lawyer
generated from Personalised Page Rank approach Modularity
0.67

No	Cluster	community_label	Size
1	WORK , SCHOOL , MATH , MONEY , SMART , BOOK , PAPER , LAWYER , TEST , JOB , FIRM , HARD , PROFESSIONAL , DOCTOR , CLOTHES , SLEEP , ATTORNEY , BRIEFCASE , NURSE , SUE , CLAIMS , LAWSUIT , LIBEL ,	WORK	23
2	PAIN , BLACK , HURT , FIGHT , STOP , WIN , PROSECUTE , DEFEND , ADJOURN , LEAVE , END , ARMY , HELP , COUNSEL , DEATH , GUN , RED , WAR , HAIR ,	HURT	19
3	COURT , JUDGE , TELL , TRIAL , KING , TALK , DAY , TIME , LIGHT , CASE , JUSTICE , WITNESS , EVIDENCE , JURY , JUDGMENT , GAVEL , VERDICT , TESTIFY ,	COURT	18
4	GOOD , BAD , CRIME , CRIMINAL , DRUGS , MEAN , JAIL , ILLEGAL , PRISON , LEGAL , UNLAWFUL , LAWFUL ,	BAD	12
5	SPORT , CAR , TENNIS , MOVIE , GAME , FUN , BASKETBALL , FOOTBALL , BALL , RACQUETBALL , RACKET ,	TENNIS	11
6	POLICE , MAN , OLD , GOVERNMENT , ORDER , BREAK , LAW , COP , RULES , RIGHTS , LAWS ,	LAW	11
7	ANIMAL , PEOPLE , TREE , GIRL , HOUSE , FRIEND , CLEAN , BIRD , DOG , HORSE ,	ANIMAL	10
8	SICK , FOOD , WATER , HOT , BOAT , FIRE , DRINK , COLD , SMELL , FISH ,	WATER	10
9	SEX , LOVE , MARRIAGE , DIVORCE , GOD , APPEAL , CHURCH ,	MARRIAGE	7
10	TAKE , STEAL , LIE , TRUTH , THIEF ,	STEAL	5
11	ACCUSE , BLAME , GUILTY , INNOCENT ,	GUILTY	4

```
clus2_comm_label_rand<-define_community_labels(cluster2_rand, community_table2_rand)
plot_table(clus2_comm_label_rand, "Children", "Parent", cluster2_rand,
          "Random Walk")
```

Table 5: Communities detected for word pair Children and Parent
generated from Random Walk approach Modularity 0.62

No	Cluster	community_label	Size
1	PEOPLE , HOME , LOVE , CARE , AFFAIR , DIVORCE , ECSTASY , WEDDING , FRIENDS , NEIGHBORHOOD , ELDERLY , CONTACTS , GRANDPARENTS , GHETTO ,	LOVE	14
2	MISSING , KIDS , WITHOUT , ADULTS , CHILDREN , PARENTS , GROWN-UPS , RESTRICTION , SUPPORT , BLOCKS , PLAY DOUGH , OBEY , GUARDIANS , FOLKS ,	CHILDREN	14
3	PARENT , MOTHER , AUTHORITY , MOM , MEAN , CARING , NICE , BOSS , RESPONSIBILITY , DAD , KINDNESS , STRICT , STUBBORN ,	MOM	13
4	WORK , SCHOOL , NONSENSE , MONEY , TEACH , TEACHER , MACHINE , EDUCATION ,	SCHOOL	8
5	CHILD , KID , BABY , INFANT , BRAT , REPRODUCE , CHILDISH , IMMATURE ,	BABY	8
6	GAME , FUN , BORING , ENTERTAIN , TEASE , CREATIVITY ,	FUN	6
7	INFLUENCE , PERSUADE , DISCOURAGE , SWAY , PEERS ,	PERSUADE	5
8	BIG , SMALL , LESS , SLENDER ,	SMALL	4
9	LAUGH , CRY , BURST , TICKLE ,	LAUGH	4
10	ALLOW , PERMISSION , FORBID , REFUSE ,	ALLOW	4
11	MISTREAT , BAD , DOG , TABOO ,	BAD	4

```
clus2_comm_label_page<-define_community_labels(cluster2_page, community_table2_page)
plot_table(clus2_comm_label_page, "Children", "Parent", cluster2_page,
          "Personalised Page Rank")
```

Table 6: Communities detected for word pair Children and Parent generated from Personalised Page Rank approach Modularity 0.65

No	Cluster	community_label	Size
1	KIDS , ADULTS , CHILDREN , PARENTS , GROWN-UPS , RED , TOYS , CLAY , PLAY DOUGH , DISOWN , GENE , GUARDIANS , FOLKS ,	CHILDREN	13
2	UNDERSTAND , SMART , YOUNG , MATURE , OLD , NEW , DEVELOP , GRANDMA , GRANDPARENTS ,	OLD	9
3	CHILD , RESPECT , ADULT , KID , PARENT , AUTHORITY , RESPONSIBILITY , DEVELOPMENT ,	CHILD	8
4	GOOD , BAD , INFLUENCE , OKAY , NICE , APPROVAL , PERSUADE , MORALS ,	GOOD	8
5	STOP , ALLOW , ASK , NO , LET , PERMIT , PERMISSION , FORBID ,	ALLOW	8
6	WORK , SCHOOL , MONEY , BOOK , TEACH , HORSE ,	SCHOOL	6
7	WATER , TREE , PARK , SWING , SLIDE , PLAYGROUND ,	PLAYGROUND	6
8	PLAY , GAME , FUN , GAMES , BALL , PLAYING ,	FUN	6
9	PEOPLE , PERSON , FRIEND , GROUP , DOG ,	PEOPLE	5
10	MUSIC , LOUD , NOISE , NOISY ,	LOUD	4
11	BABY , INFANT , NURSERY , RHYME ,	BABY	4
12	MOTHER , FATHER , CRITICAL , DAUGHTER ,	MOTHER	4
13	RESTRICTION , LAW , RULES , OBEY ,	RULES	4

```
clus3_comm_label_rand<-define_community_labels(cluster3_rand, community_table3_rand)
plot_table(clus3_comm_label_rand, "King", "Royal", cluster3_rand,
          "Random Walk")
```

Table 7: Communities detected for word pair King and Royal generated from Random Walk approach Modularity 0.67

No	Cluster	community_label	Size
1	CRAZY , KING , TALK , ENGLISH , ADORN , CROWN , LION , EVERYTHING , DEMOCRACY , ANARCHY , RULER , MONARCHY , RULE , KINGDOM , CARD , QUEEN , COMMUNIST , EMPIRE , CASTLE , ENGLAND , PALACE , DICTATOR , DOMINATE , ROYAL , MONARCH , DYNASTY , EMPEROR , STAR WARS , REGAL ,	KING	29
2	POWER , LOTS , MONEY , WEALTH , SUCCESS , SALE , DAMAGE , ECONOMIC , PRESTIGE ,	MONEY	9
3	TEMPLE , CHURCH , RELIGION , BISHOP , PRIEST , SHRINE , FRIAR , MONK , SYNAGOGUE ,	CHURCH	9
4	SKY , BLUE , ORANGE , RED , LAKE , PURPLE , INK , HAZE ,	BLUE	8
5	HEAD , MASTER , CHIEF , LEADER , BUMPS , CHAIRPERSON , COMMANDER , MAYOR ,	LEADER	8
6	HOME , HOUSE , FAMILY , CLEAN , BACKGROUND , DOMAIN , FLUSH ,	HOME	7
7	PAIN , PLEASURE , FOOTBALL , BRUISE , THORN , VIKING ,	PAIN	6
8	BOY , JUNGLE , MONKEY , GORGEOUS , GEORGE ,	GEORGE	5
9	FRIEND , PET , BROTHER , BEST , LOYAL ,	FRIEND	5
10	FOOD , YOGURT , BREAD , CHECKERS ,	FOOD	4
11	GRACEFUL , PRINCESS , PRINCE , CHARMING ,	PRINCE	4
12	WITHOUT , POOR , BEG , GHETTO ,	POOR	4

```
clus3_comm_label_page<-define_community_labels(cluster3_page, community_table3_page)
plot_table(clus3_comm_label_page, "King", "Royal", cluster3_page,
          "Personalised Page Rank")
```


Table 8: Communities detected for word pair King and Royal generated from Personalised Page Rank approach Modularity 0.36

No	Cluster	community_label	Size
1	BLACK , PAPER , TREE , GREEN , HOUSE , BLUE , FIRE , CLEAN , BIRD , RED , CAT , LIGHT , WHITE , DOG , HAIR , COLOR , FLOWER , PURPLE , BUTTERFLY ,	WHITE	19
2	ONE , COURT , KING , GOVERNMENT , MEASURE , ANARCHY , RULER , MONARCHY , RULE , PRESIDENT , LEADER , DICTATOR , MONARCH , OLIGARCHY , TOPPING ,	RULER	15
3	GOOD , FOOD , FIGHT , HAPPY , LOVE , ARMY , DRINK , GOD , SMELL , CHURCH , RELIGION , FRANCE ,	GOD	12
4	POWER , MONEY , BOOK , WEALTH , TELEVISION , JEWEL , RING , RICH , GOLD , DIAMOND , DYNASTY ,	MONEY	11
5	STRONG , PEOPLE , BAD , MAN , CITY , OLD , BIG , FRIEND , POOR , MUMMY ,	PEOPLE	10
6	WATER , RAIN , GAME , FUN , HOT , SUN , BALL , BEACH , LONDON , FLUSH ,	SUN	10
7	PAIN , SICK , HURT , FALL , HEAD , DEATH , SAD , GUN , COLD ,	HURT	9
8	ME , BEAUTIFUL , BEE , QUEEN , BRIDE , PRINCESS , PRINCE , ROYALTY ,	PRINCESS	8
9	MUSIC , COUNTRY , CLOTHES , CHINA , ENGLAND , ROMAN , EMPEROR ,	COUNTRY	7
10	WORK , MATH , HARD , HORSE , ARMOR , KNIGHT , MEDIEVAL ,	KNIGHT	7
11	HIGH , SKY , SCHOOL , MOVIE , BUILDING , EMPIRE ,	BUILDING	6
12	HOME , SAND , MANSION , CASTLE , MOAT , PALACE ,	CASTLE	6
13	ANIMAL , LION , FROG , KINGDOM , SNAKE ,	ANIMAL	5
14	TOP , CROWN , HAT , TOILET , THRONE ,	CROWN	5
15	CAR , SHIP , BOAT , FISH , DRAG ,	BOAT	5
16	GIRL , BEAUTY , BABY , SMALL , SLEEP ,	BEAUTY	5
17	FAMILY , ROYAL , PEASANT , REGAL ,	ROYAL	4

```

plot_table_interpretation = function(interpretation, target_word1, target_word2, type, cluster){
  interpretation %>%
    #index order set
    select(No, Cluster, Interpretation, community_label, Size) %>%
    #tables print
    knitr::kable(
      format = "latex",
      align = "l",
      booktabs = TRUE,
      longtable = TRUE,
      linesep = "",
      #printing the modularilty and word pairs
      caption = paste0("Communities detected for word pair ", target_word1, " and ",
        target_word2, " generated from ", type,
        " approach", " Modularity ",
        round(modularity(cluster), 2))
    ) %>%
    # column size define
    column_spec(column = 2, width = "4in") %>%
    column_spec(column = 3, width = "2in") %>%
    column_spec(column = 4, width = "1in") %>%
    column_spec(column = 5, width = "0.5in") %>%
    kableExtra::kable_paper(
      position = "left",
      latex_options = c("striped", "repeat_header", "scale_down"),
      stripe_color = "gray!15",
      font_size = 8
    )
}

```

```

}

# adding interpretation
interpretation1 <- clus1_comm_label_rand %>%
  mutate(
    Interpretation = c(
      "Terms related to money and law",
      "Words related to judiciary procesess carried out in court",
      "Words related to decribe a criminal or a criminal activity",
      "Attorney and law related terms",
      "Non-relavent",
      "Thinsg related to a judiciary process",
      "Non-relavent",
      "Physical objects",
      "A set of lawful and unlawful terms",
      "Things brought up during a court proceeding",
      "Non-relevant",
      "Related to sports",
      "Non-Relevant"
    )
  )

#plotting the community tables with interpretation
plot_table_interpretation(interpretation1,
  "Court", "Lawyer", "Random Walk", cluster1_rand)

```

Table 9: Communities detected for word pair Court and Lawyer generated from Random Walk approach Modularity 0.31

No	Cluster	Interpretation	community_label	Size
1	TAKE , MONEY , DEGREE , FEDERAL , BILL , APPEAL , LAW , CHECK , ASSOCIATE , RELATE , SUE , CLAIMS , COMPLAINTS , ADJUSTER , LAWSUIT , LIBEL ,	Terms related to money and law	MONEY	16
2	ACCUSE , GUILTY , PROSECUTE , COURT , TRIAL , PRESUME , HEARING , WITNESS , INNOCENT , EVIDENCE , PEOPLE'S , JUDGMENT , LOSS , DEFENDANT ,	Words related to judiciary procesess carried out in court	COURT	14
3	BAD , CRIME , CRIMINAL , GUN , JAIL , COP , HOOK , ROBBER , CROOK , CROOKED , CORRUPT , CONVENT , GANGSTER ,	Words related to decribe a criminal or a criminal activity	CRIMINAL	13
4	LAWYER , FIRM , RICH , LEGS , ATTORNEY , BROKEN , CURVED , BULL , EXTRAVAGANT , LAWS , LEGISLATURE ,	Attorney and law related terms	LAWYER	11
5	HELP , COUNSEL , ADVISE , CAUSE , GOVERNMENT , DOCTOR , ANATOMY , CLERK , JUST , UNJUST ,	Non-relavent	HELP	10
6	JUDGE , ADJOURN , MARRIAGE , DIVORCE , CONTRACT , DISEASE , BRIBE , LEGAL , GAVEL , LEASE ,	Thinsg related to a judiciary process	JUDGE	10
7	DEFEND , CLAIM , OWN , ARMY , DECLARE , BATTLE , GOLD , RIGHTS , CONSTITUTION , DEFENSE ,	Non-relavent	RIGHTS	10
8	BOOK , DATE , CASE , BRIEFCASE , BUSINESSMAN , WALLET , FILE , CLOSED , DO IT ,	Physical objects	BRIEFCASE	9
9	GOOD , CORRECT , BEER , ILLEGAL , PROVE , INNOCENCE , LAWFUL , LAW ABIDING ,	A set of lawful and unlawful terms	GOOD	8
10	LIE , CONFESS , TRUTH , ORDER , CONSEQUENCE , CONFESSION , FACT , TESTIFY ,	Things brought up during a court proceeding	TRUTH	8
11	EXCUSE , DISMISS , INSURANCE , BUREAU , FAULT , JUSTICE , JUSTIFY ,	Non-relevant	EXCUSE	7
12	FAST , ACE , KING , TENNIS , BASKETBALL , HOOP , RACQUETBALL ,	Related to sports	TENNIS	7
13	FREEDOM , TROUBLE , GULLIBLE , LIBERTY , FLUNK , INSULT , CRISIS ,	Non-Relevant	TROUBLE	7

```

# adding interpretation
interpretation2 = clus1_comm_label_page %>%
  mutate(

```

```

Interpretation = c(
  "Non-relevant",
  "Terms related to legal process",
  "Terms related to court proceedings",
  "Term related to crime",
  "Terms related to sport",
  "Terms related to law and order",
  "Non-Relevant",
  "Non-Relevant",
  "Terms related to intimacy",
  "Terms related to robbery",
  "Status of the convict"
)
)

#plotting the community tables with interpretation
plot_table_interpretation(interpretation2, "Court", "Lawyer",
  "Personalised Page Rank", cluster2_page)

```

Table 10: Communities detected for word pair Court and Lawyer generated from Personalised Page Rank approach Modularity 0.65

No	Cluster	Interpretation	community_label	Size
1	WORK , SCHOOL , MATH , MONEY , SMART , BOOK , PAPER , LAWYER , TEST , JOB , FIRM , HARD , PROFESSIONAL , DOCTOR , CLOTHES , SLEEP , ATTORNEY , BRIEFCASE , NURSE , SUE , CLAIMS , LAWSUIT , LIBEL ,	Non-relevant	WORK	23
2	PAIN , BLACK , HURT , FIGHT , STOP , WIN , PROSECUTE , DEFEND , ADJOURN , LEAVE , END , ARMY , HELP , COUNSEL , DEATH , GUN , RED , WAR , HAIR ,	Terms related to legal process	HURT	19
3	COURT , JUDGE , TELL , TRIAL , KING , TALK , DAY , TIME , LIGHT , CASE , JUSTICE , WITNESS , EVIDENCE , JURY , JUDGMENT , GAVEL , VERDICT , TESTIFY ,	Terms related to court proceedings	COURT	18
4	GOOD , BAD , CRIME , CRIMINAL , DRUGS , MEAN , JAIL , ILLEGAL , PRISON , LEGAL , UNLAWFUL , LAWFUL ,	Term related to crime	BAD	12
5	SPORT , CAR , TENNIS , MOVIE , GAME , FUN , BASKETBALL , FOOTBALL , BALL , RACQUETBALL , RACKET ,	Terms related to sport	TENNIS	11
6	POLICE , MAN , OLD , GOVERNMENT , ORDER , BREAK , LAW , COP , RULES , RIGHTS , LAWS ,	Terms related to law and order	LAW	11
7	ANIMAL , PEOPLE , TREE , GIRL , HOUSE , FRIEND , CLEAN , BIRD , DOG , HORSE ,	Non-Relevant	ANIMAL	10
8	SICK , FOOD , WATER , HOT , BOAT , FIRE , DRINK , COLD , SMELL , FISH ,	Non-Relevant	WATER	10
9	SEX , LOVE , MARRIAGE , DIVORCE , GOD , APPEAL , CHURCH ,	Terms related to intimacy	MARRIAGE	7
10	TAKE , STEAL , LIE , TRUTH , THIEF ,	Terms related to robbery	STEAL	5
11	ACCUSE , BLAME , GUILTY , INNOCENT ,	Status of the convict	GUILTY	4

```

# adding interpretation
interpretation3 = clus2_comm_label_rand %>%
  mutate(
    Interpretation = c(
      "Non-relevant",
      "Things related to children and their upbringing",
      "Things you'd relate to mother",
      "Things related to education",
      "Words related to infants",
      "Characteristics of activities",
      "Words related to influencing someone",
      "Characteristics of an object",
      "Related to laughter",

```

```

    "Related to consent",
    "Non-Relevant"
  )
)

#plotting the community tables with interpretation
plot_table_interpretation(interpretation3, "Children", "Parent",
                          "Random Walk", cluster2_rand)

```

Table 11: Communities detected for word pair Children and Parent generated from Random Walk approach Modularity 0.62

No	Cluster	Interpretation	community_label	Size
1	PEOPLE , HOME , LOVE , CARE , AFFAIR , DIVORCE , ECSTASY , WEDDING , FRIENDS , NEIGHBORHOOD , ELDERLY , CONTACTS , GRANDPARENTS , GHETTO ,	Non-relevant	LOVE	14
2	MISSING , KIDS , WITHOUT , ADULTS , CHILDREN , PARENTS , GROWN-UPS , RESTRICTION , SUPPORT , BLOCKS , PLAY DOUGH , OBEY , GUARDIANS , FOLKS ,	Things related to children and their upbringing	CHILDREN	14
3	PARENT , MOTHER , AUTHORITY , MOM , MEAN , CARING , NICE , BOSS , RESPONSIBILITY , DAD , KINDNESS , STRICT , STUBBORN ,	Things you'd relate to mother	MOM	13
4	WORK , SCHOOL , NONSENSE , MONEY , TEACH , TEACHER , MACHINE , EDUCATION ,	Things related to education	SCHOOL	8
5	CHILD , KID , BABY , INFANT , BRAT , REPRODUCE , CHILDISH , IMMATURE ,	Words related to infants	BABY	8
6	GAME , FUN , BORING , ENTERTAIN , TEASE , CREATIVITY ,	Characteristics of activities	FUN	6
7	INFLUENCE , PERSUADE , DISCOURAGE , SWAY , PEERS ,	Words related to influencing someone	PERSUADE	5
8	BIG , SMALL , LESS , SLENDER ,	Characteristics of an object	SMALL	4
9	LAUGH , CRY , BURST , TICKLE ,	Related to laughter	LAUGH	4
10	ALLOW , PERMISSION , FORBID , REFUSE ,	Related to consent	ALLOW	4
11	MISTREAT , BAD , DOG , TABOO ,	Non-Relevant	BAD	4

```

# adding interpretation
interpretation4 = clus2_comm_label_page %>%
  mutate(
    Interpretation = c(
      "Related to kids and parents",
      "Things you'd relate to grandparents",
      "None-relevant",
      "Characteristics of a person",
      "Actions words",
      "Related to school",
      "Related to school ground",
      "Related to sports",
      "Non-relevant",
      "Things associated with sound",
      "People/objects you'd find in a nursery",
      "Family relationships",
      "Related to law"
    )
  )

#plotting the community tables with interpretation
plot_table_interpretation(interpretation4, "Children", "Parent",
                          "Random Walk", cluster2_page)

```

Table 12: Communities detected for word pair Children and Parent generated from Random Walk approach Modularity 0.65

No	Cluster	Interpretation	community_label	Size
1	KIDS , ADULTS , CHILDREN , PARENTS , GROWN-UPS , RED , TOYS , CLAY , PLAY DOUGH , DISOWN , GENE , GUARDIANS , FOLKS ,	Related to kids and parents	CHILDREN	13
2	UNDERSTAND , SMART , YOUNG , MATURE , OLD , NEW , DEVELOP , GRANDMA , GRANDPARENTS ,	Things you'd relate to grandparents	OLD	9
3	CHILD , RESPECT , ADULT , KID , PARENT , AUTHORITY , RESPONSIBILITY , DEVELOPMENT ,	None-relevant	CHILD	8
4	GOOD , BAD , INFLUENCE , OKAY , NICE , APPROVAL , PERSUADE , MORALS ,	Characteristics of a person	GOOD	8
5	STOP , ALLOW , ASK , NO , LET , PERMIT , PERMISSION , FORBID ,	Actions words	ALLOW	8
6	WORK , SCHOOL , MONEY , BOOK , TEACH , HORSE ,	Related to school	SCHOOL	6
7	WATER , TREE , PARK , SWING , SLIDE , PLAYGROUND ,	Related to school ground	PLAYGROUND	6
8	PLAY , GAME , FUN , GAMES , BALL , PLAYING ,	Related to sports	FUN	6
9	PEOPLE , PERSON , FRIEND , GROUP , DOG ,	Non-relevant	PEOPLE	5
10	MUSIC , LOUD , NOISE , NOISY ,	Things associated with sound	LOUD	4
11	BABY , INFANT , NURSERY , RHYME ,	People/objects you'd find in a nursery	BABY	4
12	MOTHER , FATHER , CRITICAL , DAUGHTER ,	Family relationships	MOTHER	4
13	RESTRICTION , LAW , RULES , OBEY ,	Related to law	RULES	4

```
# adding interpretation
interpretation5 = clus3_comm_label_rand %>%
  mutate(
    Interpretation = c(
      "Behaviour associated with a king/leader",
      "Related to economics",
      "Buildings of worship amongnst religions along with religious leaders",
      "Non-relevant",
      "Head of a group/kingdom/country",
      "Things present in a home",
      "Related to sport/fight",
      "Non-relevant",
      "Words associated with a loyal friend",
      "Related to food",
      "Associated with a good looking person",
      "Associated with poverty"
    )
  )

#plotting the community tables with interpretation
plot_table_interpretation(interpretation5, "Children", "Parent",
                          "Random Walk", cluster3_rand)
```

Table 13: Communities detected for word pair Children and Parent generated from Random Walk approach Modularity 0.67

No	Cluster	Interpretation	community_label	Size
1	CRAZY , KING , TALK , ENGLISH , ADORN , CROWN , LION , EVERYTHING , DEMOCRACY , ANARCHY , RULER , MONARCHY , RULE , KINGDOM , CARD , QUEEN , COMMUNIST , EMPIRE , CASTLE , ENGLAND , PALACE , DICTATOR , DOMINATE , ROYAL , MONARCH , DYNASTY , EMPEROR , STAR WARS , REGAL ,	Behaviour associated with a king/leader	KING	29
2	POWER , LOTS , MONEY , WEALTH , SUCCESS , SALE , DAMAGE , ECONOMIC , PRESTIGE ,	Related to economics	MONEY	9
3	TEMPLE , CHURCH , RELIGION , BISHOP , PRIEST , SHRINE , FRIAR , MONK , SYNAGOGUE ,	Buildings of worship amongnst religions along with religious leaders	CHURCH	9
4	SKY , BLUE , ORANGE , RED , LAKE , PURPLE , INK , HAZE ,	Non-relevant	BLUE	8

5	HEAD , MASTER , CHIEF , LEADER , BUMPS , CHAIRPERSON , COMMANDER , MAYOR ,	Head of a group/kingdom/country	LEADER	8
6	HOME , HOUSE , FAMILY , CLEAN , BACKGROUND , DOMAIN , FLUSH ,	Things present in a home	HOME	7
7	PAIN , PLEASURE , FOOTBALL , BRUISE , THORN , VIKING ,	Related to sport/fight	PAIN	6
8	BOY , JUNGLE , MONKEY , GORGEOUS , GEORGE ,	Non-relevant	GEORGE	5
9	FRIEND , PET , BROTHER , BEST , LOYAL ,	Words associated with a loyal friend	FRIEND	5
10	FOOD , YOGURT , BREAD , CHECKERS ,	Related to food	FOOD	4
11	GRACEFUL , PRINCESS , PRINCE , CHARMING ,	Associated with a good looking person	PRINCE	4
12	WITHOUT , POOR , BEG , GHETTO ,	Associated with poverty	POOR	4

adding interpretation

```
interpretation6 <- clus3_comm_label_page %>%
```

```
  mutate(
```

```
    Interpretation = c(
```

```
      "Non-relevant",
```

```
      "Words associated with a powerful leader",
```

```
      "Related to religion",
```

```
      "Related to abundance/plenty",
```

```
      "Virtues/vices associated with people",
```

```
      "Associated with outdoor activities",
```

```
      "Emotions/Things felt in sadness",
```

```
      "Feminine traits associated with royalty",
```

```
      "Non-relevant",
```

```
      "Related to medieval soldiers",
```

```
      "Non-relevant",
```

```
      "Related to a large mansion",
```

```
      "Types of animals",
```

```
      "Non-relevant",
```

```
      "Related to the sea/fishery",
```

```
      "Related to feminine traits",
```

```
      "Synonyms and Antonyms to royalty"
```

```
    )
```

```
  )
```

#plotting the community tables with interpretation

```
plot_table_interpretation(interpretation6, "Children", "Parent",
```

```
  "Random Walk", cluster3_page)
```

Table 14: Communities detected for word pair Children and Parent generated from Random Walk approach Modularity 0.36

No	Cluster	Interpretation	community_label	Size
1	BLACK , PAPER , TREE , GREEN , HOUSE , BLUE , FIRE , CLEAN , BIRD , RED , CAT , LIGHT , WHITE , DOG , HAIR , COLOR , FLOWER , PURPLE , BUTTERFLY ,	Non-relevant	WHITE	19
2	ONE , COURT , KING , GOVERNMENT , MEASURE , ANARCHY , RULER , MONARCHY , RULE , PRESIDENT , LEADER , DICTATOR , MONARCH , OLIGARCHY , TOPPING ,	Words associated with a powerful leader	RULER	15
3	GOOD , FOOD , FIGHT , HAPPY , LOVE , ARMY , DRINK , GOD , SMELL , CHURCH , RELIGION , FRANCE ,	Related to religion	GOD	12
4	POWER , MONEY , BOOK , WEALTH , TELEVISION , JEWEL , RING , RICH , GOLD , DIAMOND , DYNASTY ,	Related to abundance/plenty	MONEY	11
5	STRONG , PEOPLE , BAD , MAN , CITY , OLD , BIG , FRIEND , POOR , MUMMY ,	Virtues/vices associated with people	PEOPLE	10
6	WATER , RAIN , GAME , FUN , HOT , SUN , BALL , BEACH , LONDON , FLUSH ,	Associated with outdoor activities	SUN	10
7	PAIN , SICK , HURT , FALL , HEAD , DEATH , SAD , GUN , COLD ,	Emotions/Things felt in sadness	HURT	9
8	ME , BEAUTIFUL , BEE , QUEEN , BRIDE , PRINCESS , PRINCE , ROYALTY ,	Feminine traits associated with royalty	PRINCESS	8
9	MUSIC , COUNTRY , CLOTHES , CHINA , ENGLAND , ROMAN , EMPEROR ,	Non-relevant	COUNTRY	7
10	WORK , MATH , HARD , HORSE , ARMOR , KNIGHT , MEDIEVAL ,	Related to medieval soldiers	KNIGHT	7

11	HIGH , SKY , SCHOOL , MOVIE , BUILDING , EMPIRE ,	Non-relevant	BUILDING	6
12	HOME , SAND , MANSION , CASTLE , MOAT , PALACE ,	Related to a large mansion	CASTLE	6
13	ANIMAL , LION , FROG , KINGDOM , SNAKE ,	Types of animals	ANIMAL	5
14	TOP , CROWN , HAT , TOILET , THRONE ,	Non-relevant	CROWN	5
15	CAR , SHIP , BOAT , FISH , DRAG ,	Related to the sea/fishery	BOAT	5
16	GIRL , BEAUTY , BABY , SMALL , SLEEP ,	Related to feminine traits	BEAUTY	5
17	FAMILY , ROYAL , PEASANT , REGAL ,	Synonyms and Antonyms to royalty	ROYAL	4