ISOM 673 Final Project

Tingting Brill

2022-12-08

Set up

```
setwd("C:/Users/Tingting Brill/Desktop/MSBA/ISOM673/Final Project")
library(data.table)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:data.table':
##
##     between, first, last

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(igraph)
```

```
##
## Attaching package: 'igraph'

## The following objects are masked from 'package:dplyr':
##
##     as_data_frame, groups, union

## The following objects are masked from 'package:stats':
##
##     decompose, spectrum

## The following object is masked from 'package:base':
##
##     union
```

Data Cleaning

```
df <- fread("chapter_edge.csv",header = FALSE)

df <- t(df)
df <- as.data.table(df)
colnames(df) <- c("relation")


df <- df[c(-88,-89,-108,-120),]


#df$chapter <-chapter


#blah <- unlist(strsplit(df[,1], "\\)"))

cur_relationship <- as.data.frame(matrix(ncol=4, nrow=0))
colnames(cur_relationship) <- c("Person1","Person2","Weight","Chapter")
#chapter <- seq(1,120)

#length(df$relation)
#df[88]
for (i in 1:nrow(df)){
  lol <- unlist(strsplit(df$relation[i], ","))
  max <- length(lol)
  Person1 <- c()
  Person2 <- c()
  Weight <- c()
  Chapter <- c()
    for (y in 1:max){
     if (y%%4==1){
        Person1[y] <- lol[y]}
      else if (y%%4==2){
        Person2[y] <- lol[y]}
      else if (y%%4==3){
        Weight[y] <- lol[y]}
      else{
        Chapter[y] <- lol[y]
      }
    }
  Person1 <- gsub(" ", "", Person1)
  Person1<-gsub("[()']","",as.character(Person1))
  Person1<-gsub("\\[","",as.character(Person1))
  Person1 <- as.data.frame(Person1)
  Person1 <- Person1%>%
    filter(!is.na(Person1))


  Person2 <- gsub(" ", "", Person2)
  Person2<-gsub("[()']","",as.character(Person2))
  Person2 <- as.data.frame(Person2)
  Person2 <- Person2%>%
```

```
    filter(!is.na(Person2))

  pair <- cbind(Person1,Person2)
  for (p in 1:nrow(pair)){
    pair[p, ] = sort(pair[p, ])
  }

  Weight <- gsub(" ", "", Weight)
  Weight<-gsub("[()']","",as.character(Weight))
  Weight <- as.data.frame(Weight)
  Weight <- Weight%>%
    filter(!is.na(Weight))

  Chapter <- gsub(" ", "", Chapter)
  Chapter<-gsub("[()']","",as.character(Chapter))
  Chapter<-gsub("\\]","",as.character(Chapter))
  Chapter <- as.data.frame(Chapter)
  Chapter <- Chapter%>%
    filter(!is.na(Chapter))

  Relationship_new <- cbind(pair,Weight,Chapter)
  Relationship_new <- Relationship_new%>%
    unique()
  #for (i in 1:nrow(Relationship_new)){
    #Relationship_new[i, ] = sort(Relationship_new[i,1:2])
  #}
  Relationship_new <- rbind(cur_relationship,Relationship_new)
  cur_relationship <- Relationship_new
}


Relationship <- cur_relationship

remove(Person1,Person2,Weight,lol,max,Relationship_new,cur_relationship,df,Chapter,pair)

Relationship$Chapter <- as.integer(Relationship$Chapter)

for (i in 1:nrow(Relationship)){
  if(Relationship$Chapter[i]>=111){
    Relationship$section[i] <-"Section12"}
  else if (Relationship$Chapter[i]>=101& Relationship$Chapter[i]<=110 ){
    Relationship$section[i] <-  "Section11"}
  else if (Relationship$Chapter[i]>=91& Relationship$Chapter[i]<=100 ){
    Relationship$section[i] <-  "Section10"}
  else if (Relationship$Chapter[i]>=81& Relationship$Chapter[i]<=90 ){
    Relationship$section[i] <-  "Section9"}
  else if (Relationship$Chapter[i]>=71& Relationship$Chapter[i]<=80  ){
    Relationship$section[i] <-  "Section8"}
  else if (Relationship$Chapter[i]>=61& Relationship$Chapter[i]<=70 ){
    Relationship$section[i] <-  "Section7"}
  else if (Relationship$Chapter[i]>=51& Relationship$Chapter[i]<=60 ){
    Relationship$section[i] <-  "Section6"}
  else if (Relationship$Chapter[i]>=41& Relationship$Chapter[i]<=50 ){
```

```
    Relationship$section[i] <-  "Section5"}
  else if (Relationship$Chapter[i]>=31& Relationship$Chapter[i]<=40 ){
    Relationship$section[i] <-  "Section4"}
  else if (Relationship$Chapter[i]>=21& Relationship$Chapter[i]<=30 ){
    Relationship$section[i] <-  "Section3"}
  else if (Relationship$Chapter[i]>=11& Relationship$Chapter[i]<=20 ){
    Relationship$section[i] <-  "Section2"}
  else{
    Relationship$section[i] <- "Section1"
  }
}

Relationship$Weight <- as.numeric(Relationship$Weight)
Relationship1 <- Relationship%>%
  group_by(Person1,Person2)%>%
  mutate(over_all_weight=sum(Weight))%>%
  select(-Weight,-Chapter)%>%
  unique()%>%
  ungroup()
```

```
#characters$V1 <- stringr::str_to_title(characters$V1)
#characters$V2 <- stringr::str_to_title(characters$V2)
#for (i in 1:nrow(characters)){
  #characters$name[i] = paste0(characters$V1[i],characters$V2[i])
#}

#characters <- characters%>%
  #select(-V1,-V2)%>%
  #arrange(V3,name)

#colnames(characters) <- c("Country","Name")

#write.csv(characters,"character_country.csv",col.names = TRUE,row.name=FALSE)
characters <- fread("character_country.csv",header = TRUE)
```

Creating Graphs for each section

```
s1 <- Relationship1 %>% filter(section=="Section1")
g.s1 <- graph.data.frame(s1, directed=FALSE)
#g1 <- simplify(g.s1)
plot(g.s1)
```


```
V(g.s1)$degree <- degree(g.s1, mode="all")
V(g.s1)$betweeness <- betweenness(g.s1)
V(g.s1)$closeness <- closeness(g.s1)
n.s1 <- as.data.frame(V(g.s1)$name)
b.s1 <- as.data.frame(V(g.s1)$betweeness)
d.s1 <- as.data.frame(V(g.s1)$degree)
c.s1 <- as.data.frame(V(g.s1)$closeness)
s1_final <- cbind(n.s1,b.s1,d.s1,c.s1)
s1_final$Section <- 1
colnames(s1_final) <- c("Name","Betweenness","Degree","Closeness","Section")
remove(n.s1,b.s1,d.s1,c.s1)
```

```
s2 <- Relationship1 %>% filter(section=="Section2")
g.s2 <- graph.data.frame(s2, directed=FALSE)
#g2 <- simplify(g.s2)
plot(g.s2)
```


```
V(g.s2)$degree <- degree(g.s2, mode="all")
V(g.s2)$betweeness <- betweenness(g.s2)
V(g.s2)$closeness <- closeness(g.s2)
n.s2 <- as.data.frame(V(g.s2)$name)
b.s2 <- as.data.frame(V(g.s2)$betweeness)
d.s2 <- as.data.frame(V(g.s2)$degree)
c.s2 <- as.data.frame(V(g.s2)$closeness)
s2_final <- cbind(n.s2,b.s2,d.s2,c.s2)
s2_final$Section <- 2
colnames(s2_final) <- c("Name","Betweenness","Degree","Closeness","Section")
remove(n.s2,b.s2,d.s2,c.s2)
```

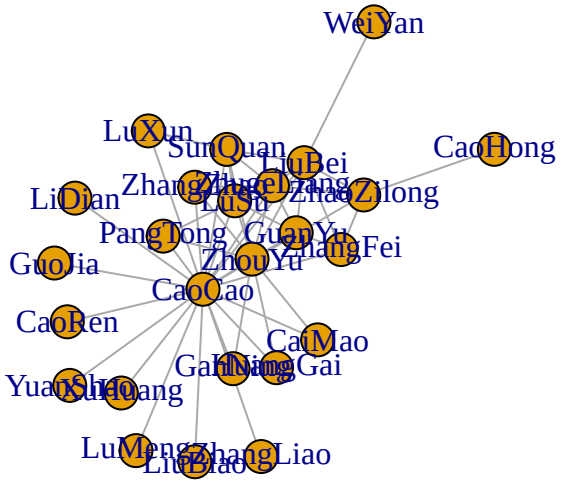
```
s3 <- Relationship1 %>% filter(section=="Section3")
g.s3 <- graph.data.frame(s3, directed=FALSE)
#g3 <- simplify(g.s3)
plot(g.s3)
```


```
V(g.s3)$degree <- degree(g.s3, mode="all")
V(g.s3)$betweeness <- betweenness(g.s3)
V(g.s3)$closeness <- closeness(g.s3)
n.s3 <- as.data.frame(V(g.s3)$name)
b.s3 <- as.data.frame(V(g.s3)$betweeness)
d.s3 <- as.data.frame(V(g.s3)$degree)
c.s3 <- as.data.frame(V(g.s3)$closeness)
s3_final <- cbind(n.s3,b.s3,d.s3,c.s3)
s3_final$Section <- 3
colnames(s3_final) <- c("Name","Betweenness","Degree","Closeness","Section")
remove(n.s3,b.s3,d.s3,c.s3)
```

```
s4 <- Relationship1 %>% filter(section=="Section4")
g.s4 <- graph.data.frame(s4, directed=FALSE)
#g4 <- simplify(g.s4)
plot(g.s4)
```


```
V(g.s4)$degree <- degree(g.s4, mode="all")
V(g.s4)$betweeness <- betweenness(g.s4)
V(g.s4)$closeness <- closeness(g.s4)
n.s4 <- as.data.frame(V(g.s4)$name)
b.s4 <- as.data.frame(V(g.s4)$betweeness)
d.s4 <- as.data.frame(V(g.s4)$degree)
c.s4 <- as.data.frame(V(g.s4)$closeness)
s4_final <- cbind(n.s4,b.s4,d.s4,c.s4)
s4_final$Section <- 4
colnames(s4_final) <- c("Name","Betweenness","Degree","Closeness","Section")
remove(n.s4,b.s4,d.s4,c.s4)
```

```
s5 <- Relationship1 %>% filter(section=="Section5")
g.s5 <- graph.data.frame(s5, directed=FALSE)
#g5 <- simplify(g.s5)
plot(g.s5)
```




```
V(g.s5)$degree <- degree(g.s5, mode="all")
V(g.s5)$betweeness <- betweenness(g.s5)
V(g.s5)$closeness <- closeness(g.s5)
n.s5 <- as.data.frame(V(g.s5)$name)
b.s5 <- as.data.frame(V(g.s5)$betweeness)
d.s5 <- as.data.frame(V(g.s5)$degree)
c.s5 <- as.data.frame(V(g.s5)$closeness)
s5_final <- cbind(n.s5,b.s5,d.s5,c.s5)
s5_final$Section <- 5
colnames(s5_final) <- c("Name","Betweenness","Degree","Closeness","Section")
remove(n.s5,b.s5,d.s5,c.s5)
```

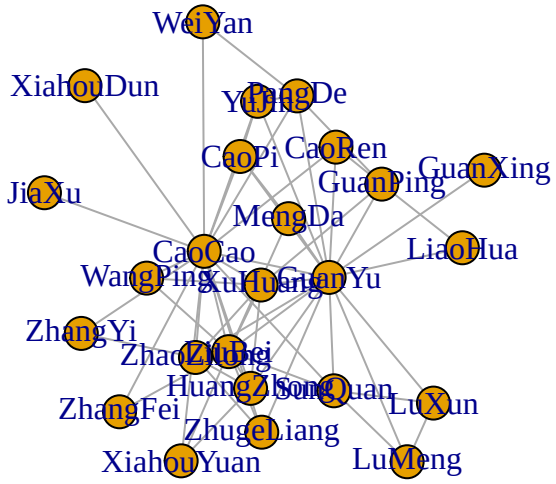
```
s6 <- Relationship1 %>% filter(section=="Section6")
g.s6 <- graph.data.frame(s6, directed=FALSE)
#g6 <- simplify(g.s6)
plot(g.s6)
```


```
V(g.s6)$degree <- degree(g.s6, mode="all")
V(g.s6)$betweeness <- betweenness(g.s6)
V(g.s6)$closeness <- closeness(g.s6)
n.s6 <- as.data.frame(V(g.s6)$name)
b.s6 <- as.data.frame(V(g.s6)$betweeness)
d.s6 <- as.data.frame(V(g.s6)$degree)
c.s6 <- as.data.frame(V(g.s6)$closeness)
s6_final <- cbind(n.s6,b.s6,d.s6,c.s6)
s6_final$Section <- 6
colnames(s6_final) <- c("Name","Betweenness","Degree","Closeness","Section")
remove(n.s6,b.s6,d.s6,c.s6)
```

```
s7 <- Relationship1 %>% filter(section=="Section7")
g.s7 <- graph.data.frame(s7, directed=FALSE)
#g7 <- simplify(g.s7)
plot(g.s7)
```


```
V(g.s7)$degree <- degree(g.s7, mode="all")
V(g.s7)$betweeness <- betweenness(g.s7)
V(g.s7)$closeness <- closeness(g.s7)
n.s7 <- as.data.frame(V(g.s7)$name)
b.s7 <- as.data.frame(V(g.s7)$betweeness)
d.s7 <- as.data.frame(V(g.s7)$degree)
c.s7 <- as.data.frame(V(g.s7)$closeness)
s7_final <- cbind(n.s7,b.s7,d.s7,c.s7)
s7_final$Section <- 7
colnames(s7_final) <- c("Name","Betweenness","Degree","Closeness","Section")
remove(n.s7,b.s7,d.s7,c.s7)
```

```
s8 <- Relationship1 %>% filter(section=="Section8")
g.s8 <- graph.data.frame(s8, directed=FALSE)
#g8 <- simplify(g.s8)
plot(g.s8)
```

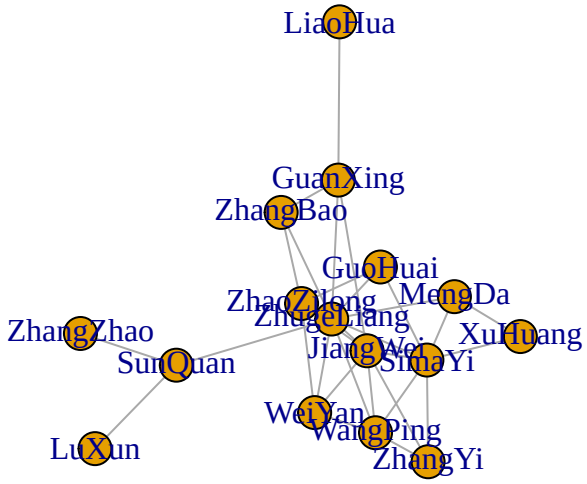



```
V(g.s8)$degree <- degree(g.s8, mode="all")
V(g.s8)$betweeness <- betweenness(g.s8)
V(g.s8)$closeness <- closeness(g.s8)
n.s8 <- as.data.frame(V(g.s8)$name)
b.s8 <- as.data.frame(V(g.s8)$betweeness)
d.s8 <- as.data.frame(V(g.s8)$degree)
c.s8 <- as.data.frame(V(g.s8)$closeness)
s8_final <- cbind(n.s8,b.s8,d.s8,c.s8)
s8_final$Section <- 8
colnames(s8_final) <- c("Name","Betweenness","Degree","Closeness","Section")
remove(n.s8,b.s8,d.s8,c.s8)
```

```
s9 <- Relationship1 %>% filter(section=="Section9")
g.s9 <- graph.data.frame(s9, directed=FALSE)
#g9 <- simplify(g.s9)
plot(g.s9)
```


```
V(g.s9)$degree <- degree(g.s9, mode="all")
V(g.s9)$betweeness <- betweenness(g.s9)
V(g.s9)$closeness <- closeness(g.s9)
n.s9 <- as.data.frame(V(g.s9)$name)
b.s9 <- as.data.frame(V(g.s9)$betweeness)
d.s9 <- as.data.frame(V(g.s9)$degree)
c.s9 <- as.data.frame(V(g.s9)$closeness)
s9_final <- cbind(n.s9,b.s9,d.s9,c.s9)
s9_final$Section <- 9
colnames(s9_final) <- c("Name","Betweenness","Degree","Closeness","Section")
remove(n.s9,b.s9,d.s9,c.s9)
```

```
s10 <- Relationship1 %>% filter(section=="Section10")
g.s10 <- graph.data.frame(s10, directed=FALSE)
#g10 <- simplify(g.s10)
plot(g.s10)
```

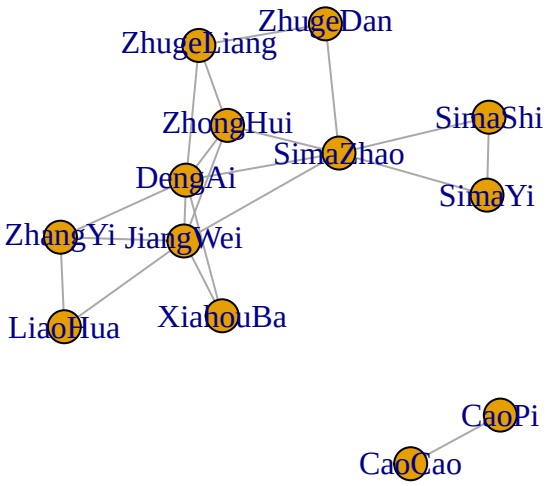



```
V(g.s10)$degree <- degree(g.s10, mode="all")
V(g.s10)$betweeness <- betweenness(g.s10)
V(g.s10)$closeness <- closeness(g.s10)
n.s10 <- as.data.frame(V(g.s10)$name)
b.s10 <- as.data.frame(V(g.s10)$betweeness)
d.s10 <- as.data.frame(V(g.s10)$degree)
c.s10 <- as.data.frame(V(g.s10)$closeness)
s10_final <- cbind(n.s10,b.s10,d.s10,c.s10)
s10_final$Section <- 10
colnames(s10_final) <- c("Name","Betweenness","Degree","Closeness","Section")
remove(n.s10,b.s10,d.s10,c.s10)
```

```
s11 <- Relationship1 %>% filter(section=="Section11")
g.s11 <- graph.data.frame(s11, directed=FALSE)
#g11 <- simplify(g.s11)
plot(g.s11)
```


```
V(g.s11)$degree <- degree(g.s11, mode="all")
V(g.s11)$betweeness <- betweenness(g.s11)
V(g.s11)$closeness <- closeness(g.s11)
n.s11 <- as.data.frame(V(g.s11)$name)
b.s11 <- as.data.frame(V(g.s11)$betweeness)
d.s11 <- as.data.frame(V(g.s11)$degree)
c.s11 <- as.data.frame(V(g.s11)$closeness)
s11_final <- cbind(n.s11,b.s11,d.s11,c.s11)
s11_final$Section <- 11
colnames(s11_final) <- c("Name","Betweenness","Degree","Closeness","Section")
remove(n.s11,b.s11,d.s11,c.s11)
```

```
s12 <- Relationship1 %>% filter(section=="Section12")
g.s12 <- graph.data.frame(s12, directed=FALSE)
#g12 <- simplify(g.s12)
plot(g.s12)
```



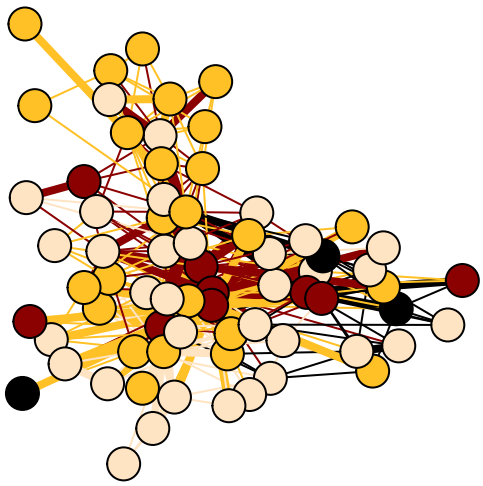

```
V(g.s12)$degree <- degree(g.s12, mode="all")
V(g.s12)$betweeness <- betweenness(g.s12)
V(g.s12)$closeness <- closeness(g.s12)
n.s12 <- as.data.frame(V(g.s12)$name)
b.s12 <- as.data.frame(V(g.s12)$betweeness)
d.s12 <- as.data.frame(V(g.s12)$degree)
c.s12 <- as.data.frame(V(g.s12)$closeness)
s12_final <- cbind(n.s12,b.s12,d.s12,c.s12)
s12_final$Section <- 12
colnames(s12_final) <- c("Name","Betweenness","Degree","Closeness","Section")
remove(n.s12,b.s12,d.s12,c.s12)
```

Combine all sections

```
#secname <- gsub(" ", "",c(paste(rep("s",12),seq(1,12),"_final",",")))
allsec <- rbind(s1_final, s2_final,s3_final,s4_final,s5_final,s6_final, s7_final,s8_final,s9_final,s10_
#write.csv(allsec, "character_info.csv",col.names = TRUE,row.names = FALSE)
```

```
all <- Relationship%>%
  group_by(Person1, Person2)%>%
  mutate(over_all_weight=sum(Weight))%>%
  select(-Weight,-Chapter,-section)%>%
  unique()%>%
```

```
  ungroup()


all <- left_join(all, characters, by= c("Person1"="Name"))

#Determine weak/strong relationship for all.
all$over_all_weight <- as.numeric(all$over_all_weight)
mean_weight <- mean(all$over_all_weight)
all$strong <- ifelse(all$over_all_weight>mean_weight, 1,0)

#Graph Settings
node_color <- all
color = c("bisque","darkred","goldenrod1","black")[as.factor(node_color$Country)]
all$color <-  color
#Shu=darkred, other= bisque, black= Wu, goldenrod1= Wei

#set width for weak and strong
all_width <- all
width <-  c("1","4")[as.factor(all_width$strong)]
all$width <-  width

net_all <- graph.data.frame(all, directed=FALSE)
plot.igraph(net_all,vertex.color=color, frame.color=color,vertex.label=NA,edge.arrow.size=0.3, edge.arr
```



```r
core <- eigen_centrality(net_all,directed=FALSE, weights=all$over_all_weight)

core <- sort(core$vector, decreasing =TRUE )
```

```r
#find adjacency matrix
all1 <- all%>%
  select(Person1, Person2)%>%
  unique()
net_all1 <- graph.data.frame(all1, directed=FALSE)
adj <- as_adjacency_matrix(net_all1)
blah <- as.data.frame(adj[1,])


adjdecomp <- eigen(adj)

correlation <- c()
for (n in 1:70){
  c_star <- c(rep(1,n),rep(0,(70-n)))
  correlation[n] <- cor(core,c_star, method = "pearson")
}
```

```
## Warning in cor(core, c_star, method = "pearson"): the standard deviation is zero
```

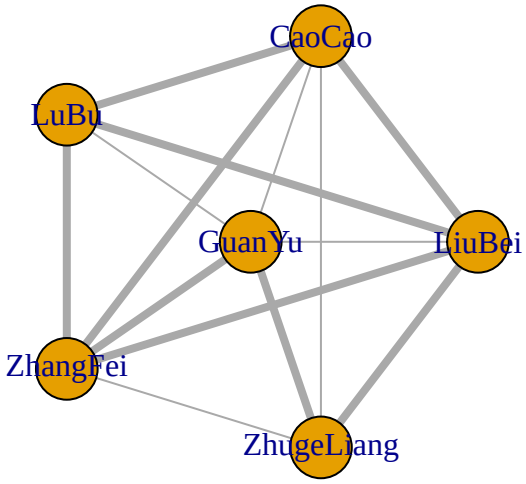
```r
correlation <- as.data.frame(correlation)
correlation <- correlation%>%filter(!is.na(correlation))
which.max(abs(correlation$correlation))
```

```
## [1] 6
```

```r
## it is corepheriphery and 6 people are in the core
core[1:6]
```

```
##     LiuBei     CaoCao ZhugeLiang     GuanYu   ZhangFei       LuBu
##  1.0000000  0.8907272  0.5791374  0.5594145  0.4244504  0.3822674
```

```r
core_people <- c("LiuBei", "CaoCao", "ZhugeLiang", "GuanYu","ZhangFei", "LuBu")

all2 <- all1%>%
  filter(Person1%in%core_people)%>%
  filter(Person2%in%core_people)

all1$is_core <- ifelse(all1$Person1%in%core_people,1,0)
node_color <- all1
color = c("bisque","darkred")[as.factor(node_color$is_core)]
all1$color <-  color

net_all2 <- graph.data.frame(all2, directed=FALSE)
plot.igraph(net_all2,
            #vertex.color="darkred",
            vertex.size= 30,
            frame.color=color,edge.arrow.size=0.3, edge.arrow.width =0.5, edge.curved=F,edge.size=20, e
```


CaoCao
LuBu
GuanYu
LiuBei
ZhangFei
ZhugeLiang