

## Annex 5

### Data extraction from newspaper's web page and comparison

On the whole set of articles, a total of 145, only for five of them has been possible to write comments straight on the article's web page. The same articles have also been available for comments at Facebook and Twitter. The next table lists those articles commented in the same web page.

Internal id	date	media	n. of comments
29	22/07/18	Independent	76
68	23/07/18	Daily Mail	41
69	22/07/18	Daily Mail	452
107	23/07/18	Daily Express	33
108	23/07/18	Daily Express	20

As shown in the table three newspapers displayed comments in their web pages, each newspaper offered apparently similar commenting systems, but in reality they use different mechanisms, it is interesting to analyse them separately.

Newspaper "Independent" shows user names with bold characters, "Reply" link with red colour, both capture the attention. Two icons "thumb up" and "thumb down" with the number of likes in the middle. The number of likes is the algebraic sum of positive and negative score.

Newspaper "Daily Mail" shows the number of likes and the number of dislikes separately.

Newspaper "Daily Express" calculates the number of likes as algebraic sum of positive and negative score, moreover allows to share or report the comment. Some users make available their avatar picture.

An important finding common to the three newspapers web site commenting system is related to anonymity, users account do not really disclose personal information, users identity cannot be validated by any mean, even for those few users with real faces as avatars. Moreover the censorship activity removed some avatars and comments.

The three newspapers do not make available Application Program Interfaces (APIs), a third part service, newsapi.org provides APIs that get news from Independent and Daily Mail. There are no APIs able to read comments.

In order to extract the network diagram and the set of comments from newspapers web sites it has been necessary to develop specific software programs that implement web scraping techniques.

The first step analyses the Hypertext Markup Language (HTML) code behind the web pages for searching patterns that allow to capture data. The syntax used is XPath, it allows to find elements in the HTML code.

## Independent newspaper web comments (internal id 29)

The next picture shows the comment section, user names are not obfuscated because they do not disclose real identities.

---

 **rogerthecat** 1 year ago

Let's wait and see what transpires instead of making assumptions about whether this was an attack by people some people don't like.

Reply

 2  1 

---

 **[removed]**


1 year ago

This comment has been deleted

Reply

 0  0 

---

 **bobi6192**

1 year ago

How paranoid and mentally unstable do you have to be to convince yourself that there is some nationwide conspiracy to promote the liberal agenda. Whatever the expression "liberal agenda" actually means. Seems to be a catch all term for anything right wing loons don't like.

Reply

 2  1 

The elements to capture are the user name, the comment text and the position to determine whether is a direct comment or a reply to an existing comment.

The analysis of HTML code resulted in the following patterns:

- user name is coded as span element with attribute class = "user"
- comments are coded as div element with attribute class = "comment-text"
- elements "data-comment-id" and "class" indicate the level of reply

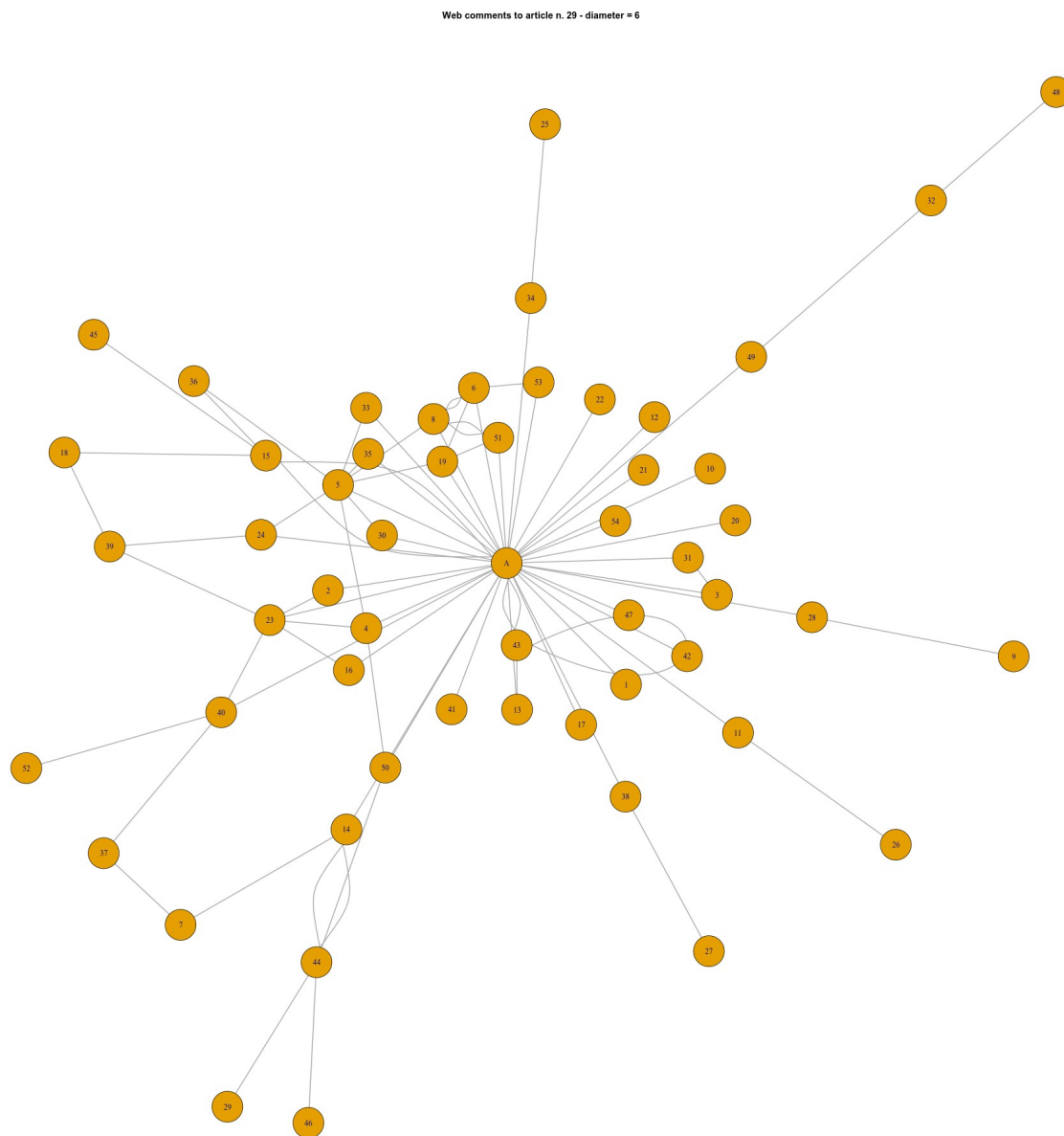
The next illustration shows the code in R programming language for extracting data.

```

library(rvest)
library(dplyr)
library(grr)
library(igraph)
html_doc <- read_html(paste0(dir_tests,art_num,'.html'), encoding = 'UTF-8')
node_comments <- html_nodes(html_doc, xpath = "//div[@class='comment card']")
len_df = length(node_comments)
df <- data.frame(
  'un' = character(len_df), # user name
  'dd' = character(len_df), # data-comment-id
  'cn' = character(len_df), # class name
  'nl' = integer(len_df), # number of likes
  'nd' = integer(len_df), # number of dislikes
  'cm' = character(len_df), # comments text
  stringsAsFactors=FALSE
)
for (ind in 1:len_df) {
  df[ind,2] <- html_attr(html_nodes(node_comments[ind], xpath = "."),'data-comment-
id')
  df[ind,3] <- html_attr(html_nodes(node_comments[ind], xpath = "..'),'class')
}
df$un <- html_text(html_nodes(node_comments, xpath = "//span[@class='user']"))
df$nl <- as.integer(html_text(html_nodes(node_comments, xpath = "//span[@class='vote-
count'] [1]")))
df$nd <- as.integer(html_text(html_nodes(node_comments, xpath = "//span[@class='vote-
count'] [2]")))
df$cm <- html_text(html_nodes(node_comments, xpath =
"//div[@class='comment-text']/text()"))
re <- 1
for (ind in 1:len_df) {
  if (df[ind,1] == '[removed]') {
    df[ind,1] <- paste0('removed_',re)
    re <- re + 1
  }
}
df[is.na(df)] <- ''
df$rp <- ''
for (ind in 1:len_df) { if ( ! df[ind,2] == '' ) { df[ind,7] <- newspaper } }
for (ind in 1:len_df) {
  if ( df[ind,7] == '' & df[ind,3] == 'replies-1' ) {
    un <- df[ind - 1,1]
    wnd <- ind
    while ( df[wnd,7] == '' ) {
      if ( df[wnd,3] == 'replies-1' ) { df[wnd,7] <- un }
      wnd <- wnd + 1
    }
    ind <- wnd
  }
}
for (ind in 1:len_df) {
  if ( df[ind,7] == '' & df[ind,3] == 'replies-2' ) {
    un <- df[ind - 1,1]
    wnd <- ind
    while ( df[wnd,7] == '' ) {
      if ( df[wnd,3] == 'replies-2' ) { df[wnd,7] <- un }
      wnd <- wnd + 1
    }
    ind <- wnd
  }
}
fqun <- as.data.frame(table(df$un)) # get unique records of user names and calculates
frequencies
fqun <- mutate(fqun, id = rownames(fqun)) # adds column id and populates it
colnames(fqun)[1] <- 'name'
df$un <- as.factor(df$un)
dict <- grr::matches(fqun$name, df$un)
dict_sorted <- dict[order(dict[,2]),]
df <- cbind(df,dict_sorted[,1])
colnames(df)[8] <- 'id_un'
df$id_rp <- match(df$rp, fqun$name, 0)
df$id_rp[df$id_rp==0] <- 'A'
ph <- vector('character');
len <- nrow(df)
for (row in 1:len){
  ph <- c(ph,c(df[row,9],df[row,8]))
}
gr <- graph(ph,directed = FALSE)

```

The next illustration shows the network diagram



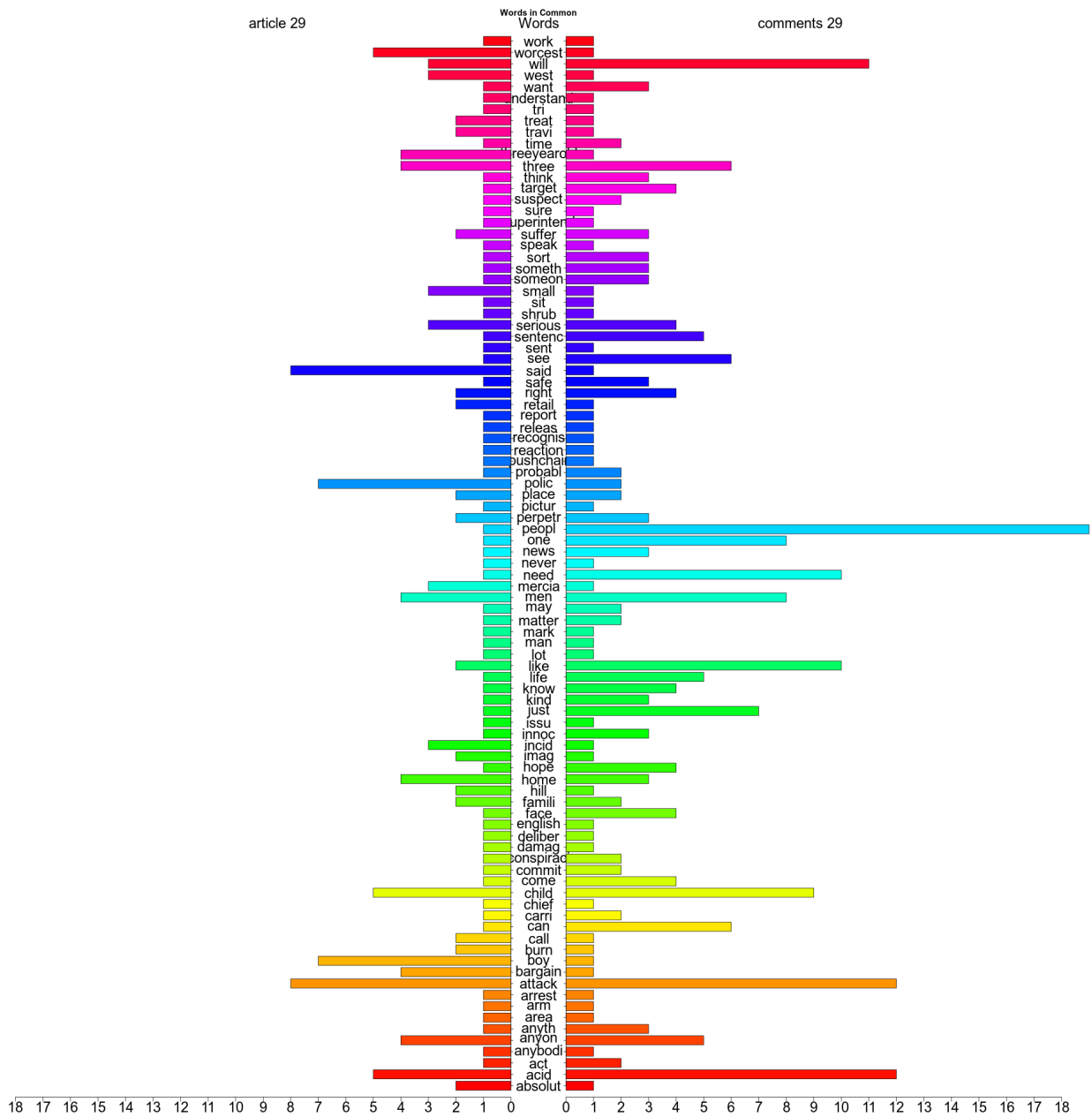
The central node is the article's author. All names are replaced by numbers. The network diameter is 6.

Now proceed with the comparison between the article's text and the set of comments.

The next illustrations show the word cloud representations of the article and the comments with a minimal frequency of two words and the pyramid diagram of common words.

police acid home  
child inform  
three investig sunday  
concern incid serious  
boy thing close shop  
ambul urg west  
hospit absolut hill contact  
small burnyoung remai will local  
suffer injur perpetr men said  
bargain longterm shock  
travi imag attack anyone  
likeplace treat  
retail

comme person street allow  
vat hate sick delet minor commit  
serv way pay sentenc flag death  
given done crime good getanoth  
jail eye drop murder catch honest carri  
upon think acid sort target joke  
old aliv kind throw human must promot  
polic men hope come post utter let  
arentvictim larg ive crimin pain life sale even warn  
walk innoc total automat paranoid perpetr agenda drive  
news care convict read tab want home dealtweb  
cours defend assumpt countri agre three  
long liber probabl one keep usual face



The R programming code is the same shown in annex 4 titled “Comparison between article’s text and comments”

Jaccard similarity value calculated is 0.33

## Daily Mail newspaper web comments (internal id 68)

The next picture shows the comment section, user names are not obfuscated because they do not disclose real identities.

The screenshot displays the 'Comments 41' section of a Daily Mail article. At the top, there is a blue speech bubble icon and the text 'Comments 41 Share what you think'. Below this are four tabs: 'Newest', 'Oldest', 'Best rated', and 'Worst rated'. A 'View newest 10' button is also present. A notice states: 'The comments below have been moderated in advance.' The comments are as follows:

- Big Denz**, The Woo, United Kingdom, 12 months ago  
I have zero confidence in our justice system, that if found guilty, the perpetrators with get a sentence that reflects the crime. Even if they get 20 years, they'll be out in 8. It's not justice anymore, it the appearance of justice.  
Click to rate: 60 upvotes, 0 downvotes
- MrsN**, essex, United Kingdom, 12 months ago  
My son had an accident with a boiled kettle whilst in someone else's care. The screams when I got the call and then holding him in hospital was the worst sound I have ever heard and my goodness he sobbed in agony. That poor child - my heart goes out to the child and mother. Must have been incredibly distressing. Speeding recovery sweetheart  
Click to rate: 77 upvotes, 1 downvote
- Leximarie1**, Nottingham , United Kingdom, 12 months ago  
I had something similar when my two year old son pulled an instant cappachino down over his arm. the screams and the pain are horrific. why someone would deliberately administer this sort of pain to an innocent 3 year old I will never comprehend. it's time the government and police gave stiffer penalties in cases like this. unfortunately this is the UK and our government imply not have the balls!  
Click to rate: 20 upvotes, 0 downvotes
- Truthful 21**, Aegean sea, Greece, 12 months ago  
I've cried and cried over this -it's shocked me to the core. That poor poor family- it's one of those occurrences that once you have heard about you can never ever forget!  
Click to rate: 14 upvotes, 0 downvotes

The elements to capture are the user name, the comment text and the position to determine whether is a direct comment or a reply to an existing comment.

The analysis of HTML code resulted in the following patterns:

- user name is coded as p element with attribute class = "user-info"
- comments are coded as p element with attribute class starting with "comment" or "reply"
- elements after elements p with attribute class = "user-info" indicate the level of reply

The next illustration shows the code in R programming language for extracting data.

```

library(rvest)
library(dplyr)
library(grr)
library(igraph)

html_doc <- read_html(paste0(dir_tests,art_num,'_comments.html'), encoding = 'UTF-8')
node_comments <- html_nodes(html_doc, xpath = "//div[starts-with(@id,'comment-')]")
len_df = length(node_comments)

# user names
un <- html_text(html_nodes(node_comments,xpath="//p[@class='user-info']/a/text()"))
# comments text
cm <- html_text(html_nodes(node_comments,xpath="//p[starts-with(@class,'comment') or
starts-with(@class,'reply')]/text()"))
# number of likes
nl <- as.integer(html_text(html_nodes(node_comments,xpath="//div[@class='rate-up']/
following-sibling::div[@class='rating-button-inline']/text()")))
# number of dislikes
nd <- as.integer(html_text(html_nodes(node_comments,xpath="//div[@class='rate-down']/
following-sibling::div[@class='rating-button-inline']/text()")))
# type of comment
tc <- html_text(html_nodes(node_comments,xpath="//p[@class='user-info']/following-
sibling::p[@class="])
tc <- gsub('.*comment.*','comment',tc)
tc <- gsub('.*reply.*','reply',tc)

df <- data.frame(
  'un' = character(len_df),
  'nl' = integer(len_df),
  'nd' = integer(len_df),
  'cm' = character(len_df),
  'tc' = character(len_df),
  stringsAsFactors=FALSE
)

df$un <- un
df$nl <- nl
df$nd <- nd
df$cm <- cm
df$tc <- tc

fqun <- as.data.frame(table(df$un)) # get unique records of user names and
calculates frequencies
fqun <- mutate(fqun, id = rownames(fqun)) # adds column id and populates it
colnames(fqun)[1] <- 'name'
df$un <- as.factor(df$un) # otherwise grr::matches fires error
dict <- grr::matches(fqun$name, df$un)
dict_sorted <- dict[order(dict[,2]),]
df <- cbind(df,dict_sorted[,1])
colnames(df)[6] <- 'id_un'

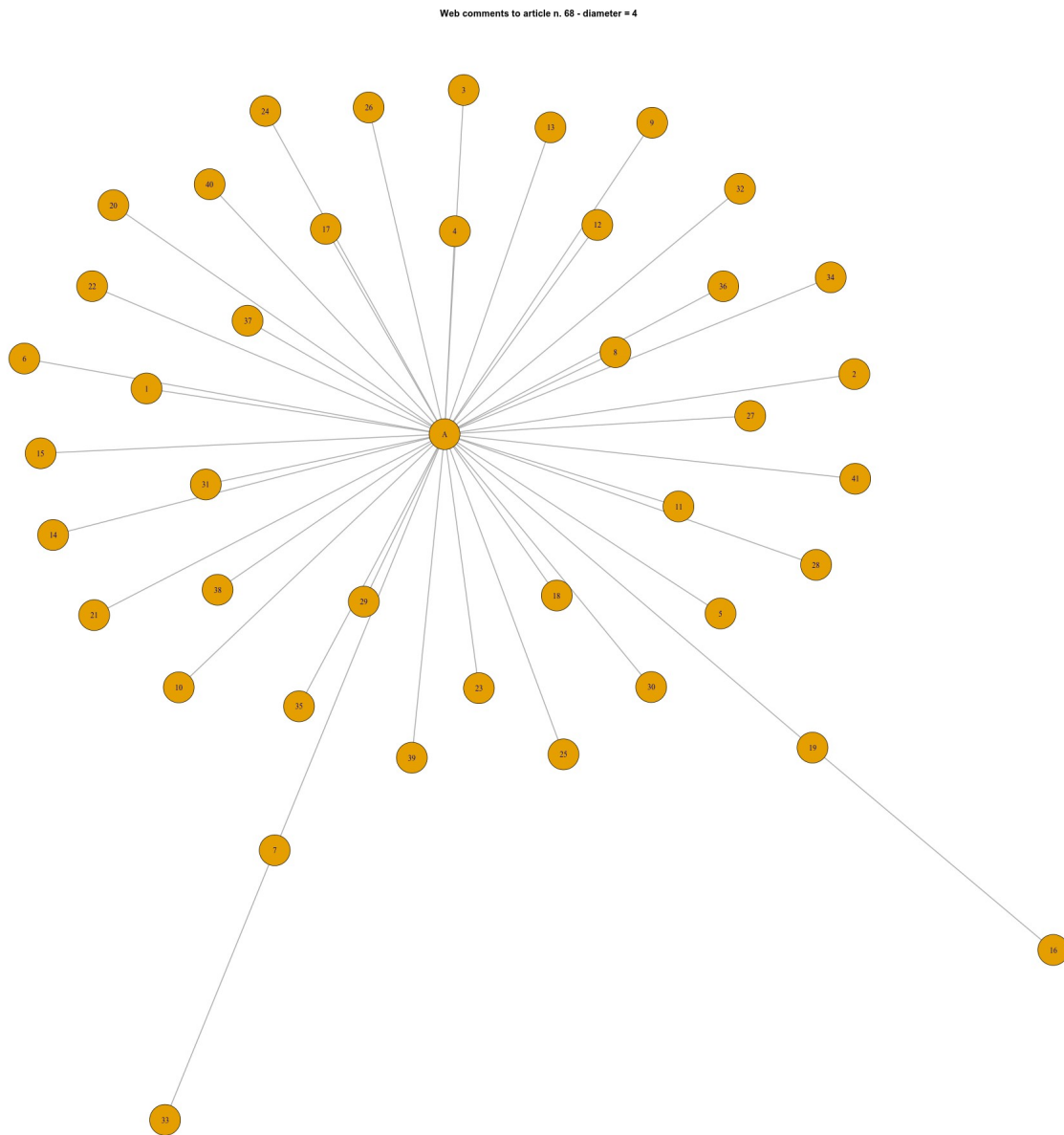
# block extract comments -----
fcon <- file(paste0(dir_tests,art_num,'_comments.txt'))
writeLines(toString(df$cm), fcon)
close(fcon)

# block net graph -----
ph <- vector('character');
len <- nrow(df)
for (row in 1:len){
  if (df[row,5] == 'comment') { ph <- c(ph,c('A',df[row,6])) }
  if (df[row,5] == 'reply') {
    uid <- df[row-1,6]
    while (df[row,5] == 'reply') {
      ph <- c(ph,c(uid,df[row,6]))
      row <- row + 1
    }
  }
}
gr <- graph(ph,directed = FALSE)

```



The next illustration shows the network diagram



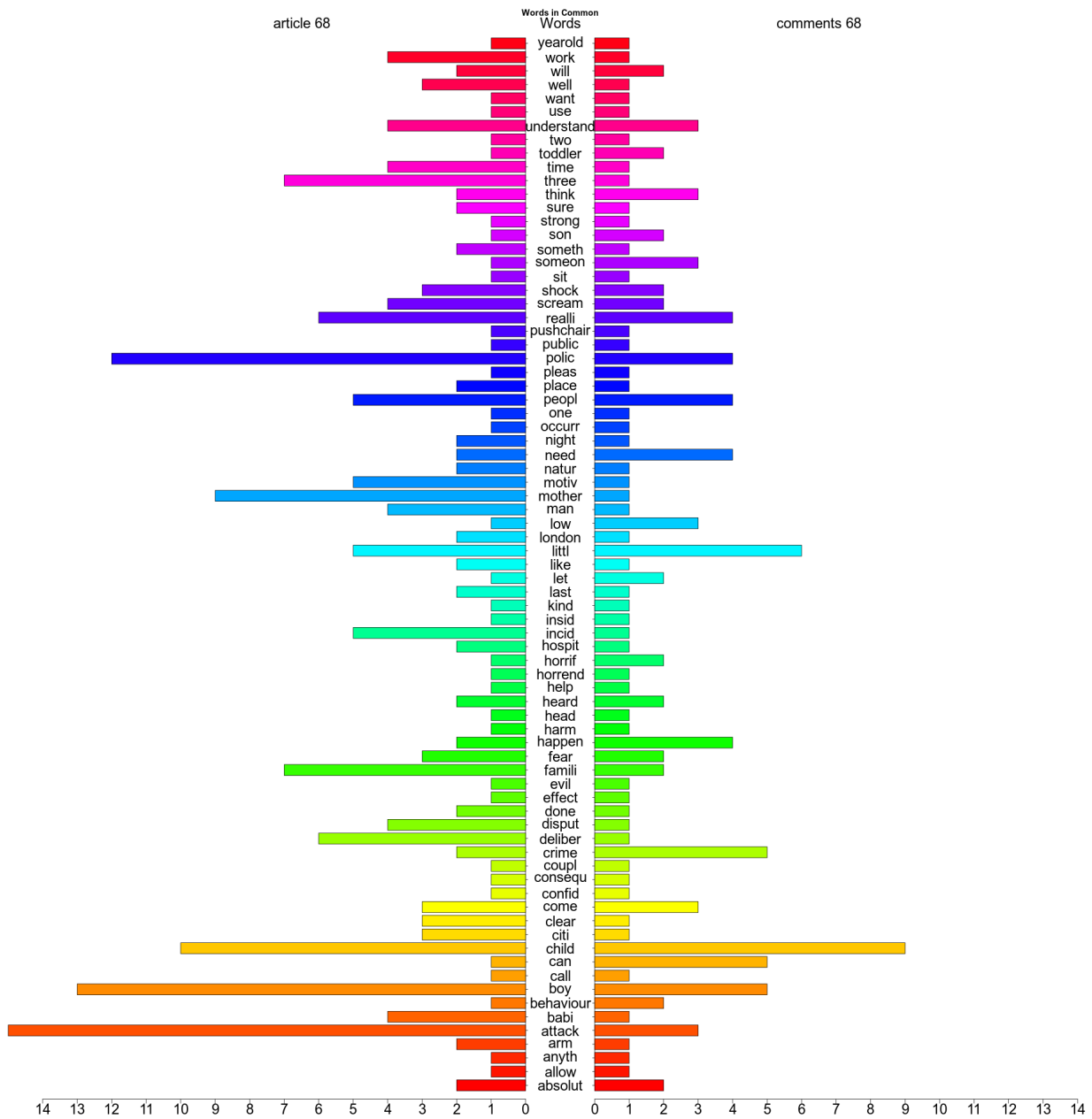
The central node is the article's author. All names are replaced by numbers. The network diameter is 4.

Now proceed with the comparison between the article's text and the set of comments.

The next illustrations show the word cloud representations of the article and the comments with a minimal frequency of two words and the pyramid diagram of common words.

west understand investig describ seen  
quit toward scream boy detect home  
well sure captur commit know  
someth citi injuri area around  
move told shop bargain like park  
heard juli away absolut shock  
ambul will time acid men threeyearold believ  
arm part travimen mercia fear need target mother  
burn hospitbst motivimag suspicion  
link show appeal markput crime 2018 suspiation  
safeti treat look done smell outsid taken  
face deliber store littl communiti  
babi london incid still three way  
saturday attack local bodili grievous  
make safe man substanc hous clear chief last longterm famili  
natur term night remain howev

never  
scream societi  
shock fear remiss innoc  
anybodi system attack  
must let ever lowest sentenc  
britain come famili justic can littl  
govern pain cri someone just thing  
low govern fail lawless hope child just stop  
toddertake wordpolic happen just kid  
world absolut anyon goe happen just ive  
horrif life peopl sick lot pull heard think son perpetr need  
beyondheart guilti sort will heard comprehend  
realli understand inexcus  
even old boy get  
poor crime



The R programming code is the same shown in annex 4 titled “Comparison between article’s text and comments”

Jaccard similarity value calculated is 0.32

## Daily Mail newspaper web comments (internal id 69)

The next picture shows the comment section, user names are not obfuscated because they do not disclose real identities.



**Comments 452**  
Share what you think

Navigation: Newest | **Oldest** | Best rated | Worst rated  
View oldest 10

Page 3 of 3  
Previous 1 2 3

The comments below have been moderated in advance.

**Mardyman**, Birmingham, United Kingdom, 1 year ago  
I am so upset I am lost for words this government is the worst we have ever had WEAK!!  
Click to rate 254 13

**MeIK**, Yorkshire, United Kingdom, 1 year ago  
The only people to blame are the vile cowards who threw the acid. It was their choice to buy it, their choice to target an innocent child and their choice to carry out the attack.  
Click to rate 77 8

**Islay Johnson**, Barking, United Kingdom, 1 year ago  
our government has been weak for 20 years now, just been EU puppets  
Click to rate 9 1

**Rmt8**, Merseyside, United Kingdom, 1 year ago  
Utter cowards this is beyond vile!!!  
Click to rate 239 4

**Thanks for reading**, Youknowwherewelve, United Kingdom, 1 year ago  
Agreed. And if we were really committed to stop this, we would vote out the usual political parties who allow this to happen. But most people are too scared to do that.  
Click to rate 81 1

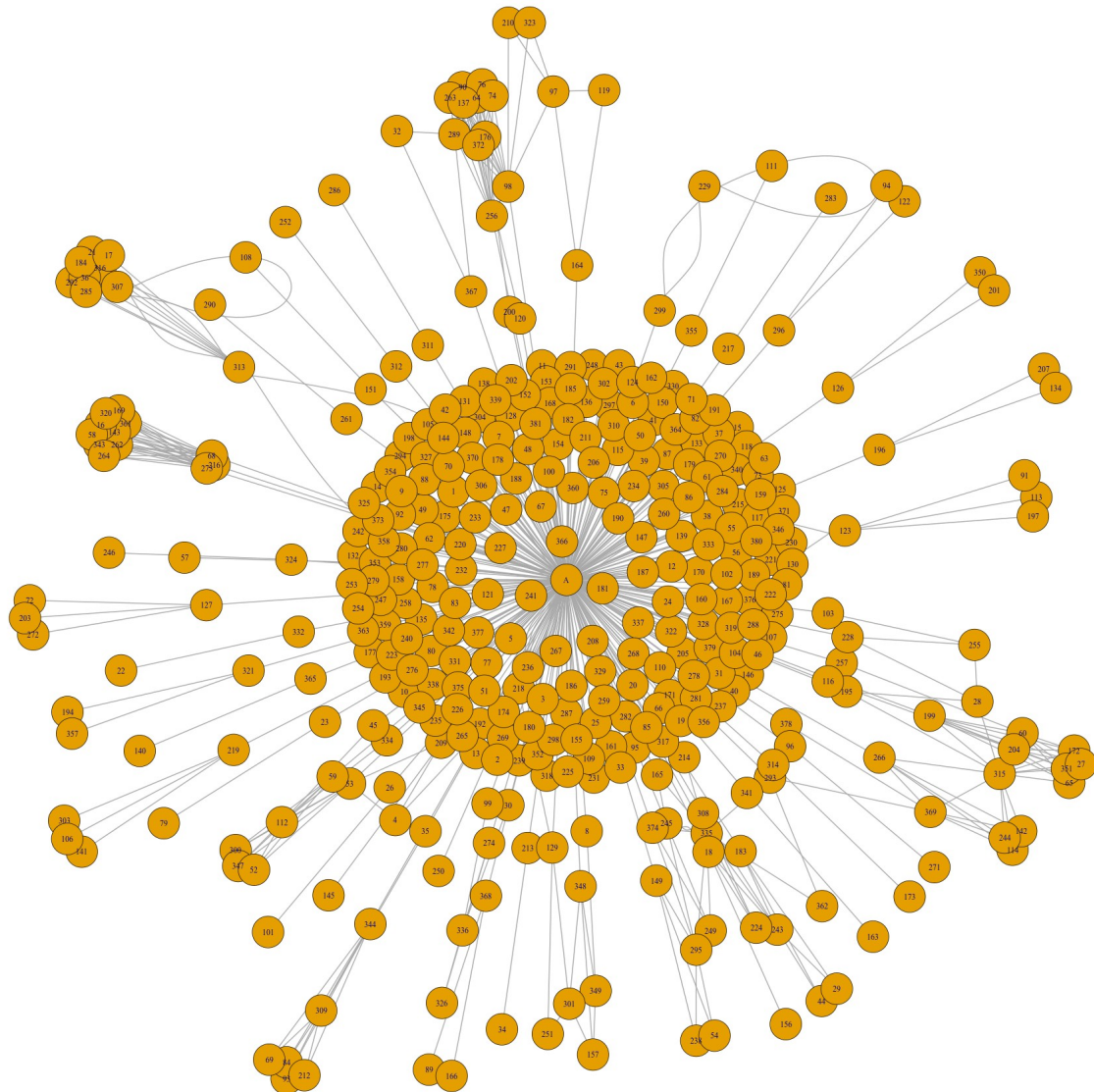
**Wassamatta**, Sunderland, United Kingdom, 1 year ago  
Thanks for Reading: Tell me which political party that is capable of running the country will do something about this.  
Click to rate 4 0

**Phillyco07**, Belfast, United Kingdom, 1 year ago  
I actually have no words... What is this world coming to. That poor baby.. I hope he recovers well.  
Click to rate 316 5

The analysis of HTML code and R programming code are the same as previous web comments (internal id 68)

The next illustration shows the network diagram

Web comments to article n. 69 - diameter = 4



The central node is the article's author. All names are replaced by numbers. The network diameter is 4.

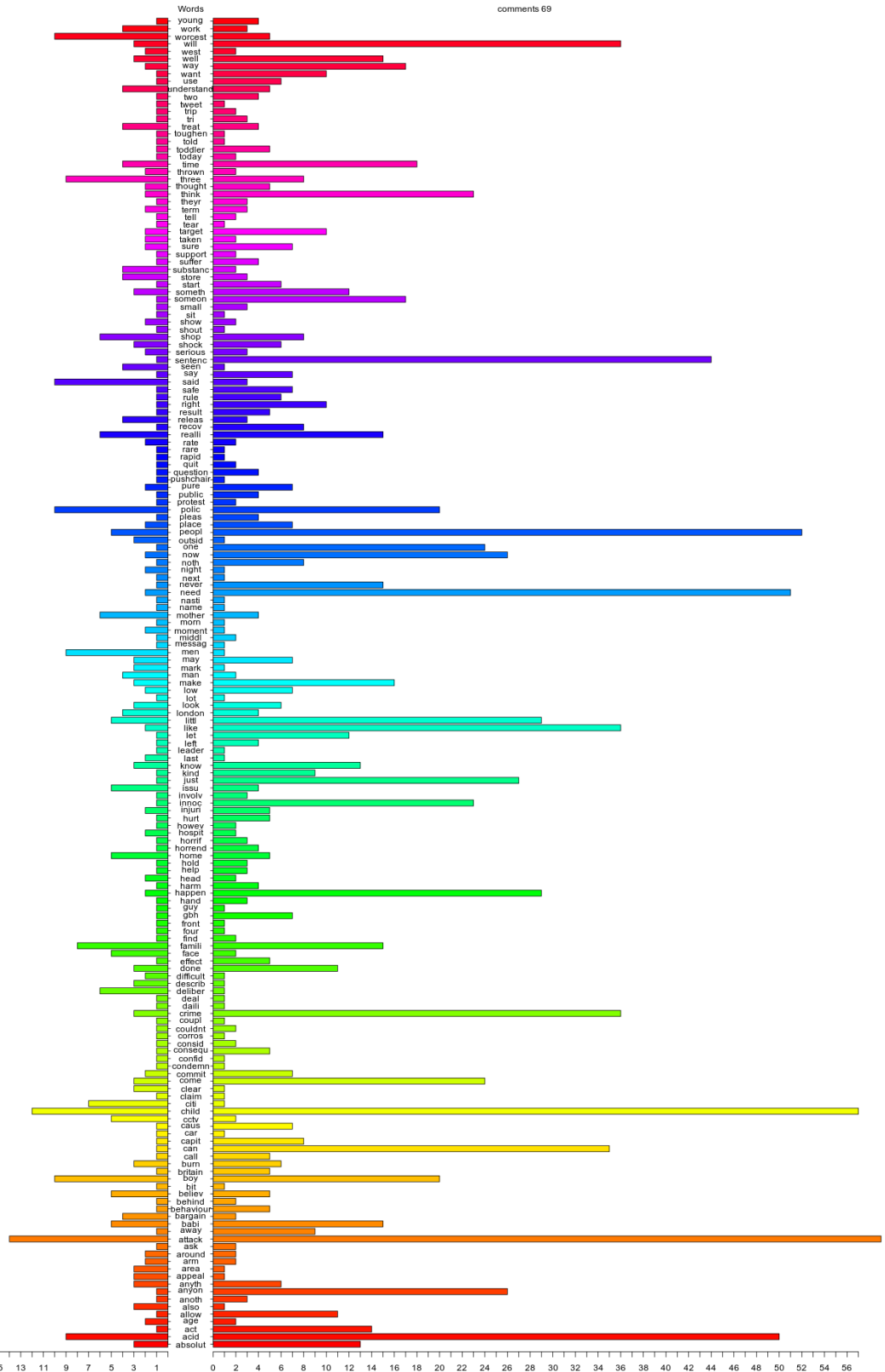
Now proceed with the comparison between the article's text and the set of comments.

The next illustrations show the word cloud representations of the article and the comments with a minimal frequency of two words and the pyramid diagram of common words.

member short scream known  
believ **worcest** come  
around night remain taken area  
outsid work **said** longterm  
bst know need pure wit anything thug heard  
took age cctv shock grievous anyth thug show  
think now investig safest incid well juli toward natur  
commit local chief **deliber** likesure ran sp low hurl  
appeal makewolverhampton difficult move disput  
hospit **realli** moment thought releas will travi  
captur way arm **child** issu littl term  
39yearold **child** also retail link last  
thrown place imag serious suspicion head rate  
bodili time park **babi** hill  
west **arrest men** footag  
absolut threeyearold look man happen  
yesterday burn superintend

unthink hospitbelief offenc shame judg vehicl sick consid realiti  
andrea dna **will** love  
mind free avail brought mess reloc  
speedi deterr **enough can** give lawless wish  
charg tax fail **need** doesnt arent risk  
home place **guilti** agre core root worri  
save youv drug run insid **caught** comment fit lock word **babi**  
that shouldnt intent **time** lock want full put  
ask pay court **evil** media establish prevent yet leadership sea heart  
receiv today **allow** assault uninhabit due lesser injuri room  
legal choic probabl **caus** visitor public weak west britain continu **sever**  
lad anoth seem what **carri** real tritougher parti feel fear job  
quit play E alight rope **carri** cover left still kept add asap protest  
treat **right** bloodi idiot grip same involv cribest  
mrs massiv **sort** send agreement ever support crisi log by culprit toni  
small found harm **kind** far recover labour quick lowest sake beat  
hous behaviour girl **kind** obvious eye **boy** **crime**  
bless **shock** expens mother god mani justic **boy** **crime**  
step **person** sign physic reinstat lenient work without  
crazi horrend lack wont trip **human** liber lenient work without  
middl face **way** blame think low nation old screen trump  
around unabl **now** minimum stay target vile of current 100 control  
much **now** respons **world** jail **crime** **crime**  
possibl **throw** **world** jail **crime** **crime**  
set **bad** protect **world** jail **crime** **crime**  
in human stuff

### Words in Common

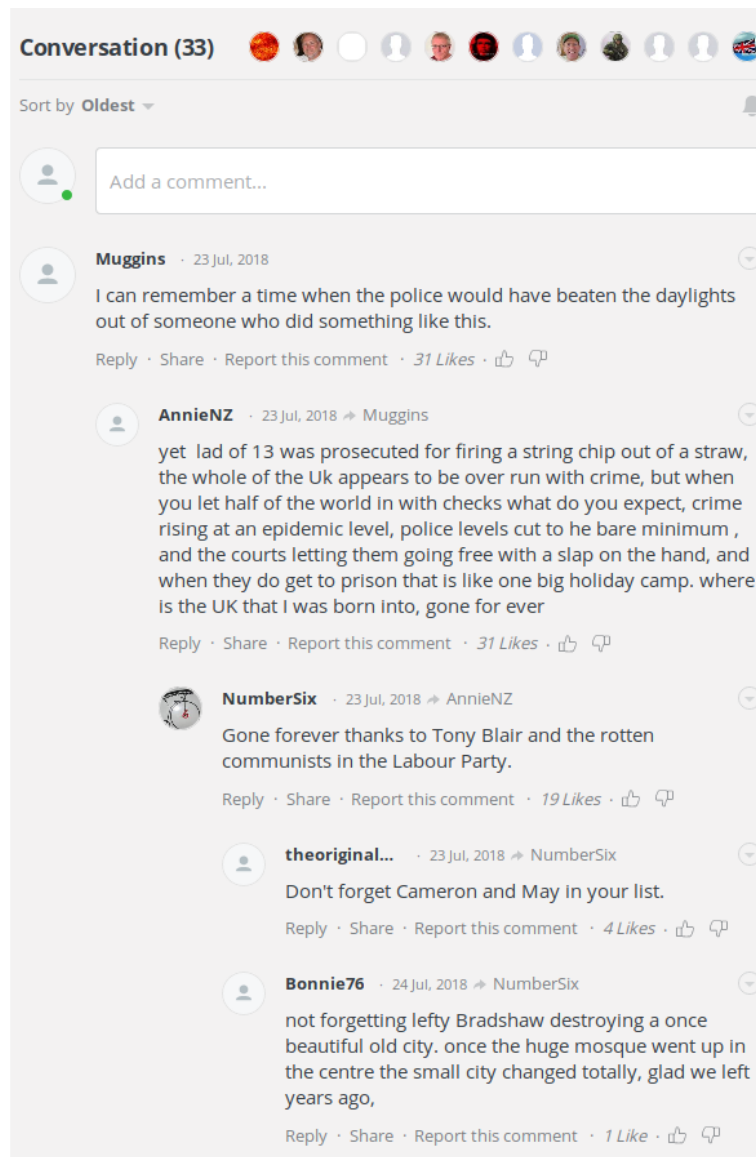


The R programming code is the same shown in annex 4 titled "Comparison between article's text and comments"

Jaccard similarity value calculated is 0.36

## Daily Express newspaper web comments (internal id 107)

The next picture shows the comment section, user names are not obfuscated because they do not disclose real identities.



The elements to capture are the user name, the comment text and the position to determine whether is a direct comment or a reply to an existing comment.

The analysis of HTML code resulted in the following patterns:

- user name is coded as span element with attribute data-spot-im-class = "message-username"
- comments are coded as div element with attribute data-spot-im-class = "message-text"
- the level of reply is coded in a extremely complicated way, since the number of comments is low therefore has been more convenient to fill this information manually based on a built table

The next illustration shows the code in R programming language for extracting data.



```

library(rvest)

html_doc <- read_html(paste0(dir_tests,art_num,'_comments.html'), encoding = 'UTF-8')
un <- html_text(html_nodes(html_doc, xpath = "//span[@data-spot-im-class='message-username']/text()"))
cm <- html_text(html_nodes(html_doc, xpath = "//div[@data-spot-im-class='message-text']/text()"))

len_df = length(un)

df <- data.frame(
  # user name
  'un' = character(len_df),
  # comments text
  'cm' = character(len_df),
  stringsAsFactors=FALSE
)

df$un <- un
df$cm <- cm

write.csv(df,paste0(dir_tests,art_num,'_comments.csv'))

# then manual fill

# block extract comments -----
fcon <- file(paste0(dir_tests,art_num,'_comments.txt'))
writeLines(toString(df$cm), fcon)
close(fcon)

# block net graph -----

library(dplyr)
library(grr)

dir_tests <- '/home/oreste/Downloads/'
art_num <- 107
newspaper <- 'Express'

df <- read.csv(paste0(dir_tests,art_num,'_graph.csv')) # filled manually

# un = user name
# cm = comments text
# nl = number of likes
# rp = replies to (character)
# le = level of reply (1 to 4)

fqun <- as.data.frame(table(df$un)) # get unique records of user names and
calculates frequencies
fqun <- mutate(fqun, id = rownames(fqun)) # adds column id and populates it
colnames(fqun)[1] <- 'name'
df$un <- as.factor(df$un) # otherwise grr::matches fires error
dict <- grr::matches(fqun$name, df$un)
dict_sorted <- dict[order(dict[,2]),]
df <- cbind(df,dict_sorted[,1])
colnames(df)[5] <- 'id_un'

df$id_rp <- match(df$rp,fqun$name, 0)

write.csv(df,paste0(dir_tests,art_num,'_igraph.csv'))

#-----
library(igraph)

dir_tests <- '/home/oreste/Downloads/'
art_num <- 107
newspaper <- 'Express'

df <- read.csv(paste0(dir_tests,art_num,'_igraph.csv'))
df[df==0]<-'A'

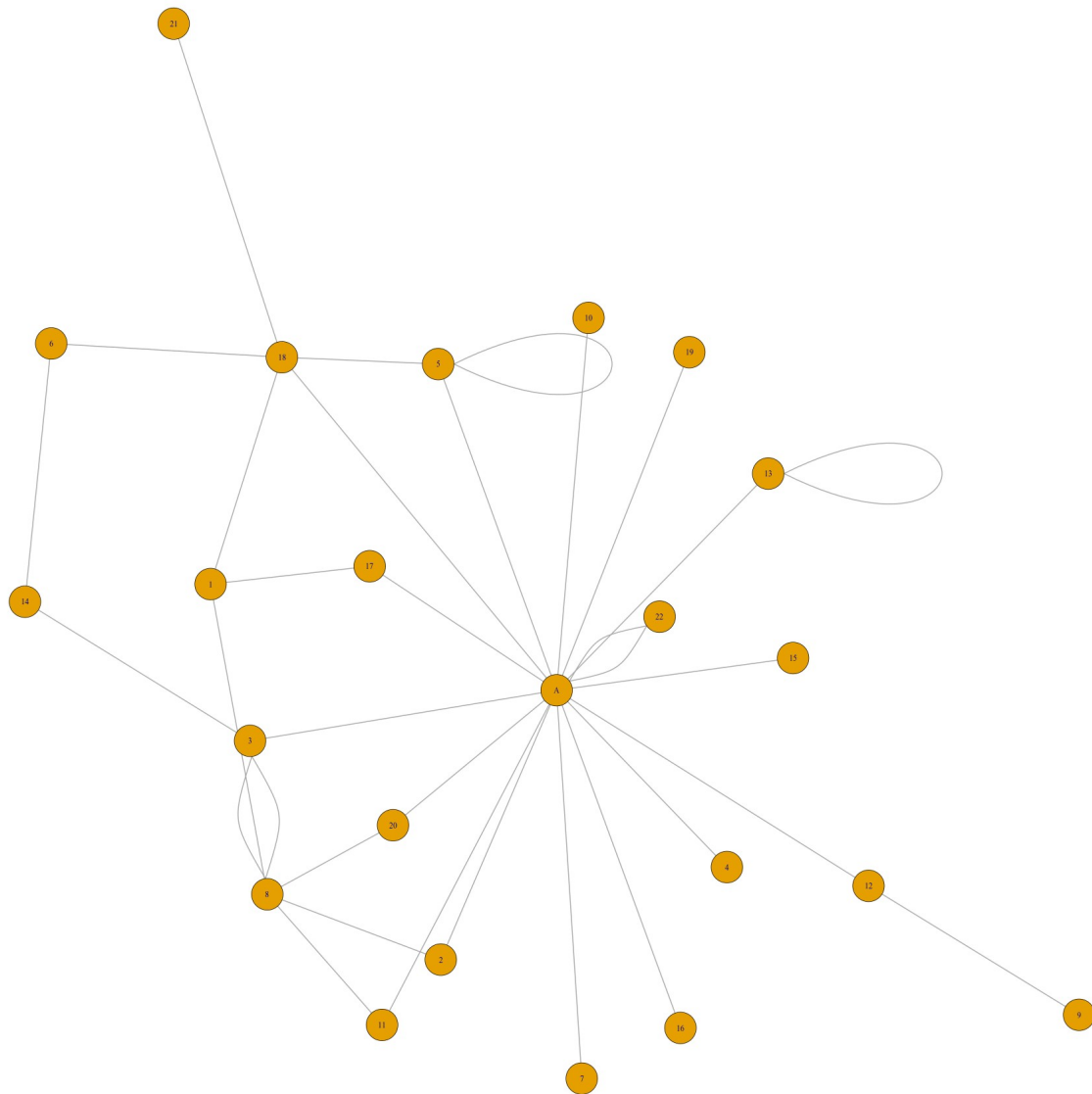
ph <- vector('character');
len <- nrow(df)
for (row in 1:len){ ph <- c(ph,c(df[row,7],df[row,6])) }

gr <- graph(ph,directed = FALSE)

```

The next illustration shows the network diagram

Web comments to article n. 107 - diameter = 4



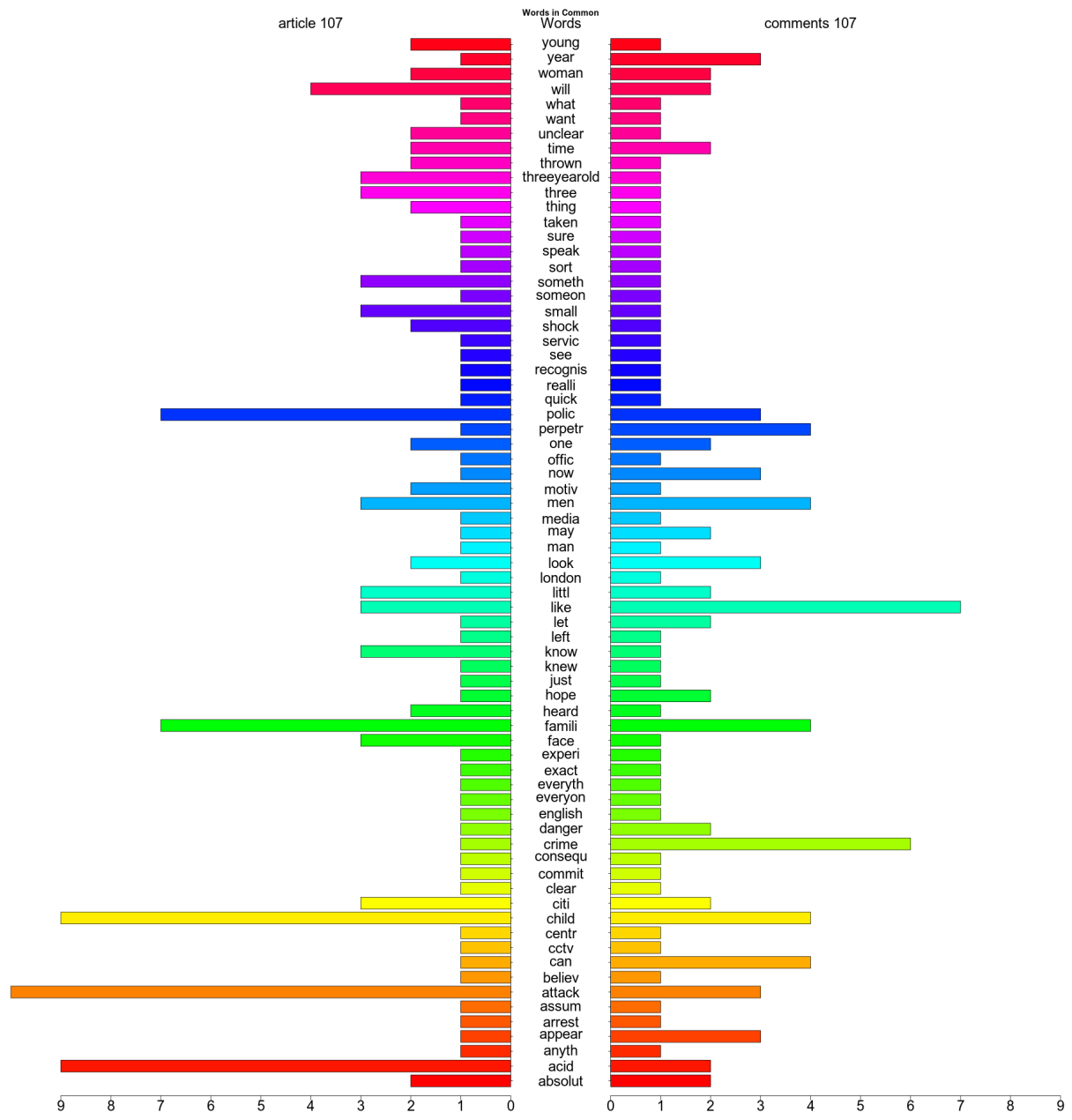
The central node is the article's author. All names are replaced by numbers. The network diameter is 4.

Now proceed with the comparison between the article's text and the set of comments.

The next illustrations show the word cloud representations of the article and the comments with a minimal frequency of two words and the pyramid diagram of common words.

thing shop famili deliber  
acid threeyearold think mon  
evil happen motiv  
thrown know said jul facemiddl form  
suspect investig still to time staffmen  
pure :iti remain look youngest will  
shock assist littl horrend store sentenc call  
work conserv boy fine local hill use three move  
woman incid polic ravi burn three  
small rachel area treat  
home council saturday one mum  
young worcest hospitalh last  
manag bargain receiv like arm  
absolut inform heard

person anoth tan allow dont  
read like year  
white mart huge free  
now can life charg  
citi littl prison inch letacid destroy attack yet  
1st glad report differ  
syrian prosecut countri old  
levelgood appear open will  
run may murder danger tri  
woman mention press time  
hope ever court famili

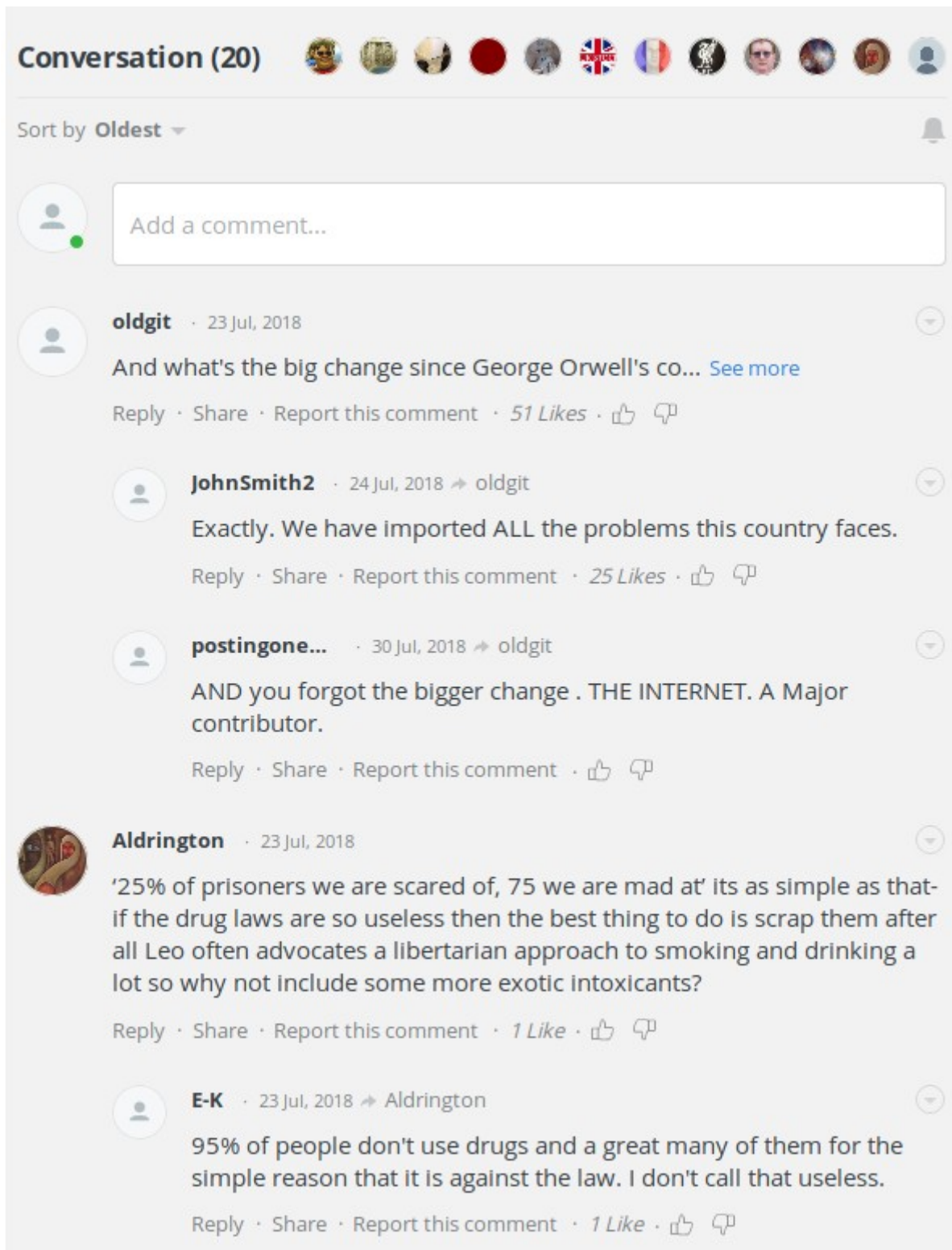


The R programming code is the same shown in annex 4 titled “Comparison between article’s text and comments”

Jaccard similarity value calculated is 0.28

## Daily Express newspaper web comments (internal id 108)

The next picture shows the comment section, user names are not obfuscated because they do not disclose real identities.



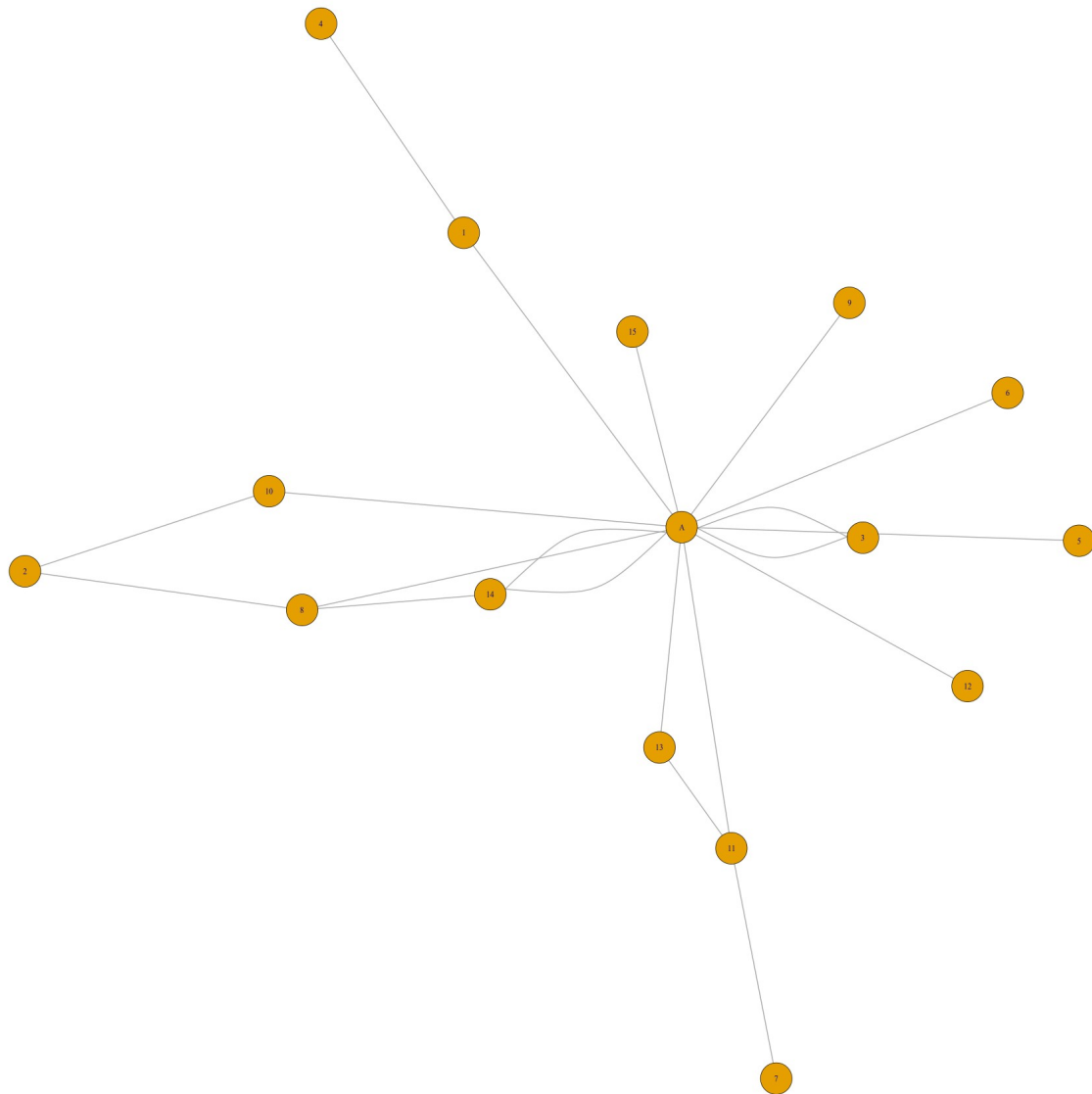
The screenshot shows a Facebook conversation interface. At the top, it says "Conversation (20)" followed by a row of 12 profile picture icons. Below this is a "Sort by Oldest" dropdown menu and a notification bell icon. A comment input box with a placeholder "Add a comment..." is visible. The conversation contains five comments:

- oldgit** · 23 Jul, 2018  
And what's the big change since George Orwell's co... [See more](#)  
Reply · Share · Report this comment · 51 Likes ·
- JohnSmith2** · 24 Jul, 2018 → oldgit  
Exactly. We have imported ALL the problems this country faces.  
Reply · Share · Report this comment · 25 Likes ·
- postingone...** · 30 Jul, 2018 → oldgit  
AND you forgot the bigger change . THE INTERNET. A Major contributor.  
Reply · Share · Report this comment ·
- Aldrington** · 23 Jul, 2018  
'25% of prisoners we are scared of, 75 we are mad at' its as simple as that- if the drug laws are so useless then the best thing to do is scrap them after all Leo often advocates a libertarian approach to smoking and drinking a lot so why not include some more exotic intoxicants?  
Reply · Share · Report this comment · 1 Like ·
- E-K** · 23 Jul, 2018 → Aldrington  
95% of people don't use drugs and a great many of them for the simple reason that it is against the law. I don't call that useless.  
Reply · Share · Report this comment · 1 Like ·

The analysis of HTML code and R programming code are the same as previous web comments (internal id 107)

The next illustration shows the network diagram

Web comments to article n. 108 - diameter = 4



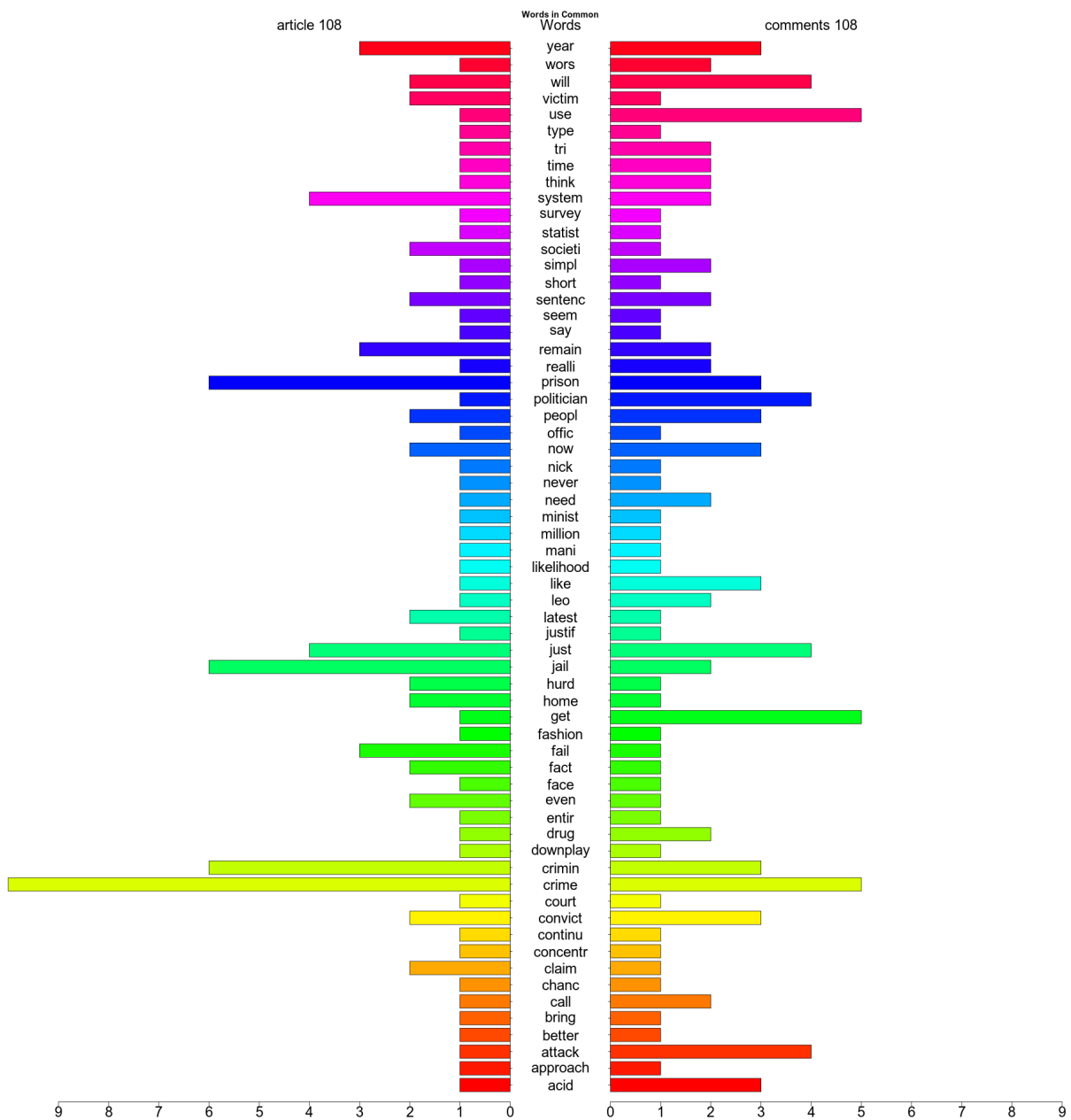
The central node is the article's author. All names are replaced by numbers. The network diameter is 4.

Now proceed with the comparison between the article's text and the set of comments.

The next illustrations show the word cloud representations of the article and the comments with a minimal frequency of two words and the pyramid diagram of common words.

govern  
offend  
system jail crime  
just murder  
now  
homeviolencwrote jul case  
year surg rise last victim  
decad one soft increas peopl knife betray week  
end polic .increas fair will cent mon fact  
sentenc also went societi hurd convict  
offenc polici britain 2018 claim result latest  
robber secretari number ever hope  
remain crimin per half  
fail term british  
prison public popul  
percent

simpl leo referendum  
countri punish like can  
peopl jail full vote convict yet time  
thing realli know  
just crime cprison  
world acid .crime real thinklaw  
trianyway capit useless  
violent attack must drug  
system sever lawyer will call  
chang sever stop import need  
remain bestgreedi wors  
sentenc corrupt live



The R programming code is the same shown in annex 4 titled “Comparison between article’s text and comments”

Jaccard similarity value calculated is 0.28