Sentiment Analysis & Keyword Clustering using RStudio

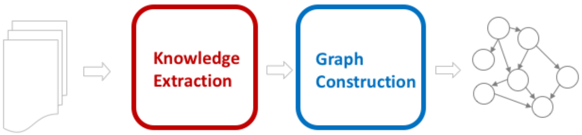
Problem:

The ever-growing interest in social web in the last decade has led to a nasty storm of issues such as misinformation, fake news and behavioural microtargeting. Although researchers have proposed numerous machine learning approaches to tackle these issues, however, deploying such solutions often require lots of computing resources. Furthermore, which content is or should be analysed is a choice of the developers.

To this end (and some other issues on the web) we designed a web annotation tool, oriented towards end-users. The tool allows end-user to select, store and link textual content on the fly. Now, to further empower end-users; we would like to allow end-users to analyse the sentiment of annotated content and then visualize the annotations as knowledge graphs. Doing so should enable better information exchange, thereby reducing the risks posed by fake news consumption.

Solution:

The tackle the previously mentioned challenge, we would like to carry out the following tasks. First, analyse the sentiments of the input textual data. Second, identify the keywords that summarize the text. And third, analyse the keyword - knowledge graph using different clustering methodologies.



Input Data: The dataset folder contains four different segments of articles including: Wikipedia page for 'Facebook–Cambridge Analytica data scandal' and articles about the same from Associated Press News, NBC News and Washington Times. For illustrative purposes, we currently only analyse the Wikipedia article.

Other large datasets that can be used to test the approach include: the 20 Newsgroups dataset, UCI Text Categorization dataset and Microsoft Research WikiQA Corpus.

Basic Sentiment Analysis:

To perform basic sentiment analysis, we first split the text into individual words i.e., tokens. We then check which tokens exist in the Bing and NRC lexicons and count the number of times the tokens occur in the text. The input data consisted of 11 Bing tokens and 57 NRC tokens as can be seen in Figure 1. The identified tokens are then categorized as Positive/Negative (Bing) and Fear/Anger/Trust/Sadness/Disgust/Joy/Surprise/Anticipation (NRC) and the results are presented in Figure 2 and 3.

For the current data, figures indicate that the context of the text is mostly negative, and the key emotional sentiments are trust and fear. This is again aligned with the context of the scandal (which we are all familiar with).

Keyword Extraction: For this step we use two different approaches i.e., ***TextRank*** and ***KCore Retention***. But before this step, the text is pre-processed using three steps:

- Tokenization: In this step, we split the text into words, symbols, phrases or other expressive elements called tokens.
- PoS Tagging & Selection: In this step, the tokens are marked with PoS (part of speech tags), based on both its definition as well as the context i.e. relationship with related and adjacent words in a sentence or phrase.
- Stopwords Removal: This step, filters out most common words.

We then use the *PageRank* function to identify the keywords from the list of pre-processed tokens. The top one-third list of keywords based on the token's occurrence count is printed as the keywords identified using *TextRank* (see Figure 4). The subgraph with the finally selected TextRank keywords is illustrated in Figure 5, followed by its respective heatmap illustration.

Followed by this, we calculate the coreness of the nodes in the token's graph. The nodes with the maximum coreness values are presented as the selected keywords (see Figure 4). The induced subgraph with the selected keywords is presented in Figure 6 as a graph (and associated heatmap).

The heatmaps are useful in visualizing the strong associations between the identified keywords. Figures 5 and 6 illustrate that there is close association between millions, scandal, analytica, personal and data. Again, reinforcing the key takeaways of the Cambridge Analytica Scandal.

Clustering Keywords: We now cluster the keyword graphs using three different methodologies. Since the input in form of a graph, we use community detection methods namely: edge-betweenness (Newman-Girvan), community detection based on propagating labels and greedy optimization of modularity. Figure 7 illustrates the communities in TextRank and KCore keywords.

These illustrations are especially useful as they can be used to summarize large chucks of text. For instance, the edge-betweenness textrank keyword communities shows the following summaries: (millions, people, facebook), (major, political, scandal), (hannes, das, magazine, swiss, publication) and (public, personal, data, cambridge, analytica). Whereas, the edge-betweenness kcore keyword communities shows the following summaries: (facebook, calls, massive, stock, price, fall) and (tighter, regulations, tech, companies, data).

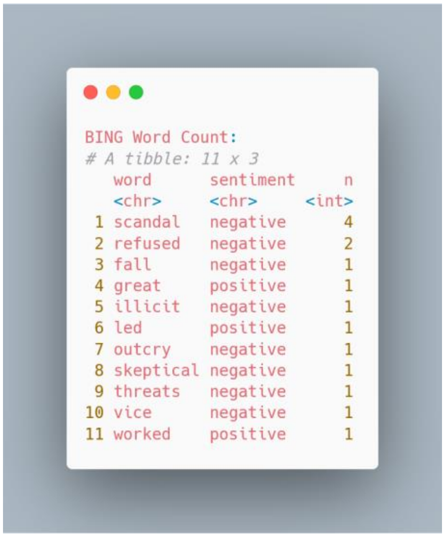
As an additional benefit of the used approaches, the code can also help users create new knowledge as it can help visualize patterns that might not be visible in huge volumes of text.

References:

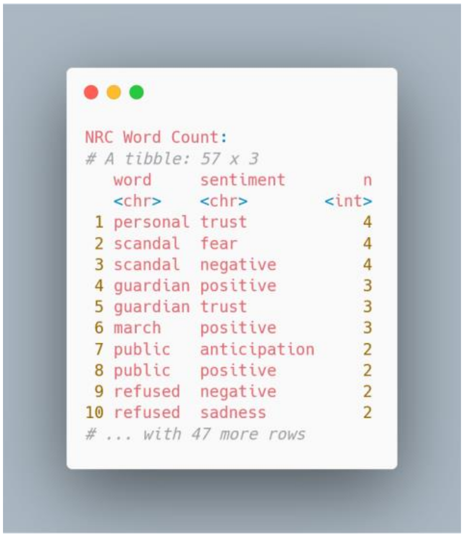
- RstudioPubs - Basic Text Mining in R [https://rstudio-pubs-static.s3.amazonaws.com/265713_cbef910aee7642dc8b62996e38d2825d.html]
- RstudioPubs - Basic Sentiment Analysis with R [https://rstudio-pubs-static.s3.amazonaws.com/302066_fe1dd2a635fa41198b18c87a64f5620c.html]
- RPubs - Single Document Keyword Extraction [https://www.rpubs.com/addyag/bda17]
- RPubs - Keywords Extraction using TextRank [https://rpubs.com/ivan_berlocher/79860]
- Kaggle - Tutorial: Sentiment Analysis in R [https://www.kaggle.com/rtatman/tutorial-sentiment-analysis-in-r]
- TextRank: Bringing Order into Texts by R. Mihalcea [https://web.eecs.umich.edu/~mihalcea/papers/mihalcea.emnlp04.pdf]
- CRAN - Textrank for Summarizing Text [https://cran.r-project.org/web/packages/textrank/vignettes/textrank.html]
- Main Core Retention on Graph-of-words for Single-Document Keyword Extraction by F. Rousseau et al. [http://frncsrss.github.io/papers/rousseau-ecir2015.pdf]
- RstudioPubs - Single Document Keyword Extraction [http://rstudio-pubs-static.s3.amazonaws.com/341868_231c841ed2d1476c9ccb3b7a07596a8c.html#kcore_retention]
- Network analysis with R and igraph: NetSci X Tutorial [https://kateto.net/networks-r-igraph]

Outputs:

Figure 1. Words from ***Bing*** and ***NRC Lexicon*** that are available in input text:



```
BING Word Count:
# A tibble: 11 x 3
   word      sentiment     n
   <chr>     <chr>     <int>
 1 scandal   negative      4
 2 refused   negative      2
 3 fall      negative      1
 4 great     positive      1
 5 illicit   negative      1
 6 led       positive      1
 7 outcry    negative      1
 8 skeptical negative      1
 9 threats   negative      1
10 vice      negative      1
11 worked    positive      1
```



```
NRC Word Count:
# A tibble: 57 x 3
   word     sentiment        n
   <chr>    <chr>        <int>
 1 personal trust            4
 2 scandal  fear             4
 3 scandal  negative         4
 4 guardian positive         3
 5 guardian trust            3
 6 march    positive         3
 7 public   anticipation     2
 8 public   positive         2
 9 refused  negative         2
10 refused  sadness          2
# ... with 47 more rows
```

Figure 2. ***Bing Lexicons*** in text contributing to different sentiments:

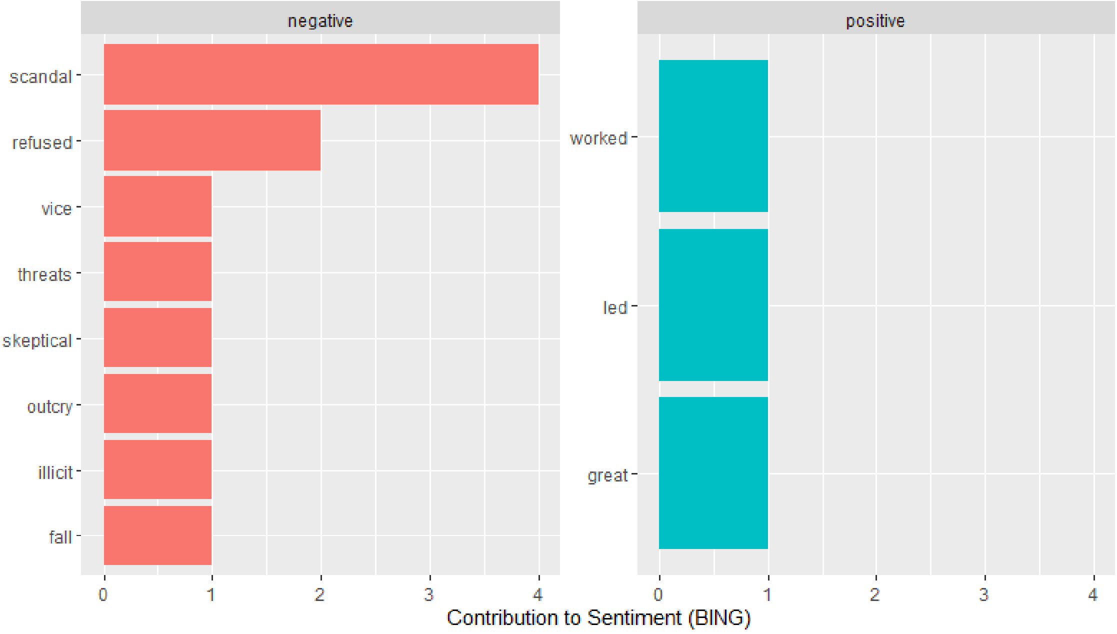


Figure 3. ***NRC Lexicons*** in text contributing to different sentiments:

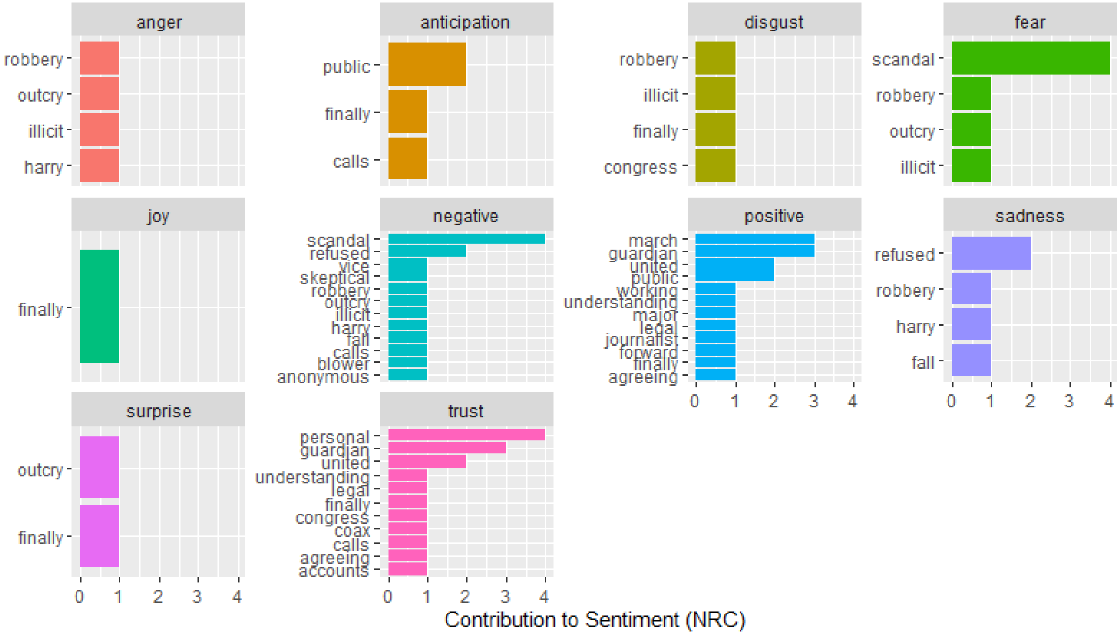
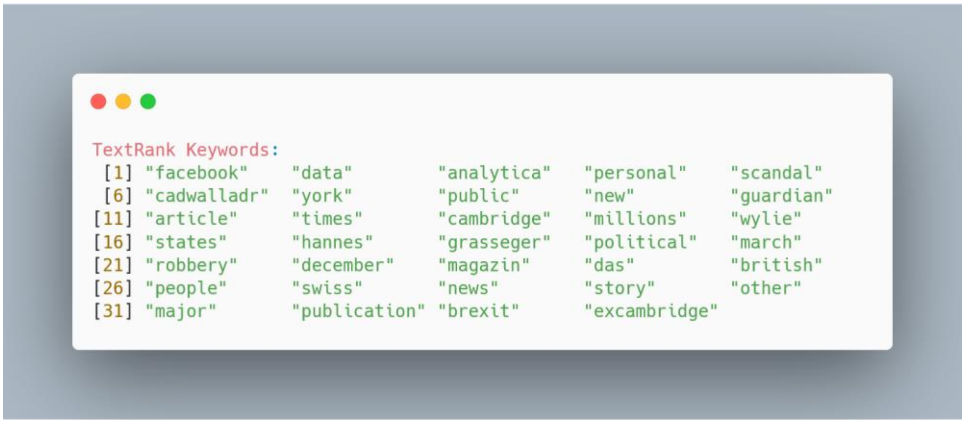
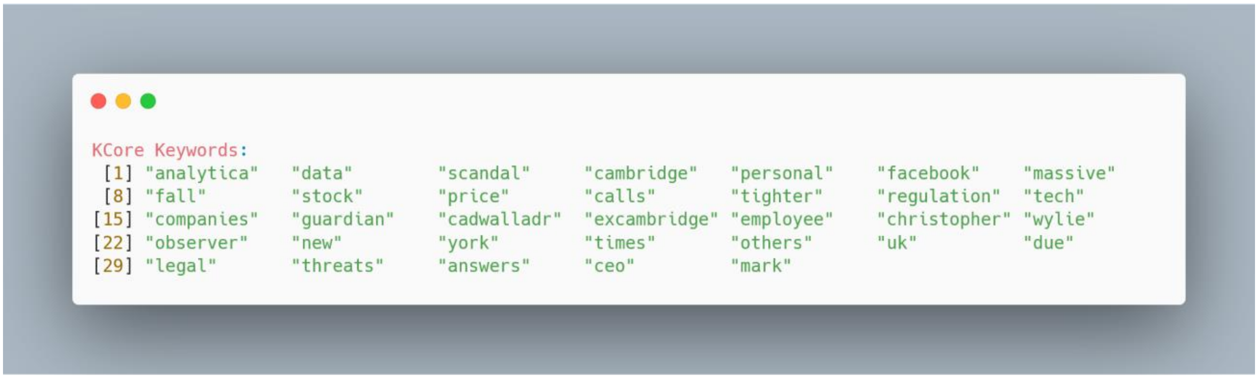


Figure 4. Keywords identified from input text (using ***TextRank*** and ***KCore algorithm***):

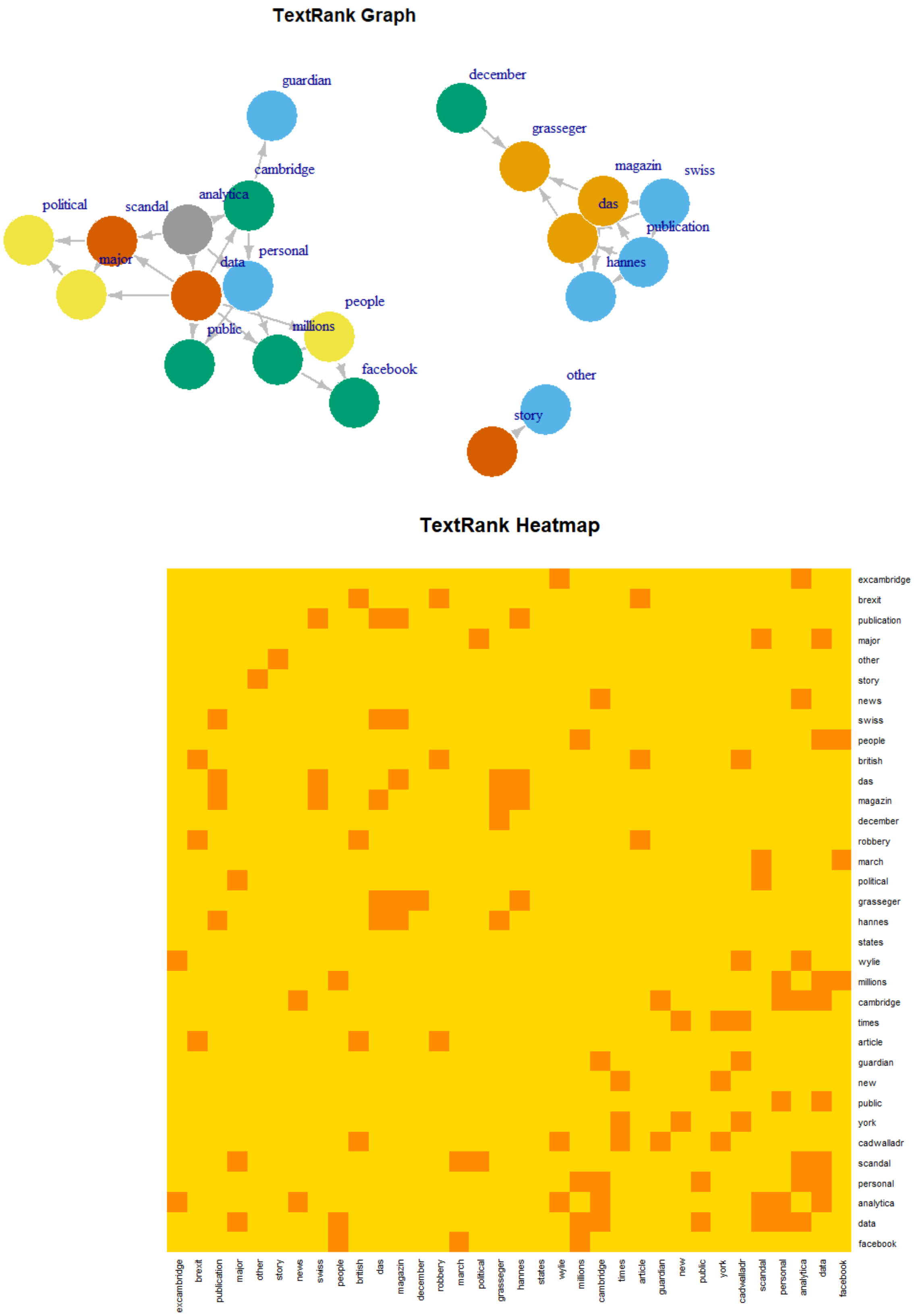


```
TextRank Keywords:
 [1] "facebook"   "data"        "analytica"  "personal"    "scandal"
 [6] "cadwalladr" "york"        "public"     "new"         "guardian"
[11] "article"    "times"       "cambridge"  "millions"    "wylie"
[16] "states"     "hannes"      "grasseger"  "political"   "march"
[21] "robbery"    "december"    "magazin"    "das"         "british"
[26] "people"     "swiss"       "news"       "story"       "other"
[31] "major"      "publication" "brexit"     "excambridge"
```



```
KCore Keywords:
 [1] "analytica" "data"     "scandal"    "cambridge"   "personal" "facebook"    "massive"
 [8] "fall"      "stock"    "price"      "calls"       "tighter"  "regulation"  "tech"
[15] "companies" "guardian" "cadwalladr" "excambridge" "employee" "christopher" "wylie"
[22] "observer"  "new"      "york"       "times"       "others"   "uk"          "due"
[29] "legal"     "threats"  "answers"    "ceo"         "mark"
```

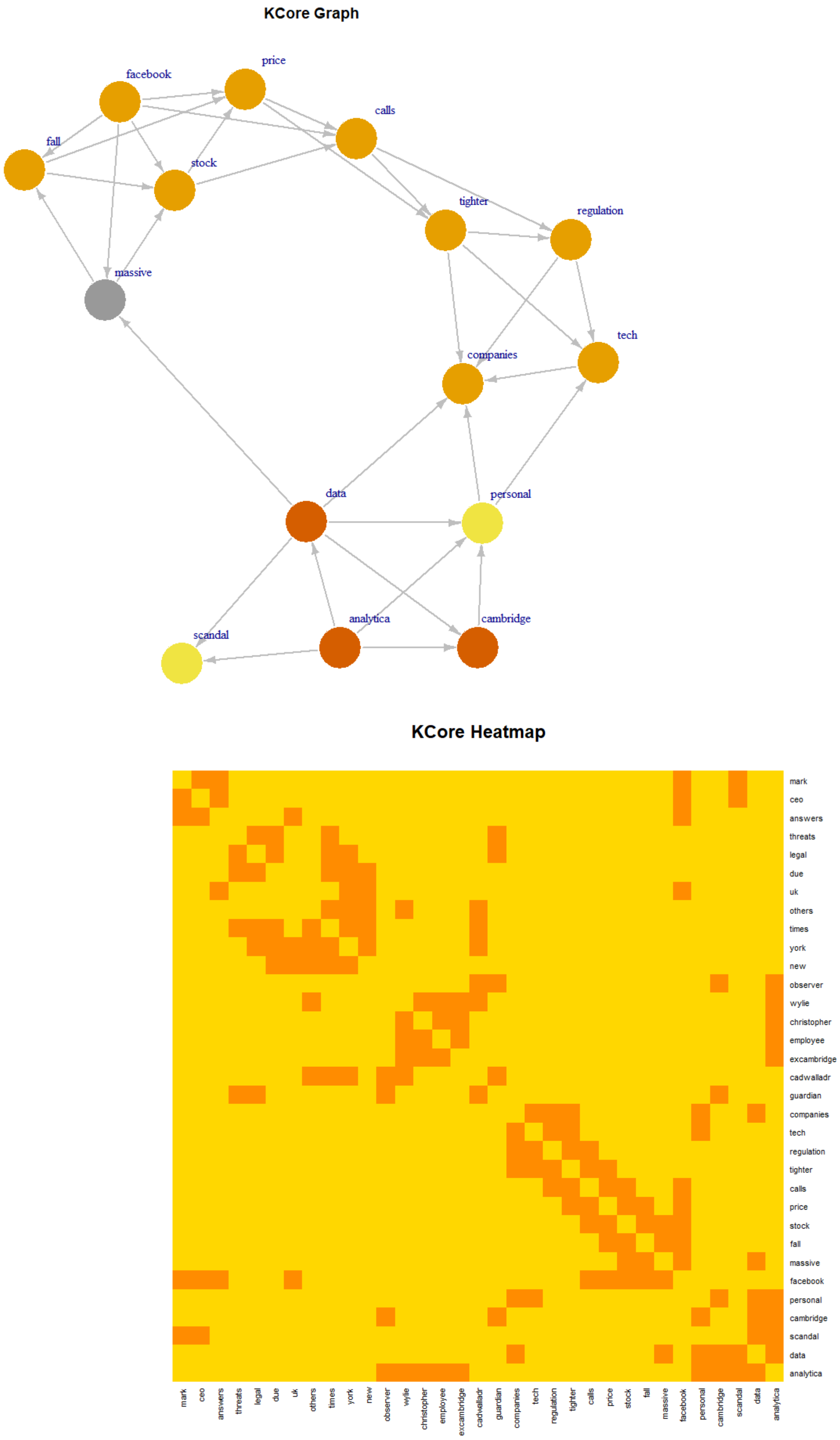
Figure 5. Knowledge graph and heatmap of *TextRank* keywords:



TextRank Graph

TextRank Heatmap

Figure 6. Knowledge graph and heatmap of ***KCore*** keywords:



KCore Graph

KCore Heatmap

Figure 7. Different clustering methods on ***TextRank*** knowledge graph:

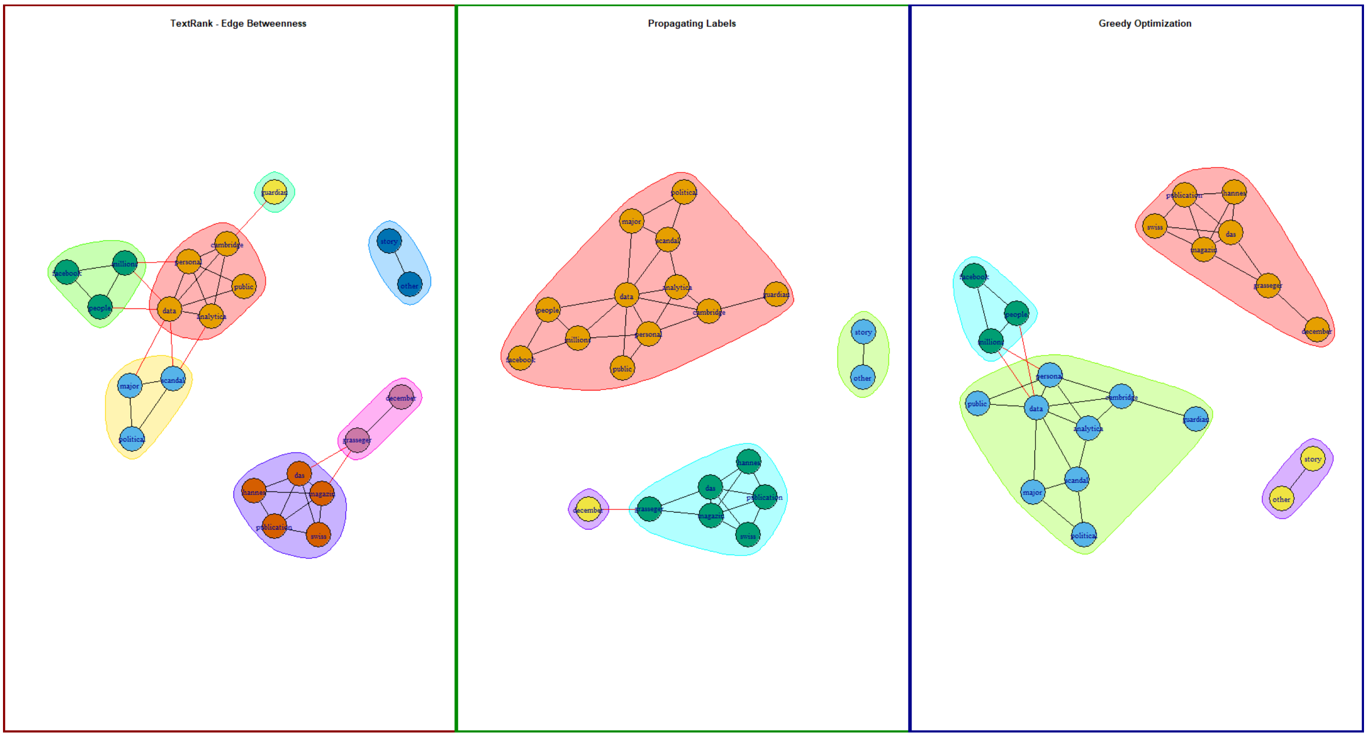
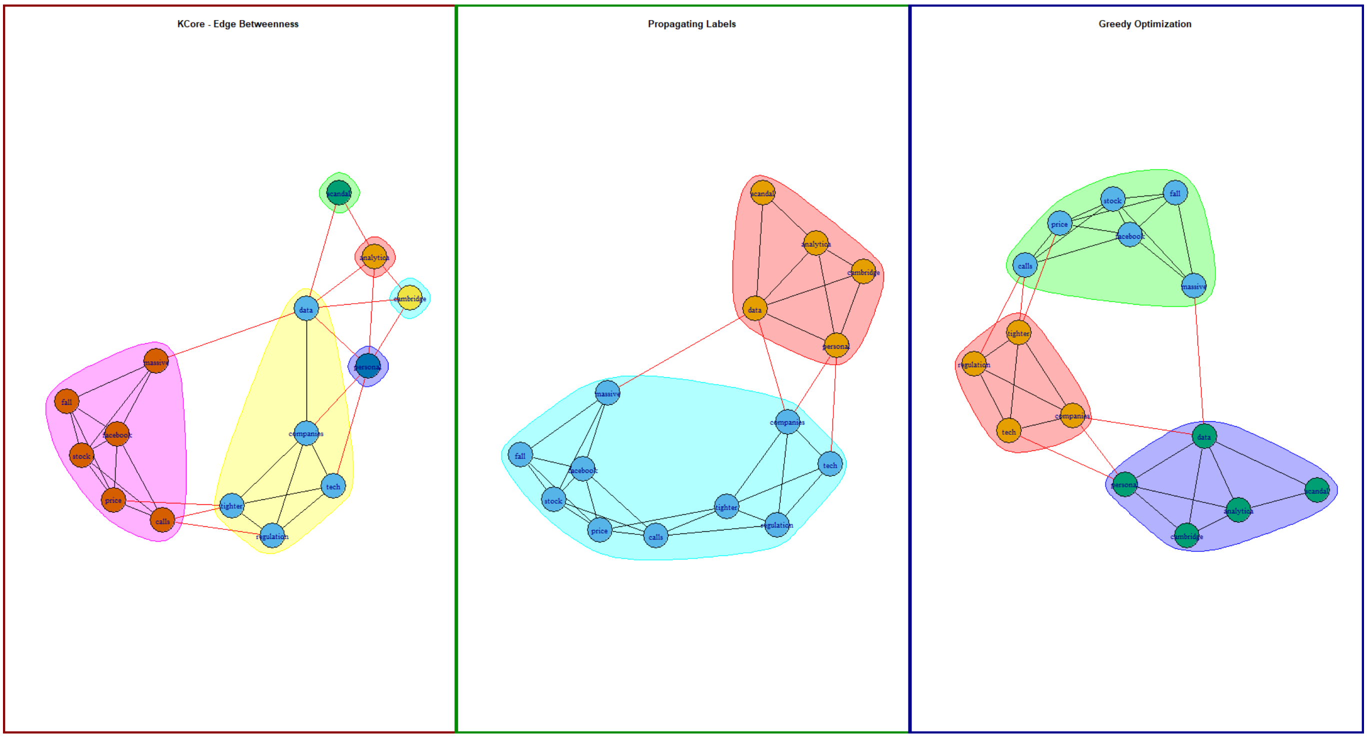


Figure 8. Different clustering methods on ***KCore*** knowledge graph:


KCore - Edge Betweenness
Propagating Labels
Greedy Optimization
scandal
analytica
cambridge
data
personal
massive
fall
facebook
stock
price
calls
companies
tech
tighter
regulation