An Analysis A Twitter Viral, #YellowVests, By Using Algorithms

Introduction:

The New Yorker journalist Luke Mogelson writes on his column Dispatch, that the Emmanuel Macron's new gas tax announcement originally galvanized the "*gilets jaunes"* on social media. The thousands of protesters belonging to the gilets jaunes movement—so called because of the yellow safety vests they wear rallied different parts of the France including Paris (Mogelson, 2018)

The beginning of the "*gilets jaunes"* is pretty interesting. According to Financial Times, "the gilets jaunes movement began as an online petition against fuel tax rises in the summer coordinated by a cosmetics saleswoman called Priscillia Ludosky from Seine-et-Marne, east of the capital. She joined forces with a truck driver from the same region who wanted to hold a day of demonstrations. The petition ballooned to a million signatures and on November 17 nearly 300,000 people blocked road junctions, tollbooths and fuel depots across France. And while the number of demonstrators has since shrunk — there is no formal structure or recognized leaders —the movement's demands and ambitions have soared" (Agnew & Hall, 2018).

Information technologies, especially social media plays a vital role in the development and persistence of many modern social movements. Among many, the Occupy Wall Street is notable for the apparent role social media played in facilitating communication among its participants (Conover, Ferrara, Menczer, & Flammini, 2013).

Twitter is one of the social media platforms that is gaining attention as a tool in protestors' inventory of contention, or the toolkit of strategies and tactics used to resist oppression (Tarrow, 2011).

The long-term goals of this research are try to understand how viral events work, how they impact the society, and their importance to the future generation—how would they possibly shape the future?

To understand these phenomena, this research will try to predict /analyze the effectiveness, connection and the impact of the Tweeter on the "*gilets jaunes"* movement by using specific key words from the extracted tweets through applying data/text mining applications.

Analysis:

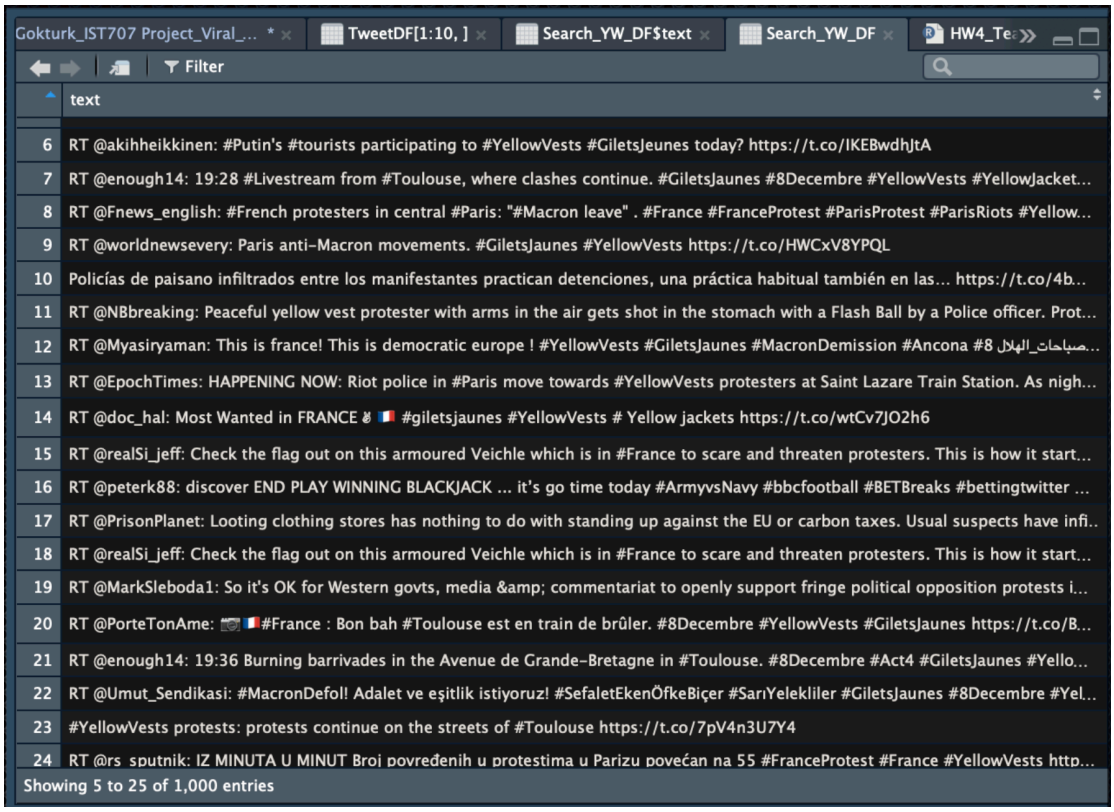
Twitter is a social networking platform that allows individuals to consume content from and contribute content to streams comprised of 280 character messages known as tweets(Tarrow, 2011). The twitter platform like many other social media platforms, has the potential to confer number of benefits to burgeoning social movements including communication between the individuals who participate to the movements (Conover et al., 2013).

In order to understand interaction or messaging behavior between the participants of the movement, the data have collected from Twitter by creating a Twitter Developer account.

Choosing appropriate algorithms is an important decision, and it requires knowledge of both the data set and the candidate algorithms (Tan, Kumar, & Steinbach, 2005).

Data:

Words choices are crucial to understand the movement. For this reason, #YellowVest hash-tag has been used to collect the data to analyze the communication between participants. 1000 tweets collected on December 1, 2018 as data. The data cleaning process is the major part of the research. The collected unstructured data converted to data frame has shown below.



Every tweet has multiple variables as shown below. This research has just analyzed the "text" data.

```
> names(tweets.df)
 [1] "text"          "favorited"     "favoriteCount" "replyToSN"     "created"       "truncated"     "replyToSID"    "id"
"replyToUID"    "statusSource"  "screenName"
[12] "retweetCount"  "isRetweet"     "retweeted"     "longitude"     "latitude"
>
```

To understand this phenomenon, the dataset will be analyzed by using some data-mining clustering algorithms tools such as k-Means, EM—Expectation Maximization, and HAC—Hierarchical agglomerative clustering as well as Support Vector Machine—SVM, Decision Tree, Random Forest and naïve Base algorithms.

Data Preparation – Creating Random Training And Test Datasets:

There are several steps needed to prepare the texts. Once the text documents convert to a corpus, the text documents in it need to be modified, e.g. stemming, stop words, punctuation, and number removals et cetera. In tm, all this functionality is subsumed into the concept of a transformation. Basically, all transformations work on a single text documents and *tm_map* just applies them all documents in a corpus (Feinerer, 2018)

In addition to previous data transformation, ignoring extremely rare words, which is less than 1% of the document and overly common words, which also appear more than 50% of the documents, and converting a document term matrix would be significantly help analyzing large dataset as shown below.

```
> tdm
<<TermDocumentMatrix (terms: 2010, documents: 1000)>>
Non-/sparse entries: 11537/1998463
Sparsity           : 99%
Maximal term length: 30
Weighting          : term frequency (tf)
```

In the tm package the classes TermDocumentMatrix and DocumentTermMatrix—depending on whether the dataset needs as rows and documents as columns, or vice versa) employ sparse matrices for corpora. Inspecting a term-document matrix displays a sample, whereas as.matrix () yields the full matrix in dense format (which can be very computer memory consuming for large matrices) (Feinerer, 2018).

Inspected document Term Matrices has shown below.

```
> (dtm <- DocumentTermMatrix(myCorpus))
<<DocumentTermMatrix (documents: 1000, terms: 1963)>>
Non-/sparse entries: 11129/1951871
Sparsity           : 99%
Maximal term length: 30
Weighting          : term frequency (tf)
>
```

Both DTM and TDM will be used for different algorithms.

The new DTM needs to convert to DTM matrix to analyze word frequencies. It seems new DTM is balanced—there is not significant difference between the values.

Cleaned data has converted to a Document Term Matrix has shown below.

```
     Terms
Docs  … forc franc french giletsjaun macron pari polic protest yellowvest
  110 0    1     0      0          0      0    1     0       0          0
  13  1    0     0      0          0      0    1     1       1          1
  153 1    0     0      0          0      0    1     1       1          1
  204 0    0     0      0          0      0    0     0       0          0
  452 0    0     0      0          0      0    0     0       0          1
  518 0    1     0      0          0      0    1     0       0          0
  645 0    0     0      0          0      0    0     0       0          0
  70  0    1     1      0          1      0    0     0       0          0
  74  0    0     0      0          0      0    0     0       0          0
  96  0    0     0      0          0      0    0     0       0          0
```

Frequency and Word Association

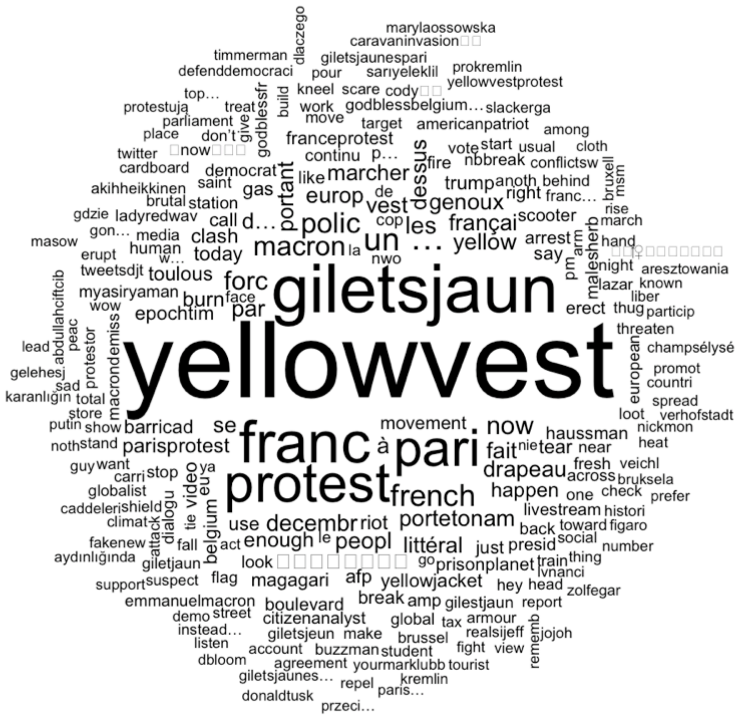
Words associated #yellowvest with correlation limit = 0.10 has shown in the chart below.

```
> findAssocs(x = tdm, terms = "yellowvest",corlimit = 0.1 )
$yellowvest
      alexandrosm           bruxell              demo           brussel
             0.25              0.24              0.24              0.23
          belgium           gelehe…           belgiqu         giletjaun
             0.22              0.22              0.21              0.21
             riot              pari            outsid            \u2b07
             0.20              0.17              0.17              0.17
          protest        giletsjaun          epochtim            happen
             0.16              0.16              0.15              0.15
             face               afp             clash           toulous
             0.14              0.14              0.13              0.13
     parisprotest yellowvestprotest             break               one
             0.13              0.13              0.13              0.13
              eye               got              lost             women
             0.13              0.13              0.13              0.13
          daimler         boulevard             erect          haussman
             0.13              0.12              0.12              0.12
       malesherb                 p…           scooter          movement
             0.12              0.12              0.12              0.12
    franceprotest      yellowjacket          democrat       myasiryaman
             0.11              0.11              0.11              0.11
             call           dialogu             fresh                pm
             0.11              0.11              0.11              0.11
   citizenanalyst           continu       macrondemiss            anoth
             0.10              0.10              0.10              0.10
```

Word Cloud

It would be helpful to have a look the most frequently used words in the new DF by creating visualization, a word cloud, also referred as text cloud or a visual representation of text data. The text mining package *tm* and the word cloud generator package *wordcloud* are evidently some of the best tool for this (Feinerer, 2018)

Both word cloud and frequency word association had similar results as shown below.

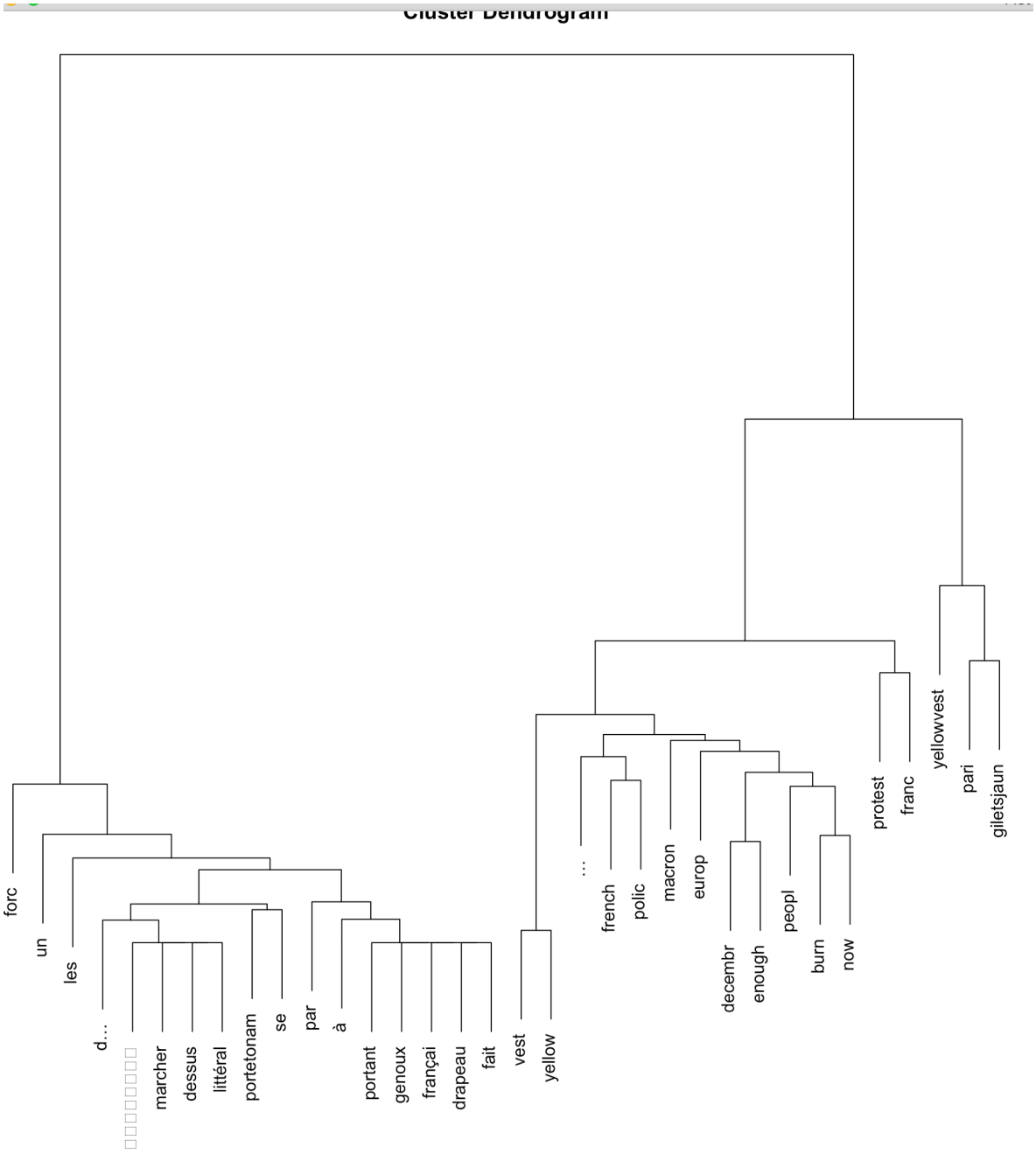


Results

Distance Measures

Euclidean Distance

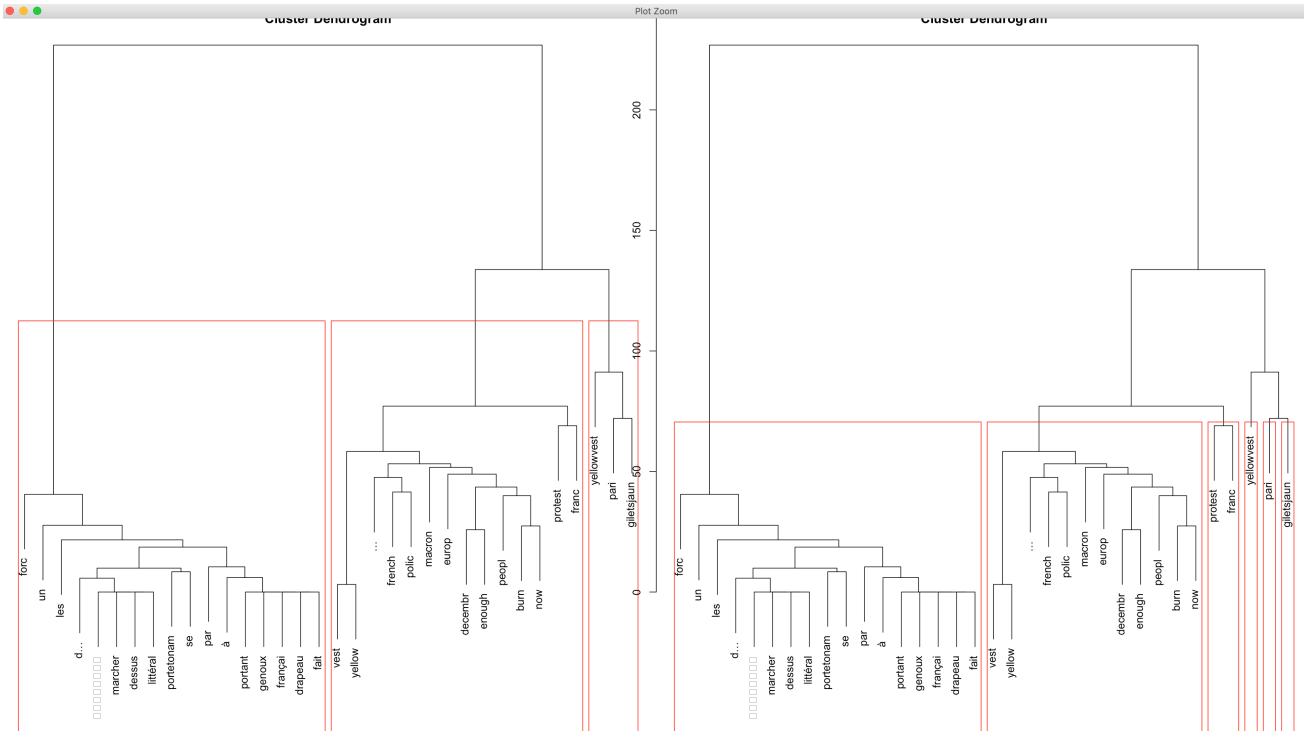
To assign a point to the closest centroid, dataset need a proximity measure that quantifies the notion of "closest" for the specific data under consideration. Euclidean (L_2) distance is often used for data points in Euclidean space, while cosine similarity is more appropriate for documents. However, there may be several types of proximity measures that are appropriate for a given type of data. For instance, Manhattan (L_1) distance can be used for Euclidean data, while the Jaccard measure is often employed for documents (Tan et al., 2005).



HAC—Hierarchical Agglomerative Clustering

Hierarchical clustering algorithms are either top-down or bottom-up. Bottom-up algorithms treat each paper as a singleton cluster at the outset and then successively merge (or agglomerate) pairs of clusters until all clusters have been merged into a single cluster that contains all documents(Manning, Raghavan, & Schütze, 2009).

Bottom-up hierarchical clustering is also called hierarchical agglomerative clustering or HAC. Top-down clustering requires a method for splitting a cluster. It proceeds by splitting clusters recursively until individual documents are reached (Manning et al., 2009).



An HAC clustering is typically visualized as a dendrogram as shown above. Despite some distortion in the data, k=3 clusters seems a better result than k=6.

K - means Clustering

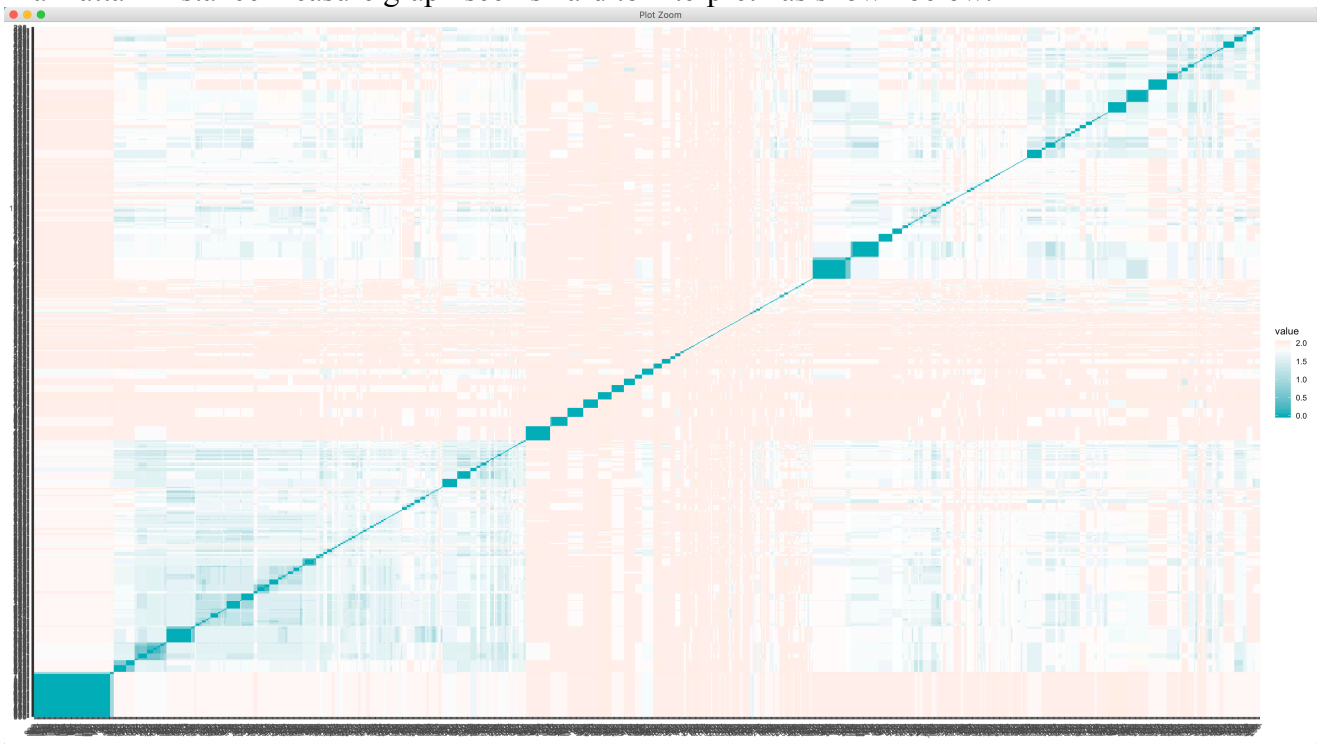
Some clustering techniques, such as K-means, have linear or near-linear time and space complexity and thus, an outlier detection technique based on such algorithms can be highly efficient. Also, the definition of a cluster is often complementary to that of an outlier and thus, it is usually possible to find both clusters and outlier at the same time (Tan et al., 2005).

On the other hand, clustering techniques such as K-means do not automatically determine the number of clusters. This is a problem when using clustering in outlier detection, since whether an object is considered an outlier or not may depend on the number of clusters(Tan et al., 2005).

K =6 cluster results has shown below.

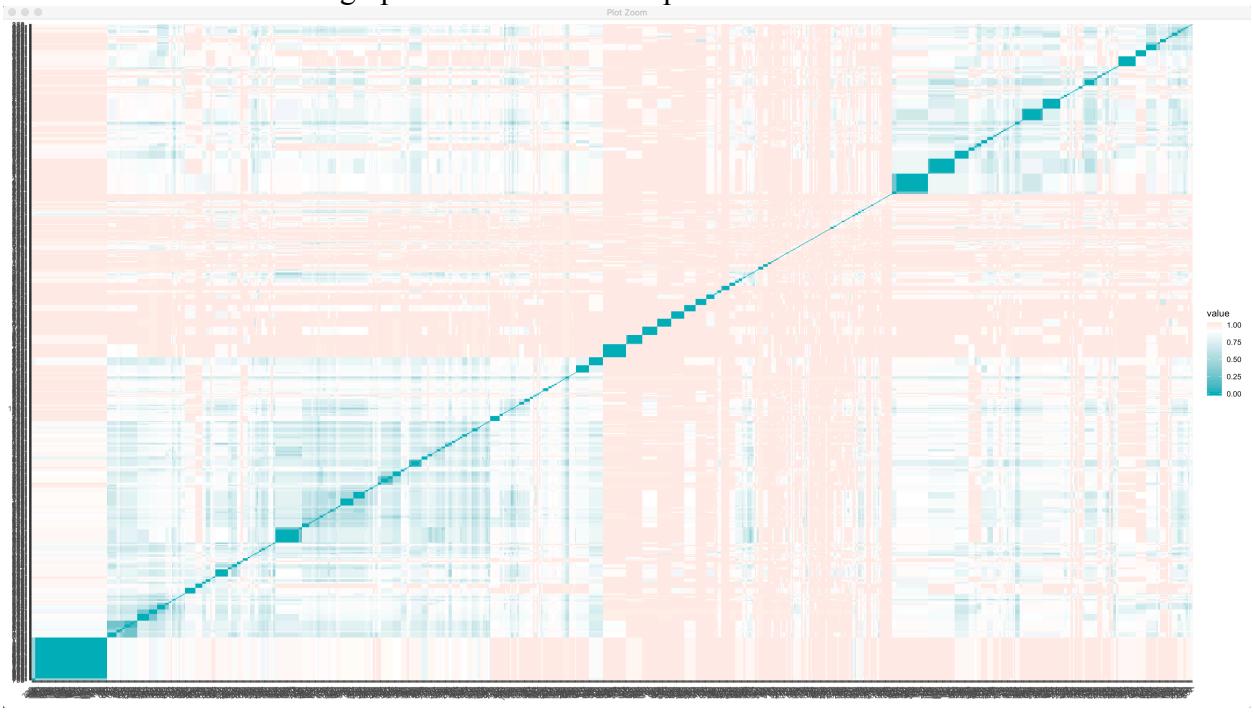
```
> k <- 6 #number of cluster
> kmeansResult <- kmeans(m3, k)
> round(kmeansResult$centers, digits = 3) ## cluster centers
   burn   now protest yellowvest  pari giletsjaun  vest yellow europ decembr enough     … franc
1 0.007 0.056   0.007      0.309 0.108      0.042 0.045  0.045 0.087   0.017  0.007 0.052 0.087
2 0.025 0.025   0.068      0.970 0.492      0.903 0.004  0.004 0.008   0.237  0.165 0.140 0.042
3 0.144 0.201   1.066      0.808 0.367      0.052 0.170  0.170 0.048   0.000  0.048 0.336 0.245
4 0.127 0.169   0.254      0.366 0.296      0.183 0.000  0.000 0.000   0.028  0.000 0.099 0.437
5 0.000 0.000   0.000      0.000 0.000      1.000 0.000  0.000 0.000   0.000  0.000 0.000 1.000
6 0.027 0.054   0.000      1.009 0.108      0.523 0.108  0.117 0.234   0.063  0.009 0.099 1.135
  french macron polic portetonam peopl  forc dessus drapeau    d… fait françai genoux   les
1  0.090  0.000 0.052      0.007 0.049 0.059  0.000       0 0.003    0       0      0 0.031
2  0.081  0.051 0.008      0.004 0.093 0.008  0.000       0 0.000    0       0      0 0.013
3  0.275  0.009 0.341      0.000 0.074 0.009  0.000       0 0.000    0       0      0 0.000
4  0.099  1.155 0.070      0.000 0.211 0.014  0.000       0 0.000    0       0      0 0.000
5  0.000  0.000 0.000      0.954 0.000 1.000  0.954       1 0.954    1       1      1 0.954
6  0.027  0.027 0.054      0.009 0.000 0.009  0.000       0 0.000    0       0      0 0.018
  littéral marcher   par portant    se    un     à \U0001f534\U0001f4f9\U0001f1eb\U0001f1f7
1    0.000   0.000 0.000       0 0.003 0.003 0.003                                    0.000
2    0.000   0.000 0.000       0 0.000 0.008 0.000                                    0.000
3    0.000   0.000 0.000       0 0.000 0.004 0.000                                    0.000
4    0.000   0.000 0.000       0 0.000 0.000 0.000                                    0.000
5    0.954   0.954 1.000       1 0.954 1.954 1.000                                    0.954
6    0.000   0.000 0.045       0 0.000 0.000 0.000                                    0.000
```

Manhattan Distance measure graph seems hard to interpret has shown below.



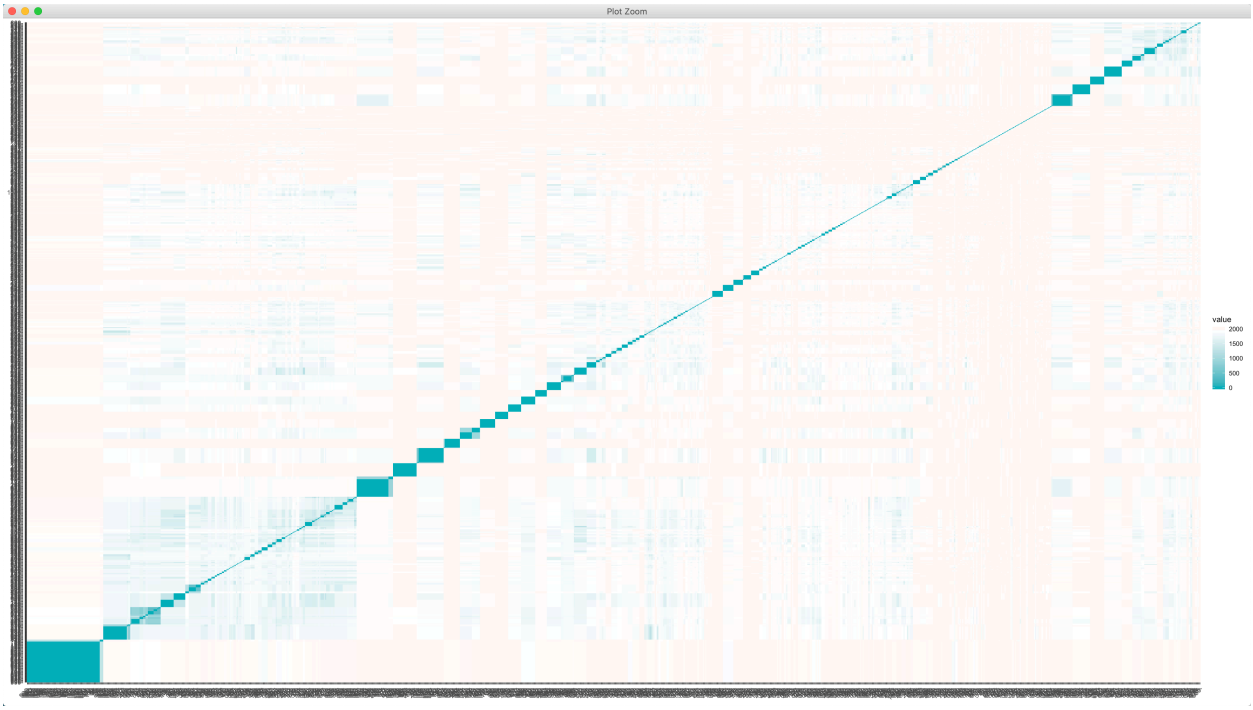
Pearson Distance

Pearson distance measure graph seems hard to interpret has shown below.



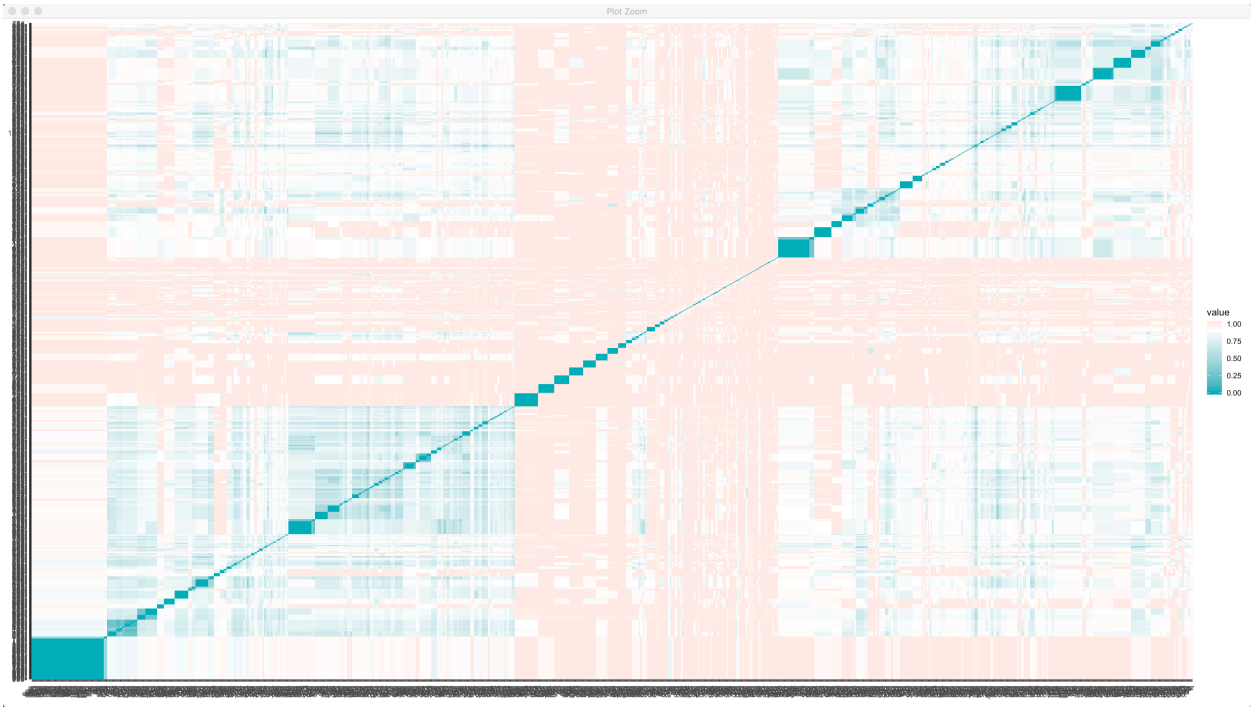
Canberra distance

Because of the dimension of the data, it is difficult to interpret Canberra distance measure has shown below.



Spearman Distance

Because of the dimension of the data, it is difficult to interpret Spearman distance measure has shown below.

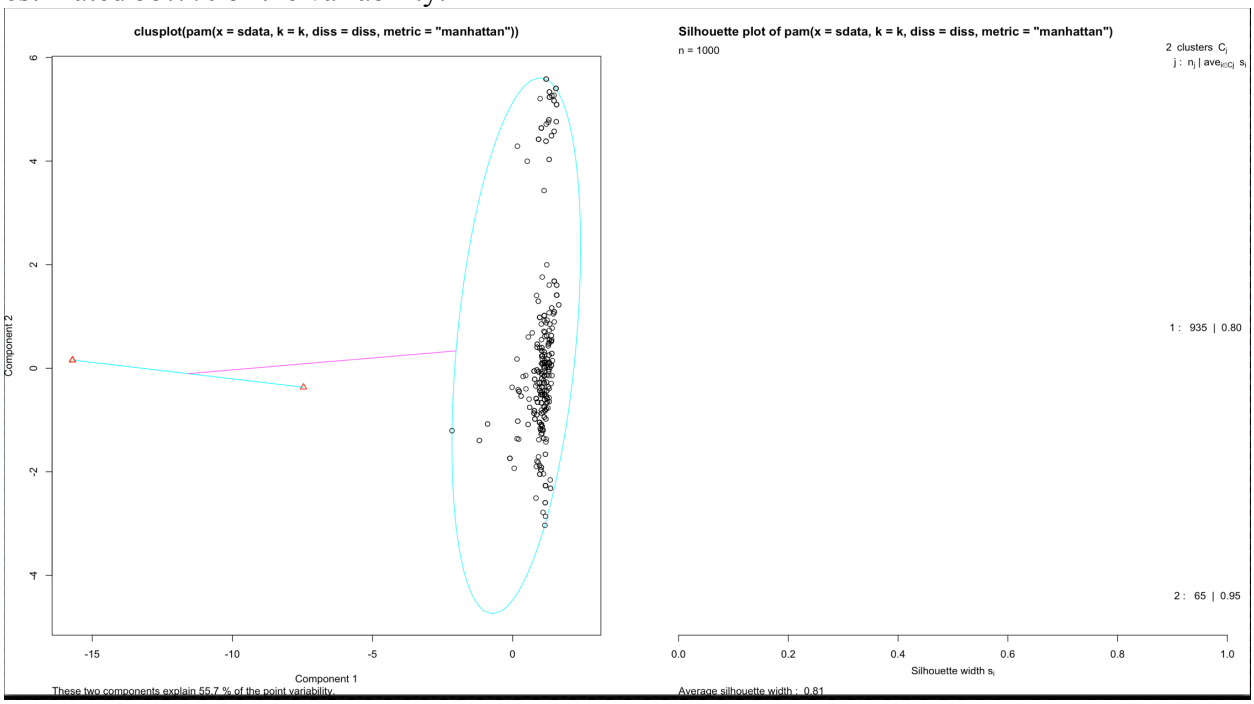


Partitioning around medoids with estimation of number of clusters

In the partitioning around medoids with Manhattan distance, with the number of clusters—nc=2 estimated by optimum average silhouette width or Calinski-Harabasz index. The Duda-Hart test is applied to decide whether there should be more than one cluster—unless 1 is excluded as number of clusters or data are dissimilarities (Feinerer, 2018).Each objects supposed to be part of the cluster whose value difference is minimal, comparing the other clusters.

```
Objective function:
build  swap
2.499 2.499

Available components:
 [1] "medoids"    "id.med"     "clustering" "objective"  "isolation"
 [6] "clusinfo"   "silinfo"    "diss"       "call"       "data"

$nc
[1] 2

$crit
 [1] 0.0000000 0.8127627 0.2708312 0.3070638 0.2505255 0.2492082
 [7] 0.2748011 0.2904164 0.3096780 0.2449174

> k <- pamResults$nc # number of clusters identified
> pamResults <- pamResults$pamobject
> for (i in 1:k) {
+   cat("cluster", i, ": ",
+       colnames(pamResults$medoids)[which(pamResults$medoids[i,]==1)], "\n")
+ }
cluster 1 :  yellowvest
cluster 2 :  giletsjaun franc portetonam forc dessus drapeau d… fait françai genoux les littéral marcher par portant se à 🔴📹🇫🇷
>
```

In the partitioning around medoids with Manhattan distance, with the number of clusters—nc=2 estimated 55.7% of the variability.



Topic model

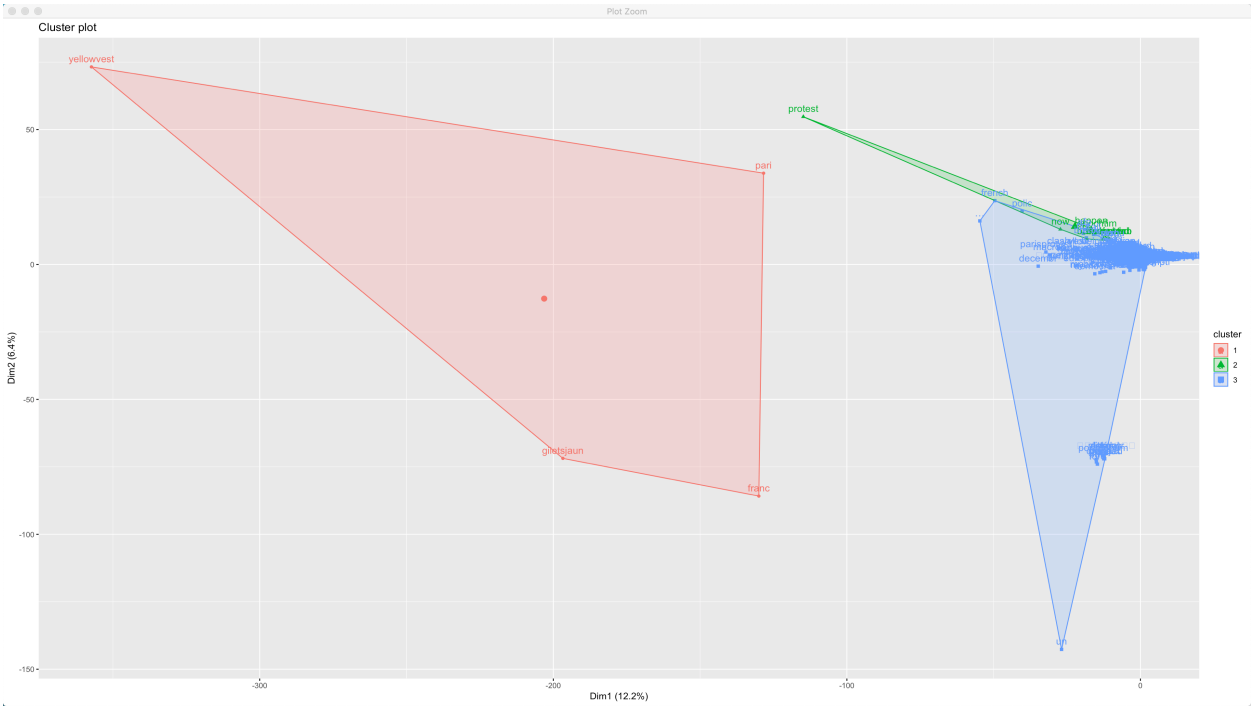
The topic model Function extracts the most likely terms for each topic or the most likely topics for each document as shown below.

```
> #### Topic Model
> dtm <- as.DocumentTermMatrix(tdm)
> lda <- LDA(dtm, k = 8) # find 8 topics
> term <- terms(lda, 4) #first 4 terms of every topic
> term
     Topic 1          Topic 2 Topic 3      Topic 4      Topic 5
[1,] "giletsjaun"     "un"    "yellowvest" "macron"     "franc"
[2,] "yellowvest"     "les"   "protest"    "yellowvest" "yellowvest"
[3,] "pari"           "forc"  "giletsjaun" "franc"      "protest"
[4,] "citizenanalyst" "par"   "decembr"    "use"        "giletsjaun"
     Topic 6      Topic 7      Topic 8
[1,] "yellowvest" "yellowvest" "yellowvest"
[2,] "giletsjaun" "protest"    "say"
[3,] "pari"       "polic"      "pari"
[4,] "peopl"      "pari"       "amp"
```

K-Means, centers = 3

In the Centroid based clustering method every cluster is referenced by a vector of values. Each objects supposed to be part of the cluster whose value difference is minimal, comparing the other clusters. Because of the pre-defined number of clusters, algorithm is kind of problematic. In this

case, the R code enforces the data fit in 3 clusters. However, the result is not useful to explain the model as shown below.



Naïve Bayes

All applied learning algorithms based on Bayes theorem make some independence assumptions. The naïve Bayes method takes this to the extreme by assuming that the attributes are statistically independent given the class. This causes to a simple algorithm where training time is linear in both the number of instances and attributes (Hall, 2007).

Even thought the independence assumption is unacceptably violated in practice, naïve Bayes performs remarkably well on many classification cases. However, because of this assumption, the subsequent probabilities valued by naïve Bayes are usually poor, such as in an extreme case where a single redundant attribute—an attribute which is perfectly correlated with another, is present in the data, that attribute effectively has twice as much as impact as the other attributes (Hall, 2007).

To analyze the accuracy of the naïve Bayes algorithm, deception data set analyzed by two different R packages:

- Package “e1071” functions for latent class analysis, short time Fourier transform, fuzzy clustering, support vector machines, shortest path computation, bagged clustering, naive Bayes classifier (David Meyer, Evgenia Dimitriadou, Kurt Hornik, Andreas Weingessel, & Friedrich Leisch, 2018). The result is not responsive.

- Package “naivebayes” computes the conditional a-posterior probabilities of a categorical class variable given independent predictor variables using the Bayes rule(Majka, 2018). The result is not responsive.

Support Vector Machine Classifier —SVM

The Support Vector Machine—SVM has been proposed first in Vladimir Naumovich Vapnik's book "The Nature of Statistical Learning Theory" in 1995 (Vapnik, 1995). The SVM technique has been used in different application domains and has outperformed the traditional techniques in terms of generalization capability. The major advantage of SVM is that, contrary to the traditional techniques, which try to minimize the empirical risk, the classification error on the training data, SVM minimizes the structural risk—the classification error on data (Chmielnicki & Stapor, 2010).

For the SVM classifier with polynomial kernel option result, the label: yellowvest removed train data was used. The SVM classifier confusion matrix is shown in figure below.

```
> print(SVM_yv)

Call:
svm(formula = yv_Train$yellowvest ~ ., data = yv_Train, kernel = "polynomial",
    cost = 100, scale = FALSE)


Parameters:
   SVM-Type:  eps-regression
 SVM-Kernel:  polynomial
       cost:  100
     degree:  3
      gamma:  0.000509684
     coef.0:  0
    epsilon:  0.1


Number of Support Vectors:  656
```

However, the SVM classifier with radial kernel option result is less accurate than polynomial kernel as shown below

```
> print(SVM_yv_L)

Call:
svm(formula = yv_Train$yellowvest ~ ., data = yv_Train, kernel = "linear", cost = 100,
    scale = FALSE)


Parameters:
   SVM-Type:  eps-regression
 SVM-Kernel:  linear
       cost:  100
      gamma:  0.000509684
    epsilon:  0.1


Number of Support Vectors:  293
```

All SVM classifier with polynomial, linear and radial kernel options different results. The SVM linear has a better result than the others.

```
> print(SVM_yv_r)

Call:
svm(formula = yv_Train$yellowvest ~ ., data = yv_Train, kernel = "radial", cost = 100,
    scale = FALSE)


Parameters:
   SVM-Type:  eps-regression
 SVM-Kernel:  radial
       cost:  100
      gamma:  0.000509684
    epsilon:  0.1


Number of Support Vectors:  310
```

Chi-square test also proves the inaccuracy of method as shown below.

Results

Both naïve Bayes models with the R Packages "naivebayes" and "e1071" have unresponsive; because of the data. The most clustering algorithms have highly accurate results.

Algorithm performance comparison

Apparently the Support Vector Machine—SVM, algorithms have less accurate results than the clustering algorithms. However none of these models have accurately predicted the sentiments of the reviews.

Conclusion

The Support Vector Machine—SVM, naïve Bayes and decision Tree classification algorithms are some of the most popular algorithms for data mining applications in data science to help researchers to understand these phenomena.

Unfortunately none of these algorithms have provided 100% accurate results regarding recognition of the truth or lie because of the inadequately structured corpus of the dat. Further research with a better prepared and cleaned data could predict to correct word predictions. With that being said, as seen in the word cloud, the majority of the poorly written words, other languages on top of that limited computing power make words prediction significantly difficult.

Evidently, Clustering, SVMs, kNN, and Random Forest as well as naïve Base algorithms are fairly common for many complex studies and applications. However, as Kumar noted that there is no silver bullet in terms of algorithm comparison – no algorithm would outperform all other algorithms on all data sets (Tan et al., 2005).

It was a good start for a new Twitter/text miner who tries to understand the virality of the events.

Bibliographies:

Agnew, H., & Hall, B. (2018). 'Look at me, I exist': French protesters send message to Macron. *FT.com*.

Chmielnicki, W., & Stapor, K. (2010). Investigation of Normalization Techniques and Their Impact on a Recognition Rate in Handwritten Numeral Recognition. *Schedae Informaticae, 19*, 53-77.

Conover, M. D., Ferrara, E., Menczer, F., & Flammini, A. (2013). The Digital Evolution of Occupy Wall Street. *PLoS One, 8*(5). doi:http://dx.doi.org/10.1371/journal.pone.0064679

David Meyer, Evgenia Dimitriadou, Kurt Hornik, Andreas Weingessel, & Friedrich Leisch. (2018). Package 'e1071'.

Feinerer, I. (2018, Oct. 2008). An introduction to text mining in R. *R News.* Retrieved from https://cran.r-project.org/web/packages/tm/vignettes/tm.pdf

Hall, M. (2007). A decision tree-based attribute weighting filter for naive Bayes. *Knowledge-Based Systems, 20*(2), 120-126. doi:http://dx.doi.org/10.1016/j.knosys.2006.11.008

Majka, M. (2018). High Performance Implementation of the Naive Bayes Algorithm.

Manning, C. D., Raghavan, P., & Schütze, H. (2009). *Introduction to information retrieval*.

Mogelson, L. (2018). Inside the Chaos of the Gilets Jaunes Protests.

Tan, P.-N., Kumar, V., & Steinbach, M. (2005). *Introduction to data mining* (1st ed.). Boston: Pearson Addison Wesley.

Tarrow, S. G. (2011). *Power in movement : social movements and contentious politics* (Rev. & updated 3rd ed.). Cambridge ; New York: Cambridge University Press.

Vapnik, V. N. (1995). *The nature of statistical learning theory* (Second Edition ed.). Springer, New York.