SOCIAL AND BEHAVIORAL NETWORKS
Final Project

Cecilia Martinez Oliva

Abstract

The purpose of this work is to analyze tweets supporting or not the 4th dicember italian referendum. In the first part the aim is to find groups of terms capturing the collective attention of the two different parties. In the second part, instead, the target is to search for the more influent users for each party, trying to identify them also thanks to the previously found politicians and the words they used. Finally it analyse the spread of influence of the two parties.

Contents

Introduction

Starting from 18850000 tweets and 2287781 users in total collected from 26/11/2016 to 06/12/2016.

First of all I created an index for all the collected tweets.

- *date* is a LongField and contains the date in the timestamp format.
- *id* is also a LongField and contains the twitter id of the user.
- *screenname* is a StringField, user's Twitter screen name.
- *name* is a TextField, the user's name. Special characters are removed.
- *followers* is a LongField, follower's number of the user.
- *text* is a TextField, the text of the tweet. In this case there is a stronger analyzer and there is the possibility to use stemming or not.
- *hashtags* is a TextField, it's applied lower case and tokenization.
- *mentions* is a TextField, mentioned users in the tweet. Only tokenization.

I used the SimpleAnalyzer in the name field in order to remove special characters. In the text field we can choose to use the StopAnalyzer (with italian stopwords) or to use also the stemmer; in the latter case the analyzer is the ItalianAnalyzer. For hashtags and mentions is used the WhitespaceAnalyzer and in the first case is applied also the lower case.
However finally for this work I decided to use the ItalianAnalyzer for the tweet's text.
Then I also created another index, from the first one, for all the collected users.

- *id* is a LongField.
- *screenname* is a StringField.
- *name* is a TextField, again using the SimpleAnalyzer.
- *followers* is a LongField.

Since screen names are unique, starting from all the screen names saved in the previous index, I query each screen name and get only the first result to get the informations needed.

1 Temporal Analysis

1.1

I manually searched on the web for politicians and journalists. To obtain bigger lists of yes or no supporters, I automatically searched for their screen name and their support.
Using the users index, I query the name and I suppose that the real screen name is associated to the one with more followers.
I then searched for the more indicative hashtags as "bastaunsi", "iovotosi", "iodicono", "iovotono", to classify the users. If they use more yes-hashtags are classified as yes-users, if they use more no-hashtags they are classified as no-users, otherwise they are not classified and are not considered in the lists.
Those lists are saved in two files: yes-p.txt and no-p.txt. The easier way to have a fast access to the tweets we are interested to is to create two new indeces: TwitterIndexYES, TwitterIndexNO. In Table 1 it's shown the real number of people and tweets found for each index.

	YES	NO
n.users	497	605
n.tweets	33200	59887

Table 1: YES/NO supporters

The number of tweets over time is represented in Figure 1. As we can imagine, the number of tweets

decreases from 00:00:00 to 12:00:00.

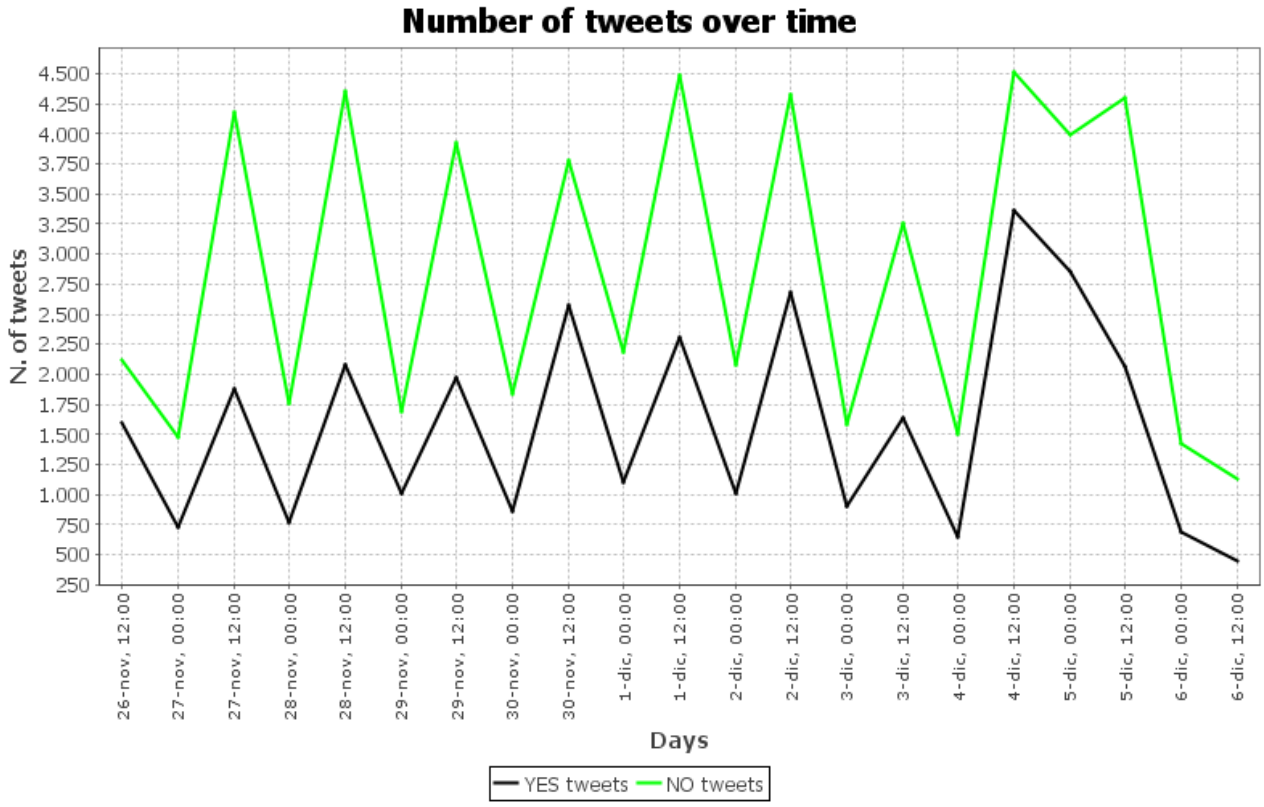

Figure 1: Number of tweet over time

1.2

We now want to collect the top 1000 words by frequencies that expose the typical pattern that capture the collective attention representative for YES and NO subset of users.
What I am doing is to first collect all terms in the index (excluding terms not containing any alphabet character and shorter than 3 letters), ordered by frequency, and then, using a range filter on dates, I do a query for each term to build SAX. In this way we can save the first 1000 terms matching the pattern of collective attention (one or two picks).
In this case SAX was built with 24h grain and two alphabet characters. Once the term is chosen, I rebuild SAX with grain 12h and numbers instead of alphabet characters.
I then wrote a kmeans algorithm to cluster terms. In order to find terms where SAX behavior is very similar I chose k=20 and 10000 randomizations (plots for the clusters are in the folder).
I also saved 100 tweets for each term to have a check of what was going on.
If we choose k too high, we can find more similar temporal behaviors for the terms but we will not be able to find many co-occurences in the same cluster.

1.3

For each cluster we want to find terms used in the same context. I built a graph where terms are the nodes. The number of co-occurences of two terms cannot overcome the frequency of the less used term. So, as a threshold to decide whether to put an edge between two terms, I used the minimum frequency of the two terms times 0.5. This means that the less used term appears more than 50% of times near to the other term.
I then searched for all the connected components of the graph. In this way I am excluding terms which do not frequently appear in the same tweets with any other term of the same cluster. Now we want to find groups of terms homogenously connected. If the connected component is composed by only two terms, we can be sure of it, otherwise a good way to ensure it can be to find the innermost core.

1.4

Previously we obtained groups of token, both for yes and for no users.
Now I set the grain to 3h and from the main index I checked up for each group of each cluster the number of tweets (containing terms of the considered group) distribution over time. I also saved text files for each component, for each interval (100 tweets for file).

```
YES CLUSTERS
cluster1
comp0:  iovotos freq:  66792 risparm freq:  12603
tfreq:  1133
```

People are talking about the savings for the state with the reform.

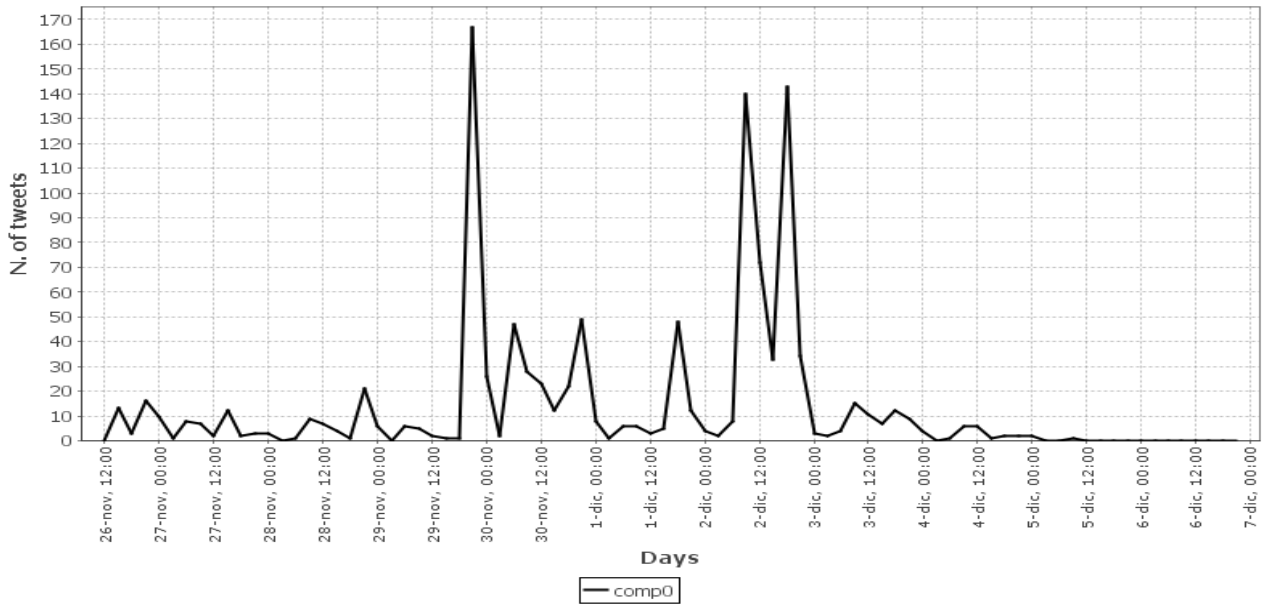

Figure 2: YES cluster1

```
cluster3
comp0:  mentan freq:  17196 enric freq:  8540 ancor freq:  251469
tfreq:  457
```

Matteo Renzi published his interview with Enrico Mentana on dicember the 2nd and many people retweeted it.

```
comp1:  insiem freq:  101951 voto freq:  148048
tfreq:  12201
```

It is again about the referendum.

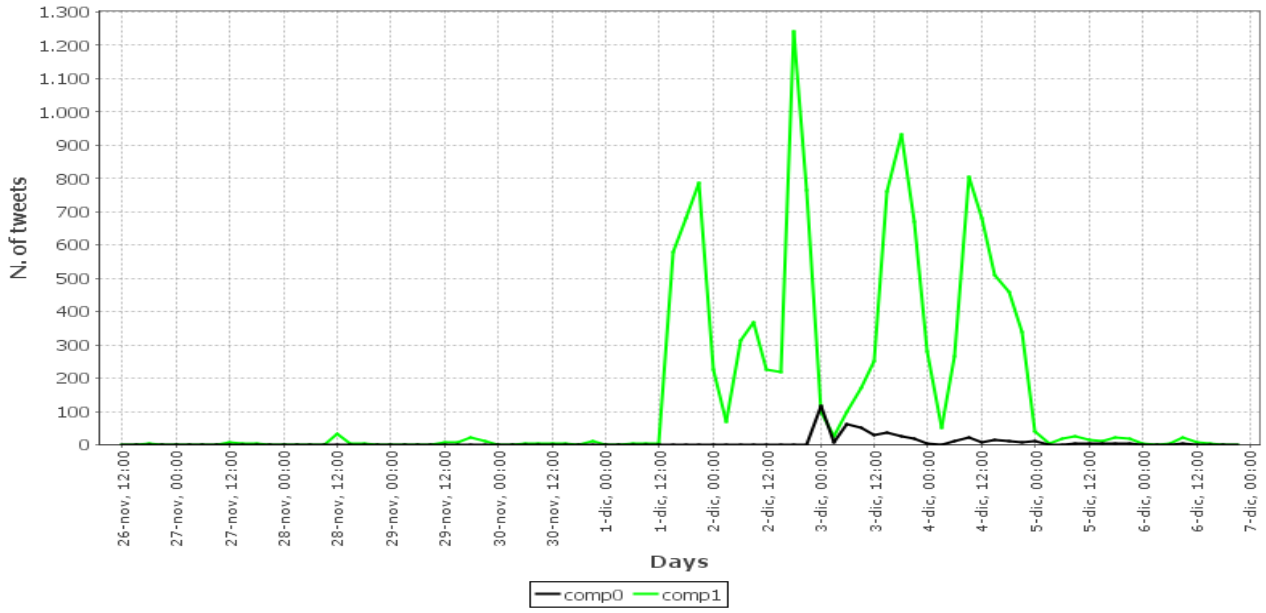

Figure 3: YES cluster3

```
cluster4
comp0:  paoloall freq:  241 maurizio_lup freq:  931
tfreq:  138
```

On november the 27th Paolo Alli, Maurizio Lupi and Maria Elena Boschi are in Legnano to talk about the referendum.

```
comp1:  violenz freq:  23415 donne freq:  65864
tfreq:  10887
```

On november the 26th there was the day against violence against women.

```
comp2:  castr freq:  66881 fidel freq:  83777
tfreq:  54238
```

On november the 25th Fidel Castro died.

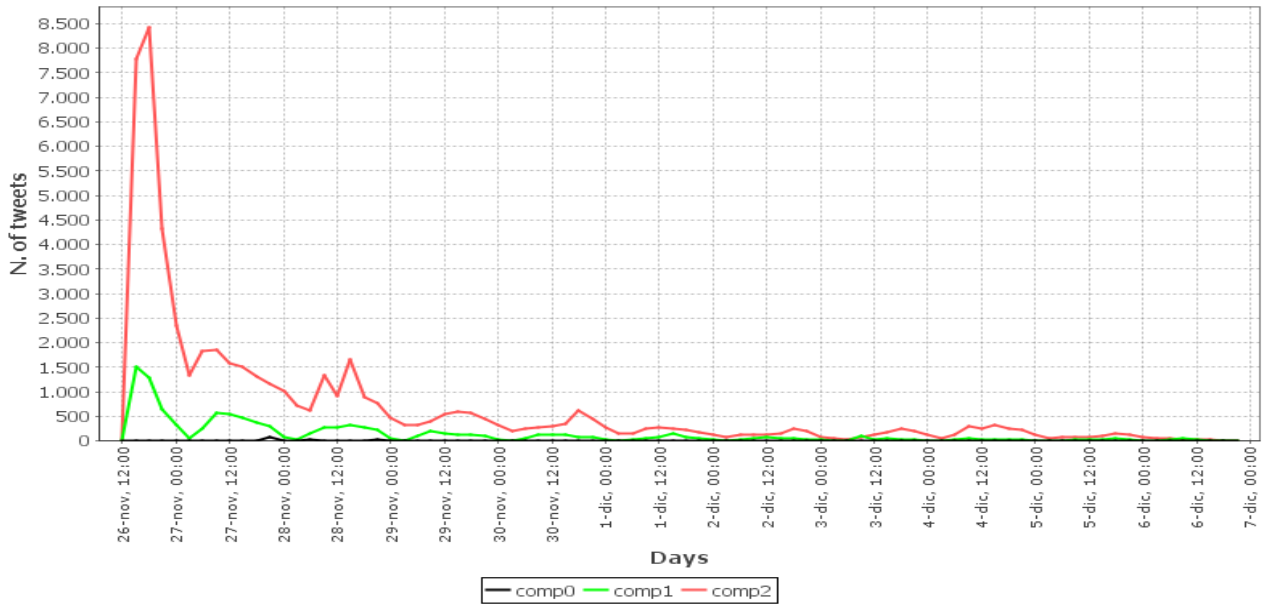

Figure 4: YES cluster4

```
cluster5
comp0:  kcgfbwnnrh freq:  2298 slide freq:  4055 belli freq:  30527 singol freq:  23949
tfreq:  2298
```

Matteo Renzi published some slides after the referendum.

```
comp1:  ieri freq:  78601 iniziat freq:  31282 ripartiam freq:  2129
tfreq:  808
```

After the referendum Luca Lotti tweeted some encouragements for Renzi.

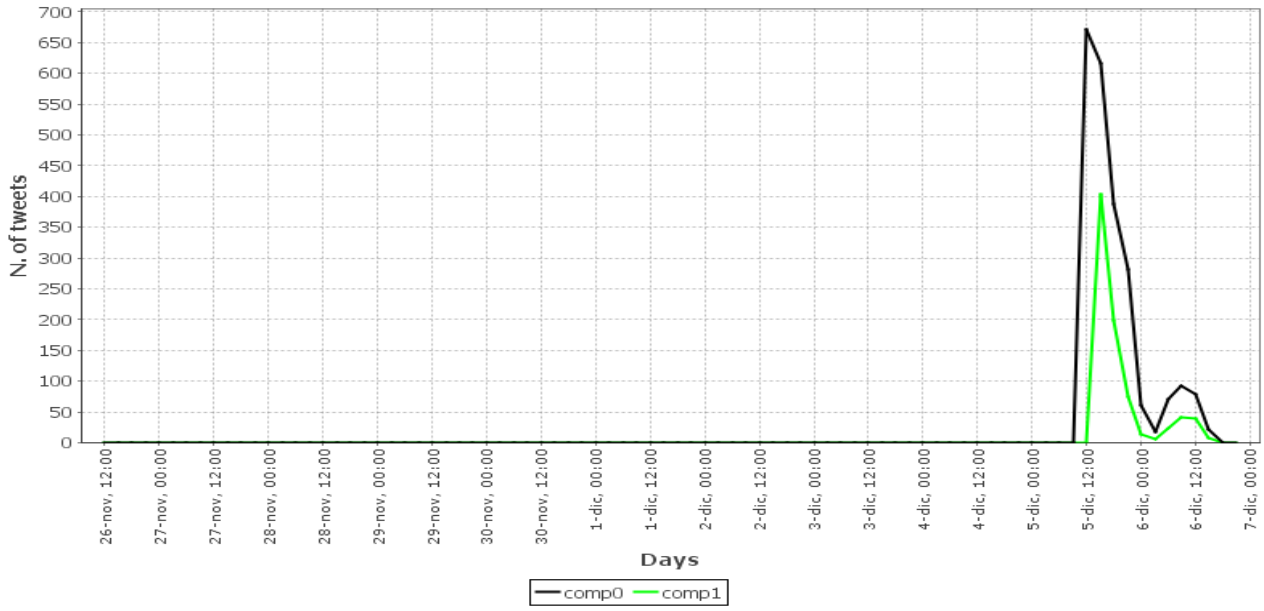

Figure 5: YES cluster5

```
cluster6
comp0:  gabanell freq:  7952 report freq:  12746
```

```
tfreq:  4778
```

On november the 28th Milena Gabanelli quit Report.

```
comp1:  mannin freq:  3337 nuti freq:  4309
tfreq:  2512
```

Mannino and Nuti from M5S had some problems with false signatures.

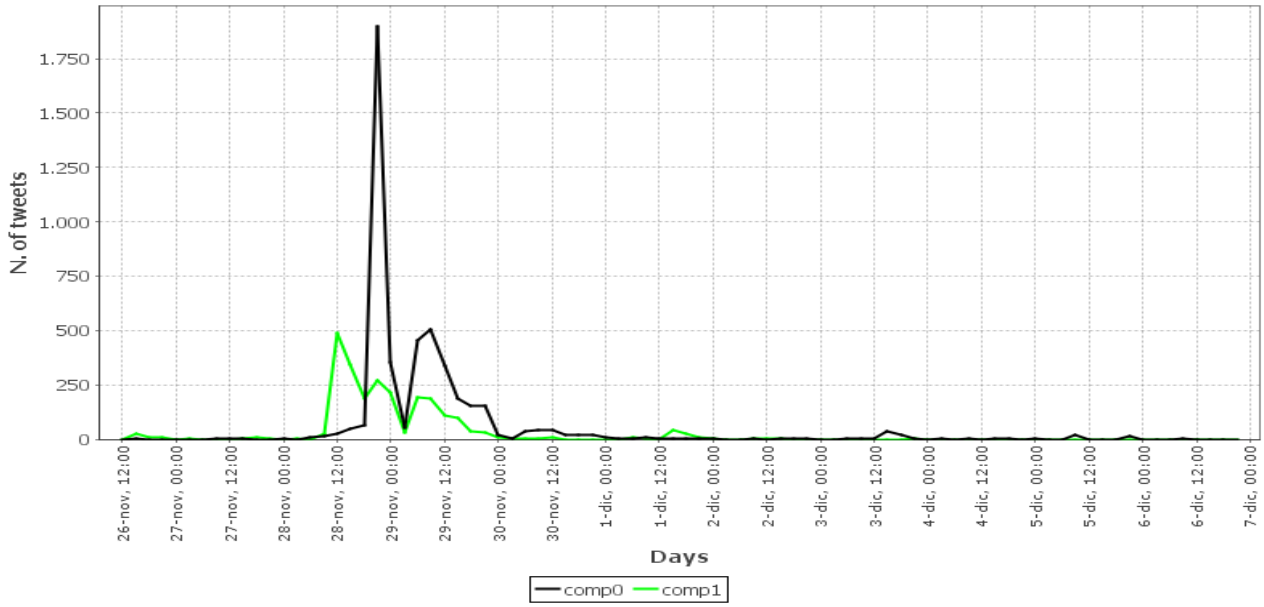

Figure 6: YES cluster6

```
cluster8
comp0:  matteosalvinim freq:  44083 terremotocentroital freq:  2831
tfreq:  319
```

On dicember the 1st Matteo Salvini was not present when the European Parliament decided to give funds for the italian earthquake.

```
comp1:  popolar freq:  19936 siren freq:  2244 abuso freq:  4382
tfreq:  524
```

I think that this tweet is against Renzi.

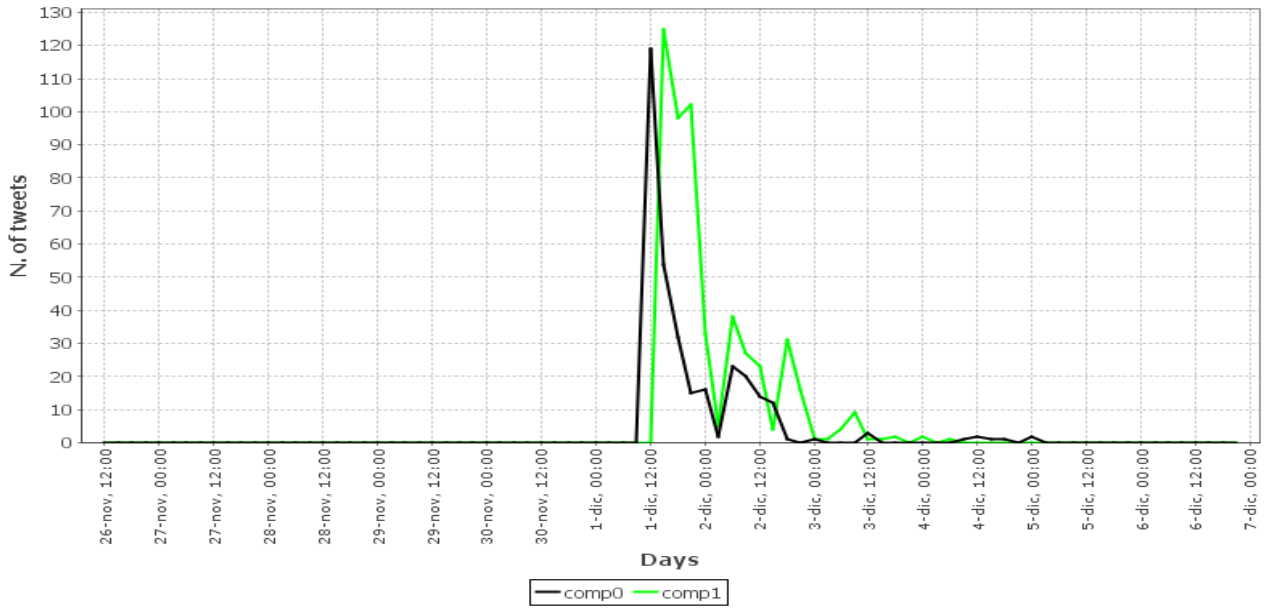

Figure 7: YES cluster8

```
cluster9
comp0:  mariaelenabozz1 freq:  219 yppi2011 freq:  364
tfreq:  162



comp1:  ulivo freq:  1679 prodi freq:  30743 roman freq:  22519
tfreq:  262
```

Romano Prodi supports yes.

```
comp2:  pubblic freq:  58649 mariannamad freq:  3832 contrattop freq:  1350 lavorator freq:
14551
tfreq:  514
```

It is about public administration wages.

```
comp3:  disoccupazion freq:  7743 ottobr freq:  7315
tfreq:  975
```

It is about unemployment.

```
comp4:  liberis freq:  715 liberi_s freq:  909
tfreq:  398
```

It is about referendum.

```
comp5:  chiar freq:  41985 sali freq:  3230 joyotmzlau freq:  1336 discors freq:  24529
tfreq:  1336
```

It is about a Matteo Renzi's video.

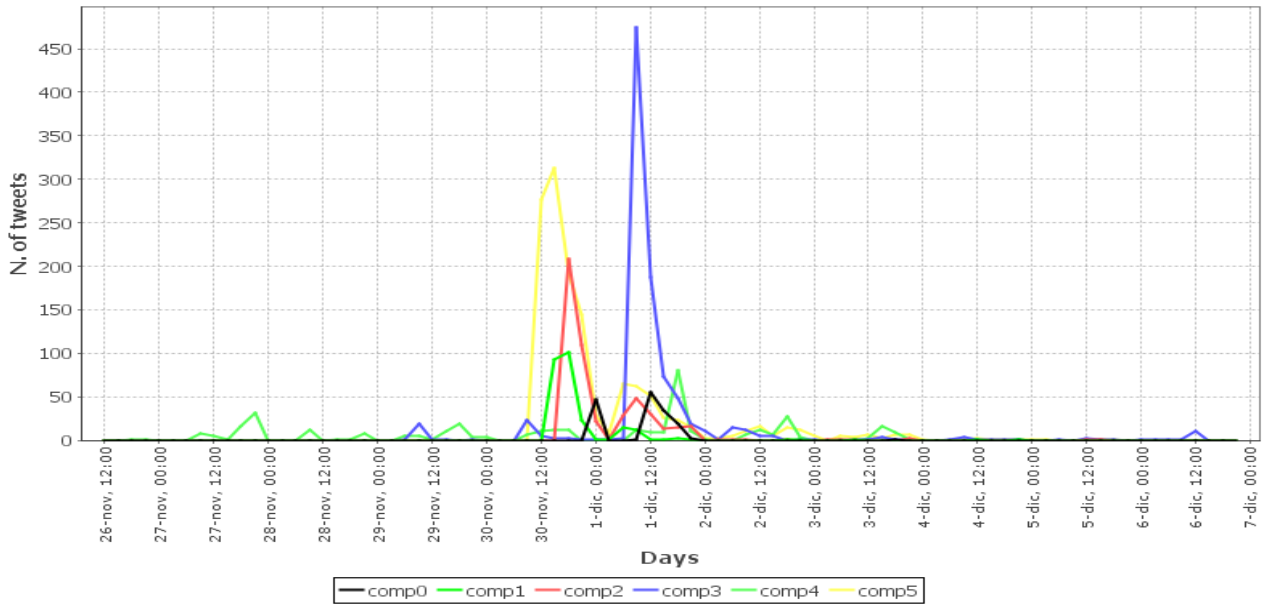

Figure 8: YES cluster9

```
cluster10
comp0:  comunqu freq:  131161 palazz freq:  27937 viva freq:  28518 minut freq:  77727 chigi
freq:  11511 qualc freq:  64923
tfreq:  5369
```

Again a Matteo Renzi's tweet.

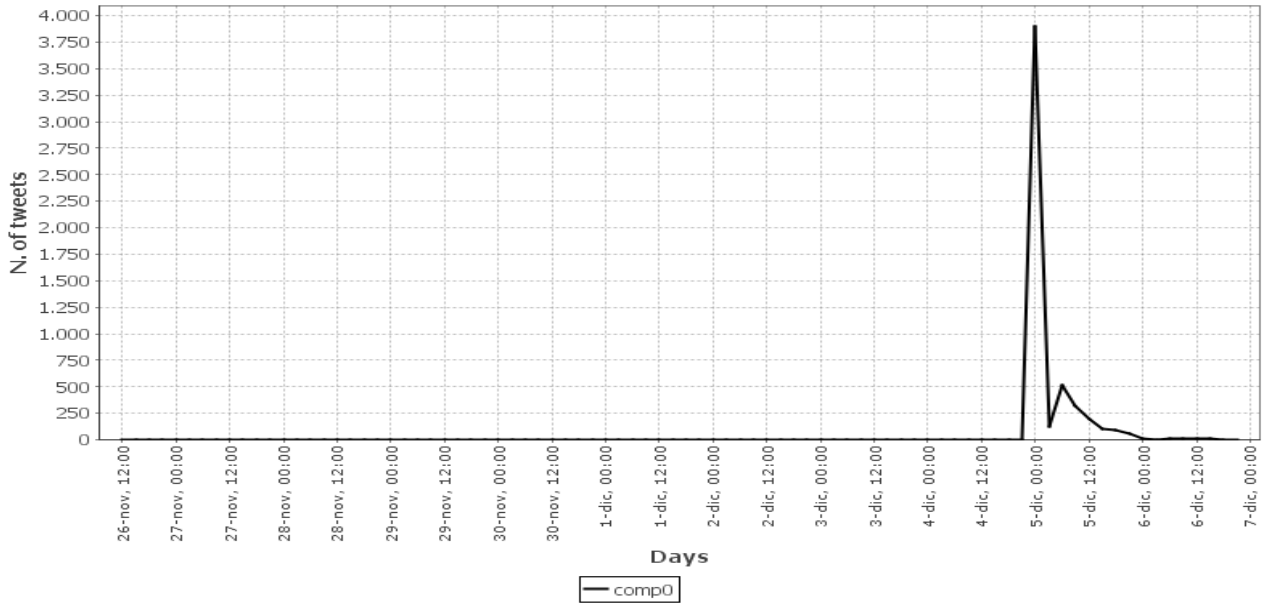

Figure 9: YES cluster10

```
cluster11
comp0:  bellen freq:  5339 austr freq:  19534 der freq:  8120 van freq:  10240
tfreq:  4386
```

Alexander Van der Bellen won in Austria.

```
comp1:  copiativ freq:  4637 matit freq:  47066 cancellabil freq:  11153
tfreq:  196
```

People are asking whether pencils are erasable.

```
comp2:  piero freq:  20272 pelù freq:  15446
tfreq:  11604
```

Piero Pelù against the referendum, "La Costituzione è NOstra".

```
comp3:  saggezz freq:  5206 segg freq:  16358
tfreq:  962
```

It is a retweet of Francesco Cundari.

```
comp4:  codacons freq:  1644 tota_mif freq:  163
tfreq:  28
```

```
comp5:  exit freq:  11922 poll freq:  10478
tfreq:  10182
```

Dopo Exit Poll voto all'estero e matite regolari #ReferendumCostituzionale? e Austria pure

```
comp6:  exitpoll freq:  4546 maratonamentan freq:  78042
tfreq:  1388
```

Enrico Mentana and exit poll.

```
comp7:  labbufal freq:  3550 sorpres freq:  25351
tfreq:  840
```

A tweet of Labbufala against Grillo.

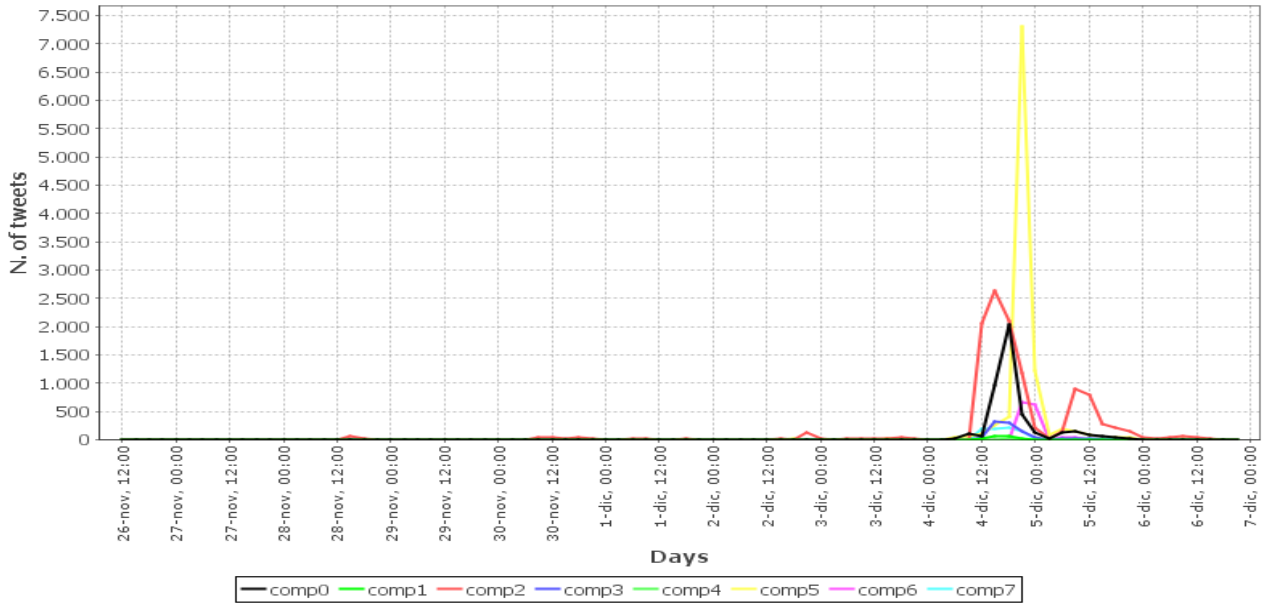

Figure 10: YES cluster11

```
cluster12
comp0:  lenelepr freq:  523 pacerenat freq:  324 bgrisaf freq:  728 deriuregin freq:  1060
vandab freq:  2115 orii151 freq:  529
tfreq:  152
```

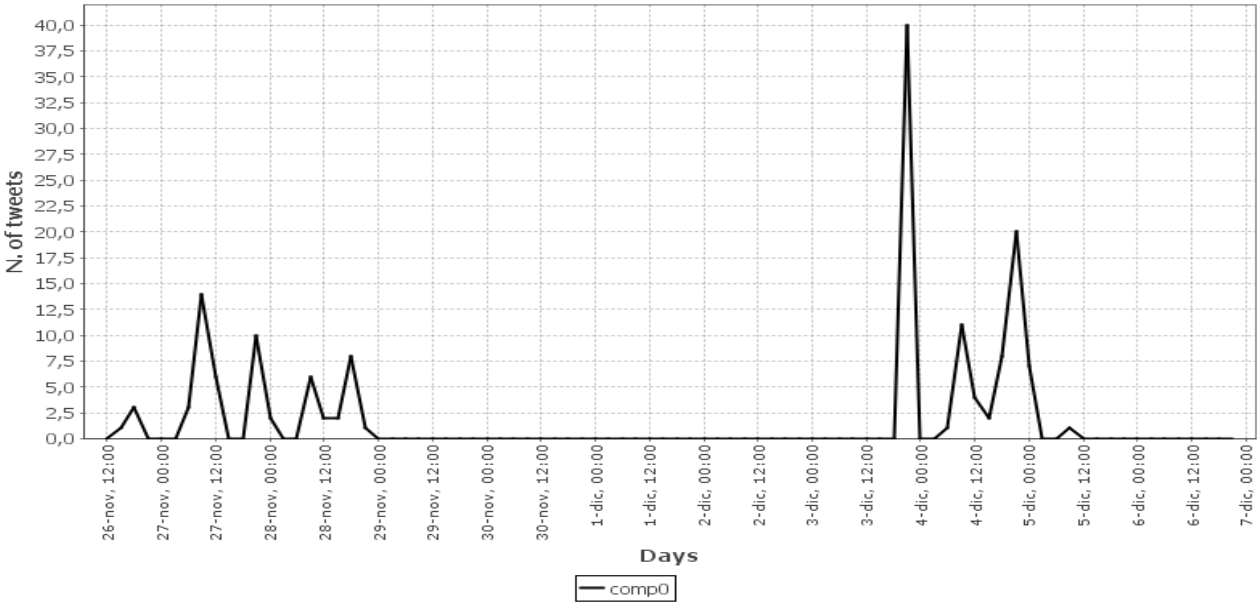

Figure 11: YES cluster12

```
cluster13
comp0:  cosedilavor freq:  711 nomfup freq:  2448
tfreq:  487
```

Again about unemployement.

```
comp1:  session freq:  3719 wpc16it freq:  1392
tfreq:  201
```

```
comp2:  matteorispond freq:  10943 bufaledeln freq:  1605
tfreq:  843
```

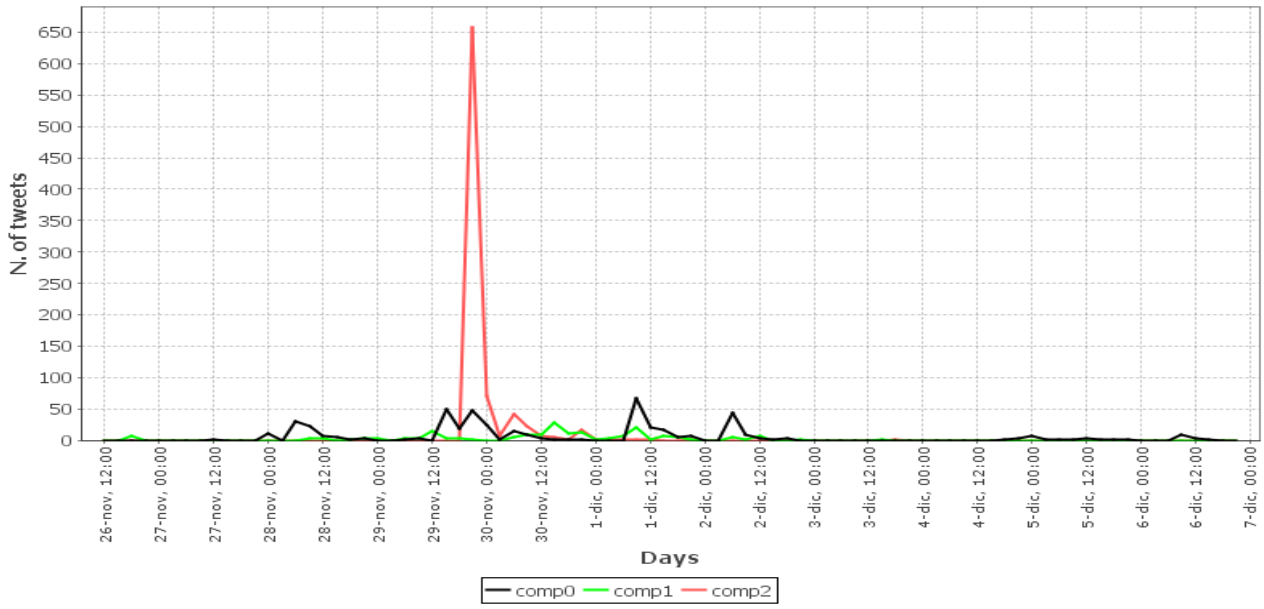

Figure 12: YES cluster13

```
cluster14
comp0:  giorn freq:  267496 mancan freq:  22081
tfreq:  5642
```

Here they talk about a lot of different things.

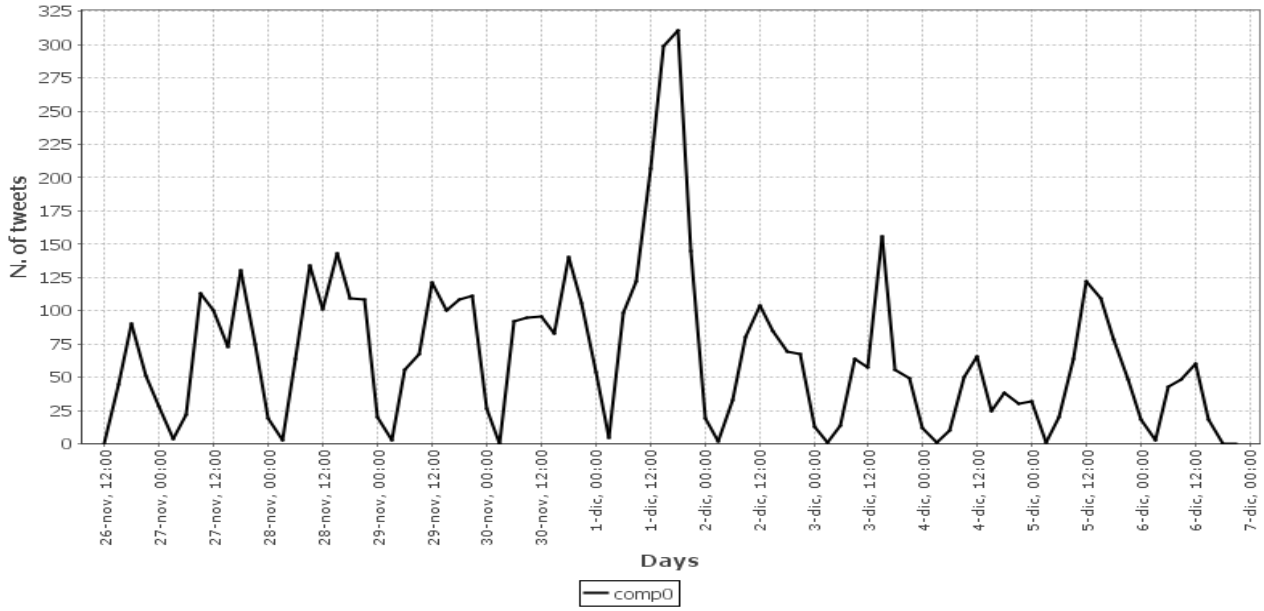

Figure 13: YES cluster14

```
cluster16
comp0:  cambiar freq:  59428 primo freq:  107405 possiam freq:  42396
tfreq:  9352
```

It is about referendum.

```
comp1:  silenzioelettoral freq:  13012 rompe freq:  4420 silenz freq:  43883
```

```
tfreq:  237
```

About referendum.

```
comp2:  chiusur freq:  9439 campagn freq:  40405 referendar freq:  12031
tfreq:  1001
```

About referendum.

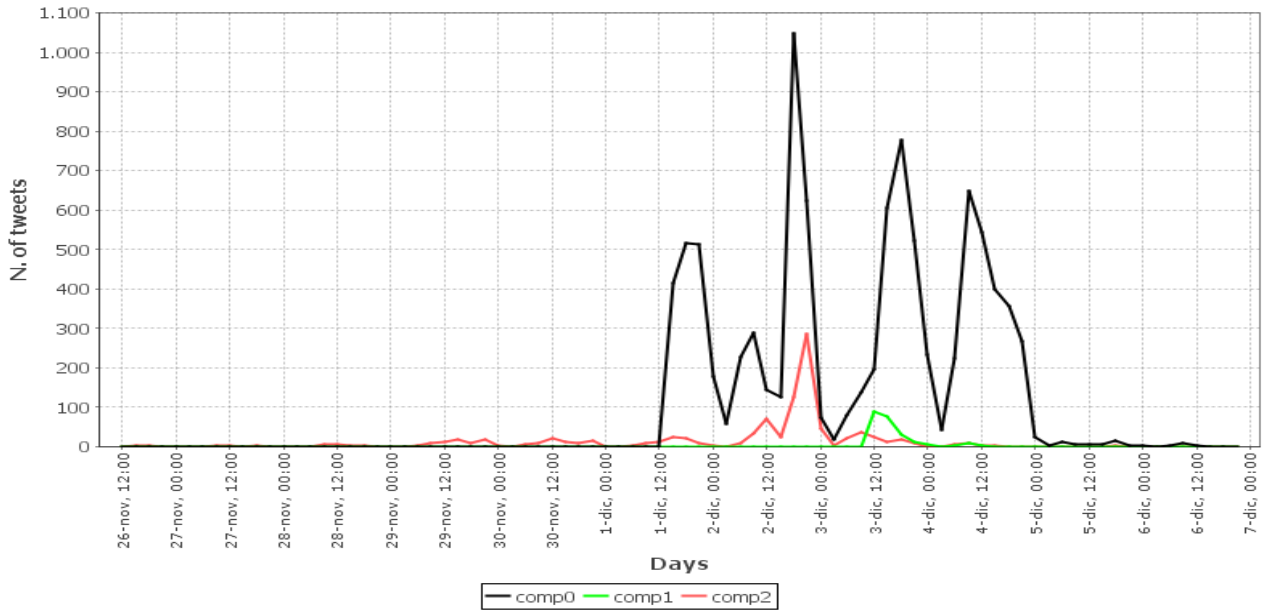

Figure 14: YES cluster16

```
cluster17
comp0:  regional freq:  14130 consiglier freq:  9085
tfreq:  3530
```

About referendum.

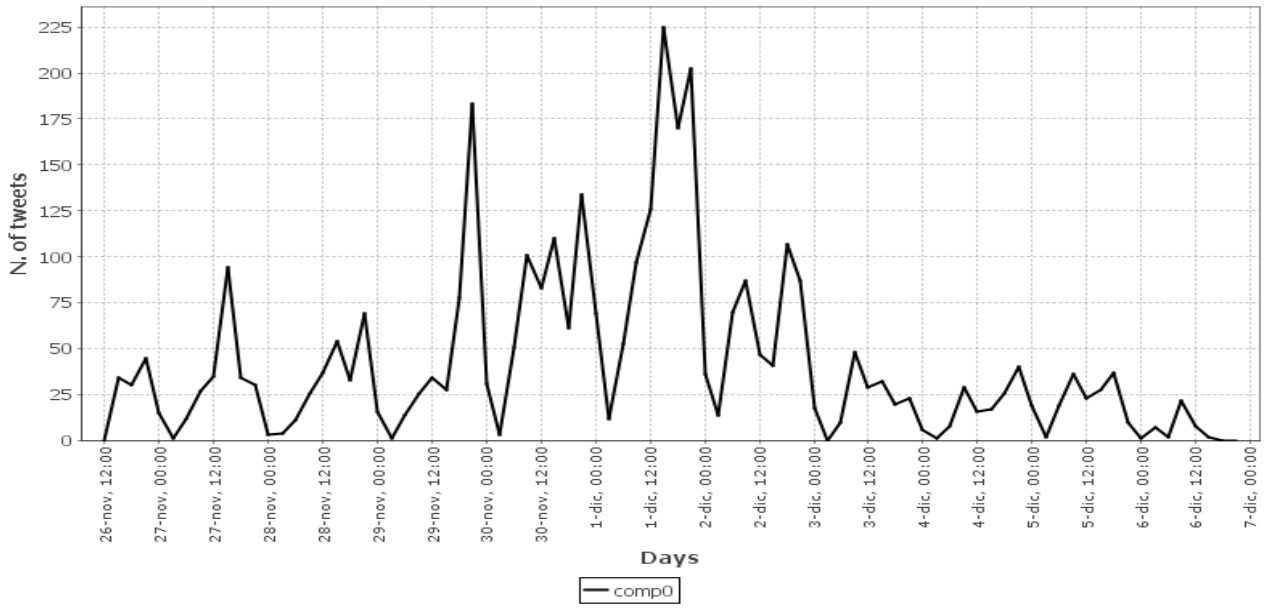

Figure 15: YES cluster17

```
NO CLUSTERS
cluster0
comp0:  grotond freq:  1020 rivcristian freq:  563
tfreq:  231
```

Rivista Cristiana and Gianfranco Rotondi.

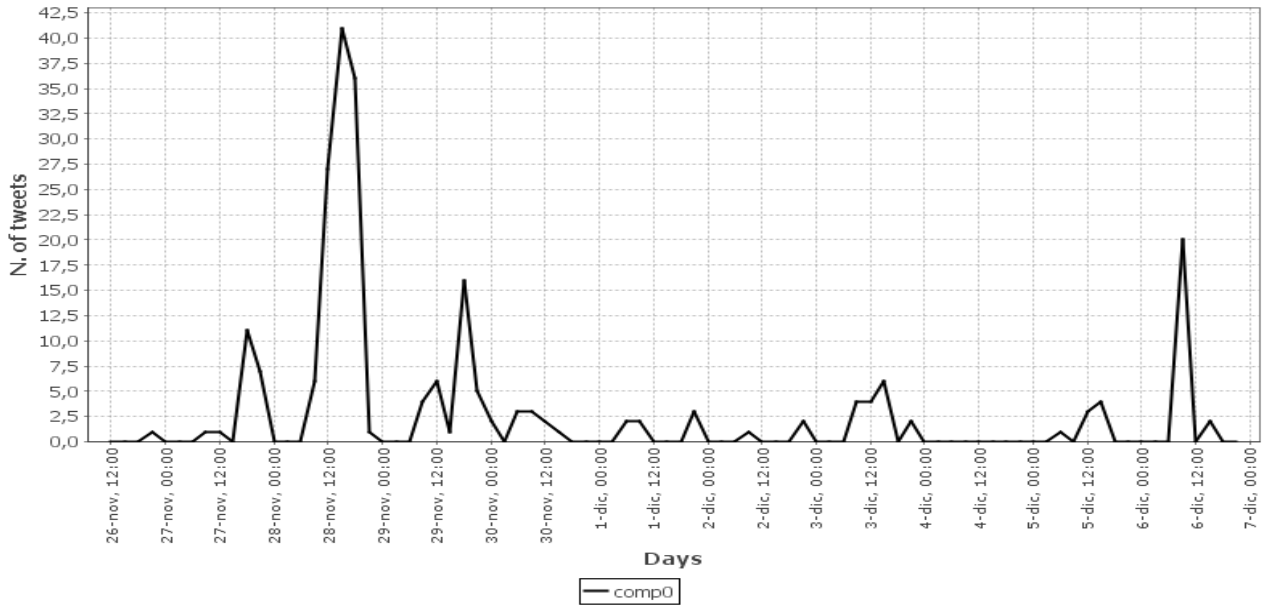

Figure 16: NO cluster0

```
cluster2
comp0:  tetto freq:  4886 dare freq:  39308
tfreq:  43
```

potreste dare visibilitá ai 21 licenziati che hanno occupato il tetto del San Camillo?

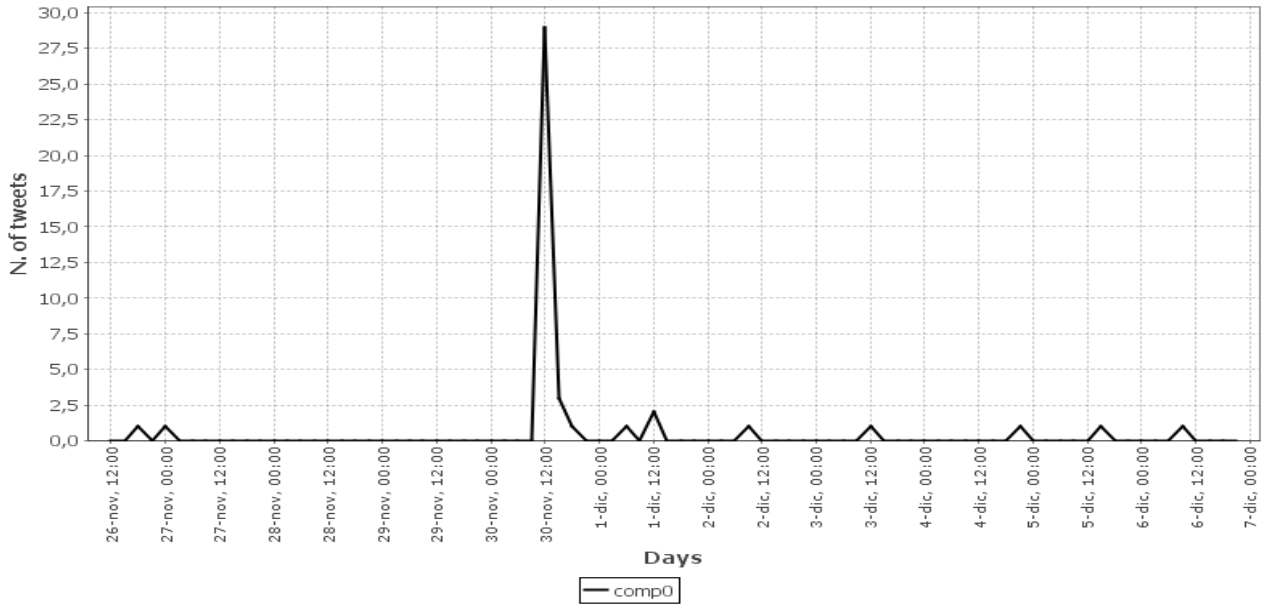

Figure 17: NO cluster2

```
cluster3
comp0:  regg freq:  11170 emil freq:  12910
tfreq:  4526
```

Reggio Emilia.

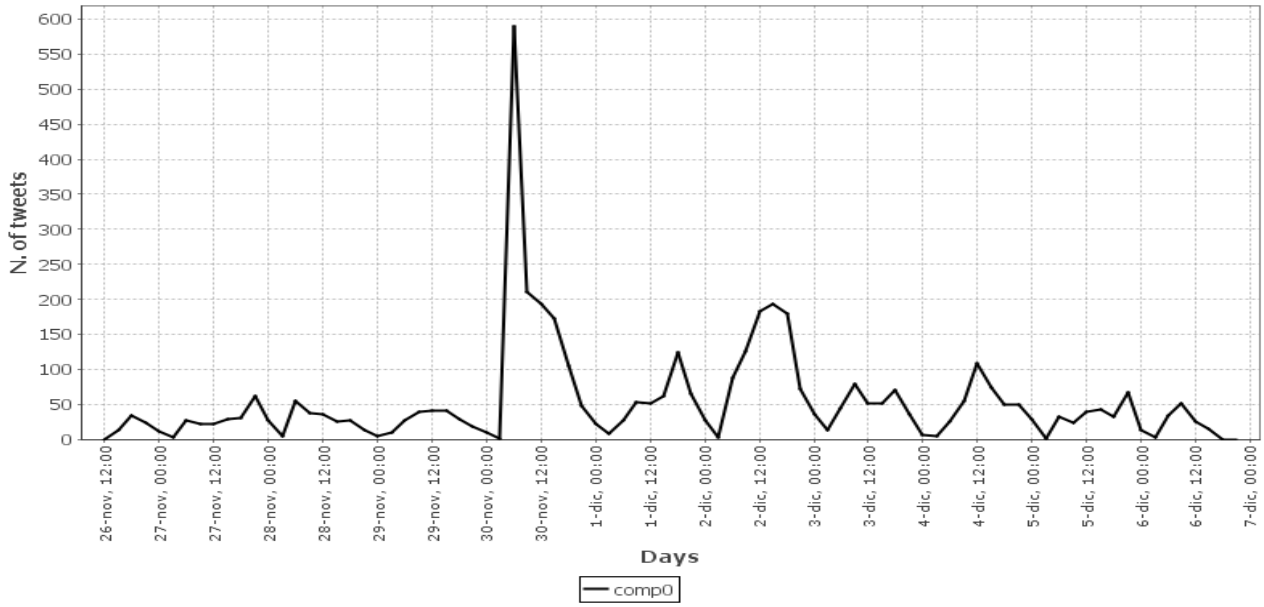

Figure 18: NO cluster3

```
cluster6
comp0:  giornalism freq:  4669 inchiest freq:  10244
tfreq:  288
```

It is about Milena Gabanelli.

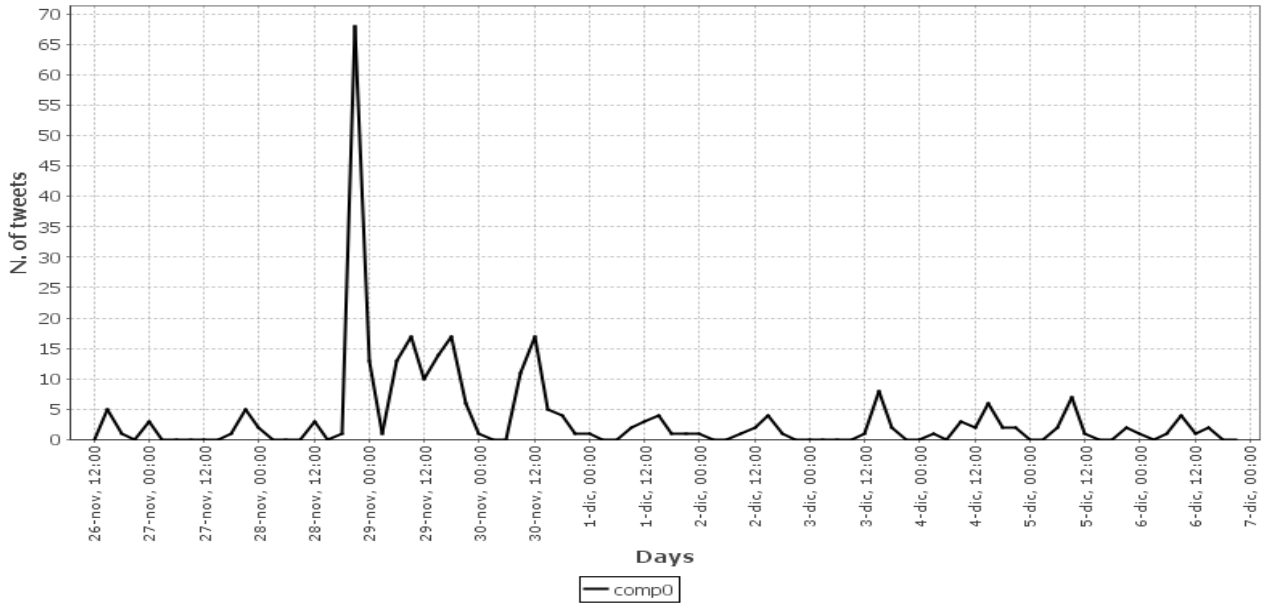

Figure 19: NO cluster6

This cluster is very similar to YES cluster10.

```
cluster7
comp0:  austr freq:  19534 der freq:  8120 van freq:  10240
tfreq:  4413
```

Alexander Van der Bellen won in Austria.

```
comp1:  exit freq:  11922 poll freq:  10478
tfreq:  10182
```

Dopo Exit Poll voto all'estero e matite regolari #ReferendumCostituizionale? e Austria pure.

```
comp2:  piero freq:  20272 pelù freq:  15446
tfreq:  11604
```

Piero Pelù against the referendum, "La Costituzione è NOstra".

```
comp3:  matit freq:  47066 cancellabil freq:  11153
tfreq:  10494
```

People are asking whether pencils are erasable.

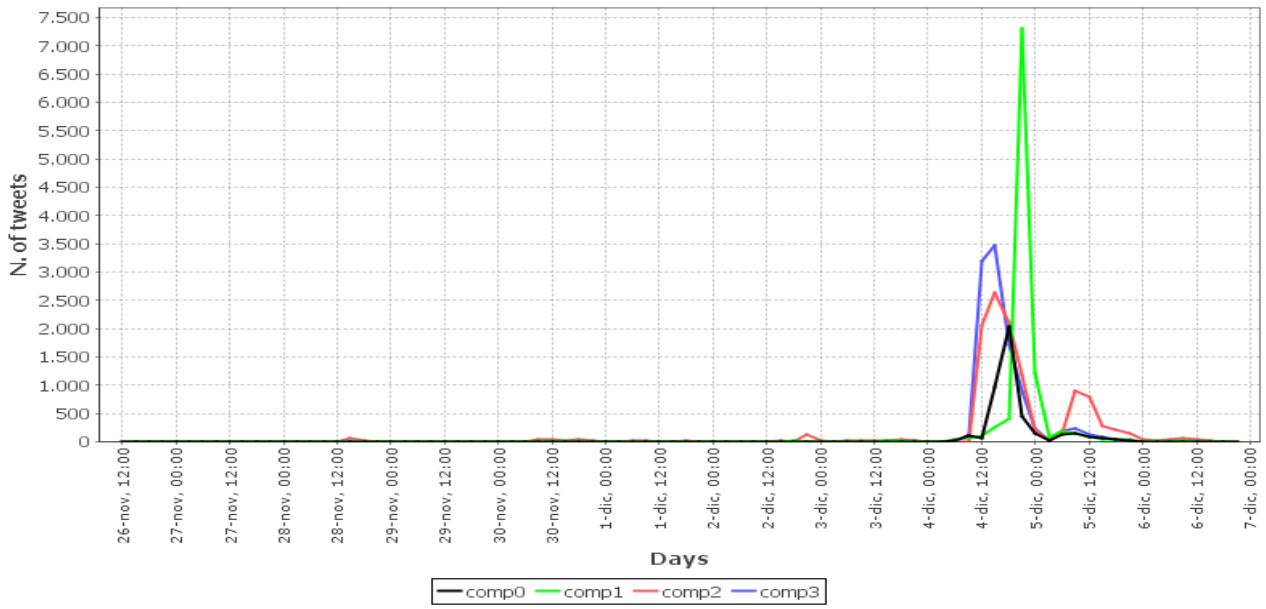

Figure 20: NO cluster7

```
cluster11
comp0:  chigi freq:  11511 palazz freq:  27937
tfreq:  11276
```

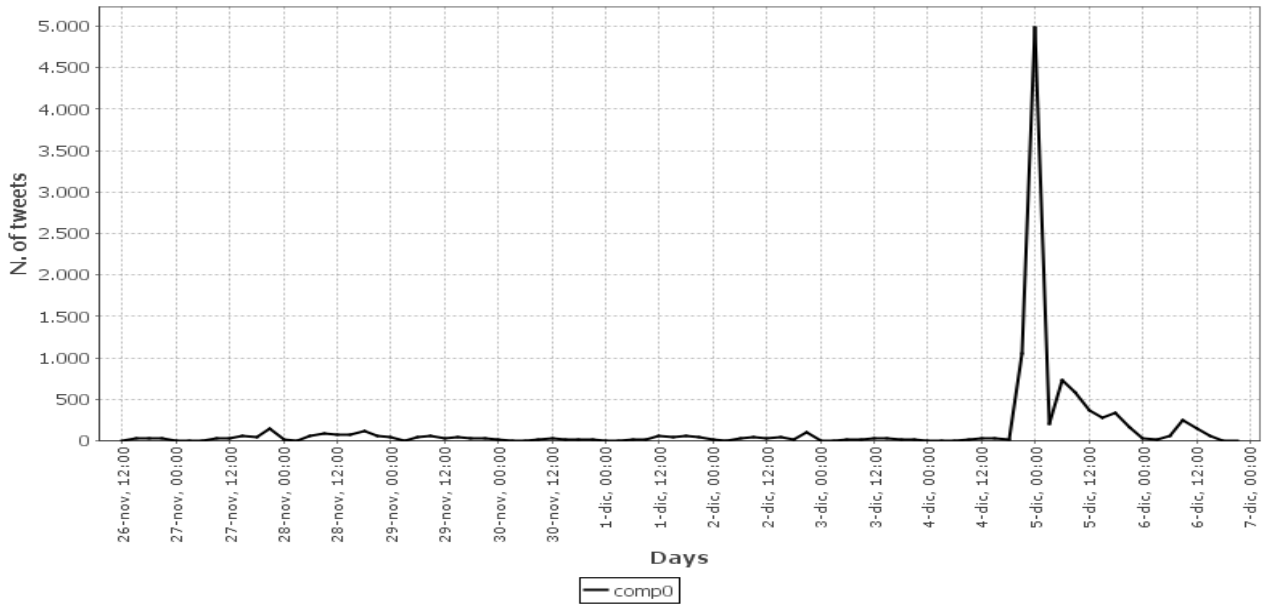

Figure 21: NO cluster11

```
cluster12
comp0:  contratt freq:  24206 rinnov freq:  12167
tfreq:  3408
```

About unemployement.

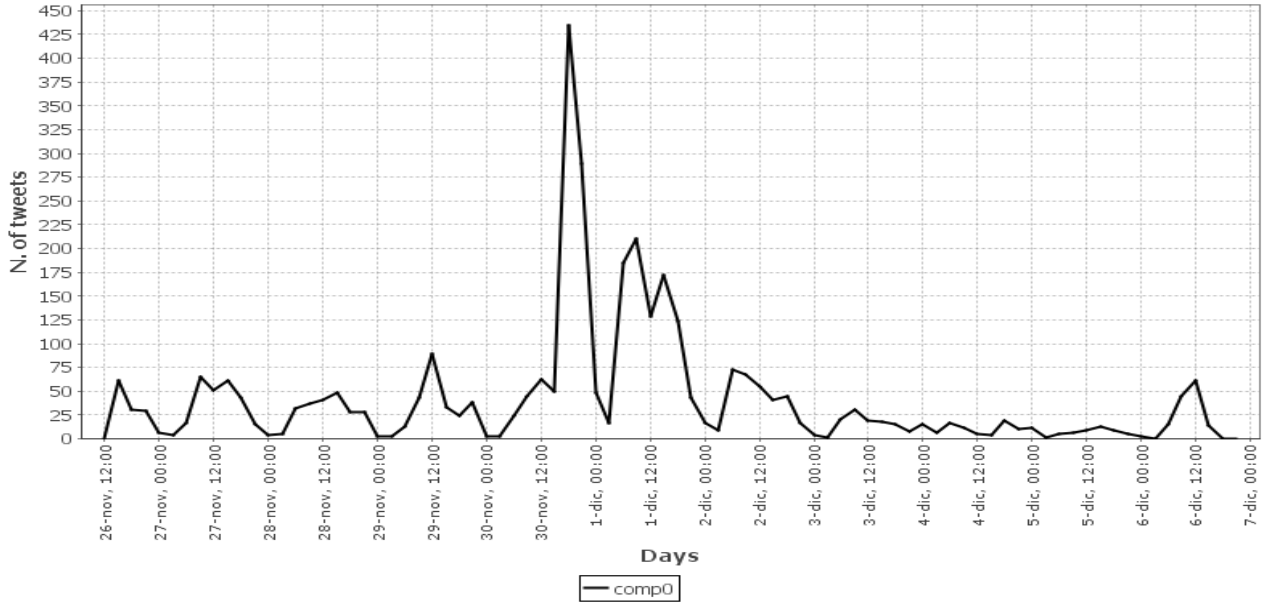

Figure 22: NO cluster12

```
cluster13
comp0:  donneinart freq:  5458 alecoscin freq:  5648
tfreq:  3236
```

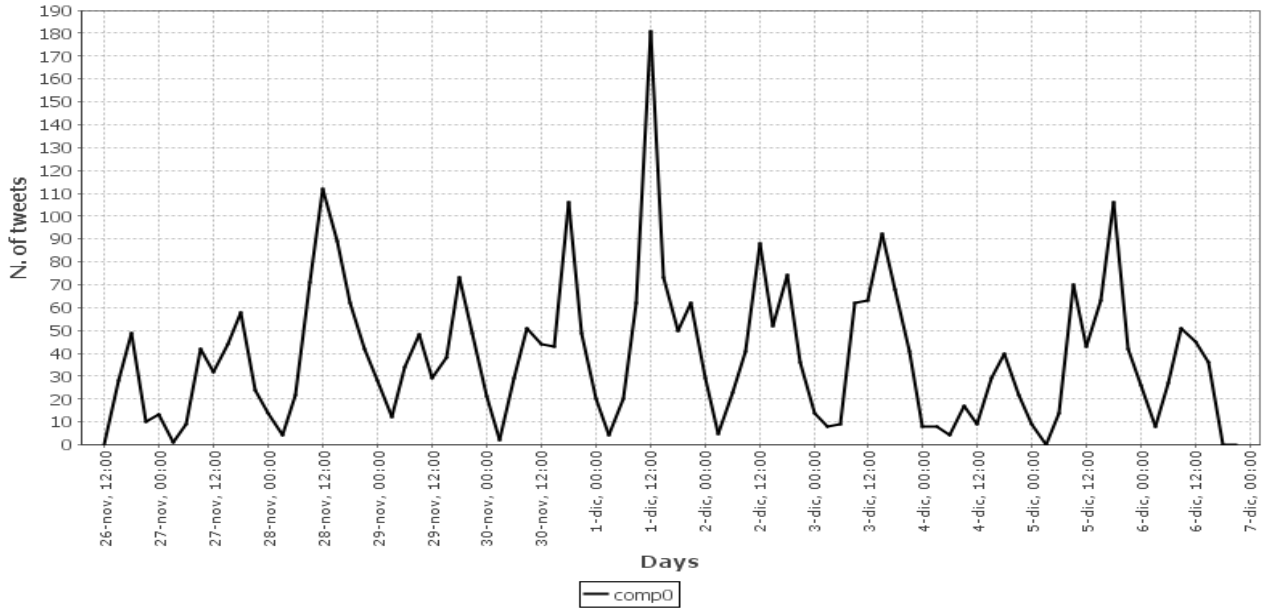

Figure 23: NO cluster13

```
cluster14
comp0:  donne freq:  65864 odproroga2018 freq:  4535 esclus freq:  6406 vbenetell freq:  2228
cristalizzazion freq:  1092 termin freq:  16627 speranz freq:  28526 chiediam freq:  6391 temporal
freq:  3105 o.d freq:  1098
tfreq:  931
comp1:  opzion freq:  7745 marialuisazocc1 freq:  2157 prorog freq:  6159
tfreq:  1144
```

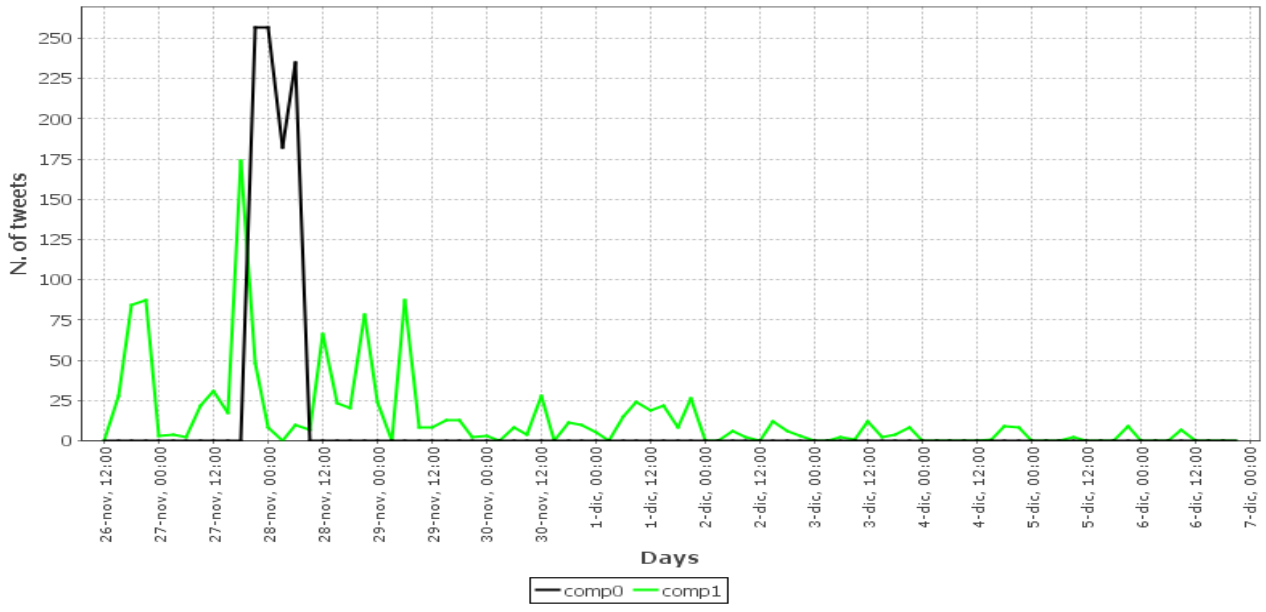

Figure 24: NO cluster14

```
cluster15
comp0:  pubblicat freq:  76201 nuova freq:  146707
tfreq:  52025
```

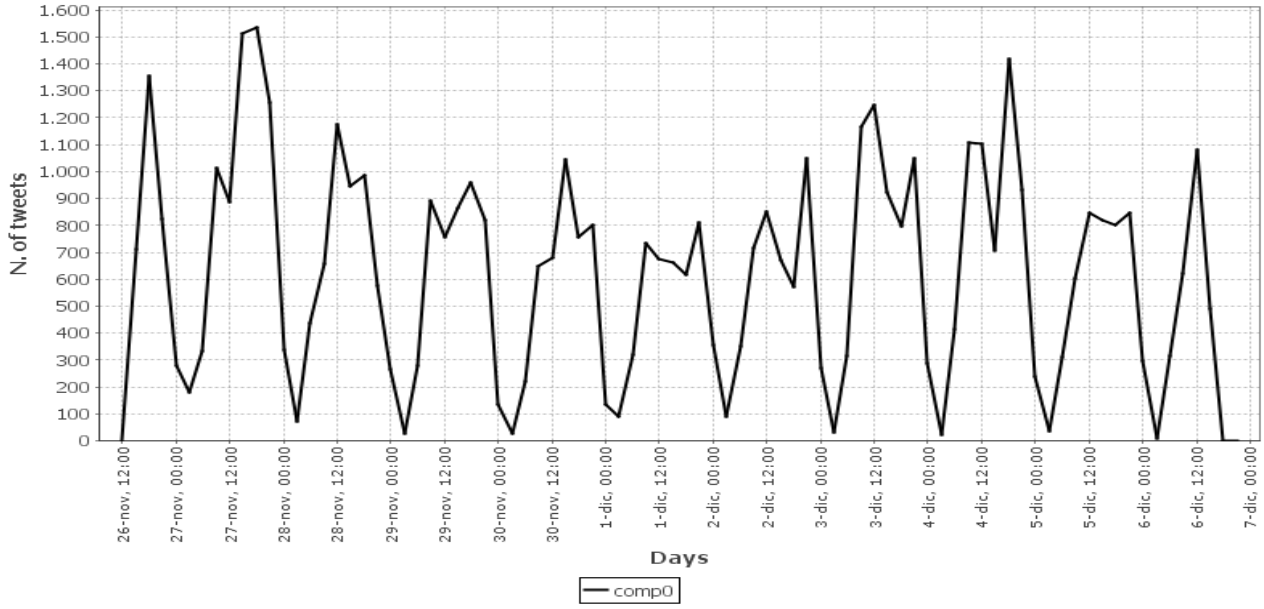

Figure 25: NO cluster15

```
cluster18
comp0:  cultur freq:  31234 librer freq:  7352 simonetoscano86 freq:  286
tfreq:  59
comp1:  berluscon freq:  54005 silv freq:  13758
tfreq:  3455
comp2:  iovoton freq:  228603 iodicon freq:  60142 5bc32772e3fb467 freq:  4278
tfreq:  2275
```

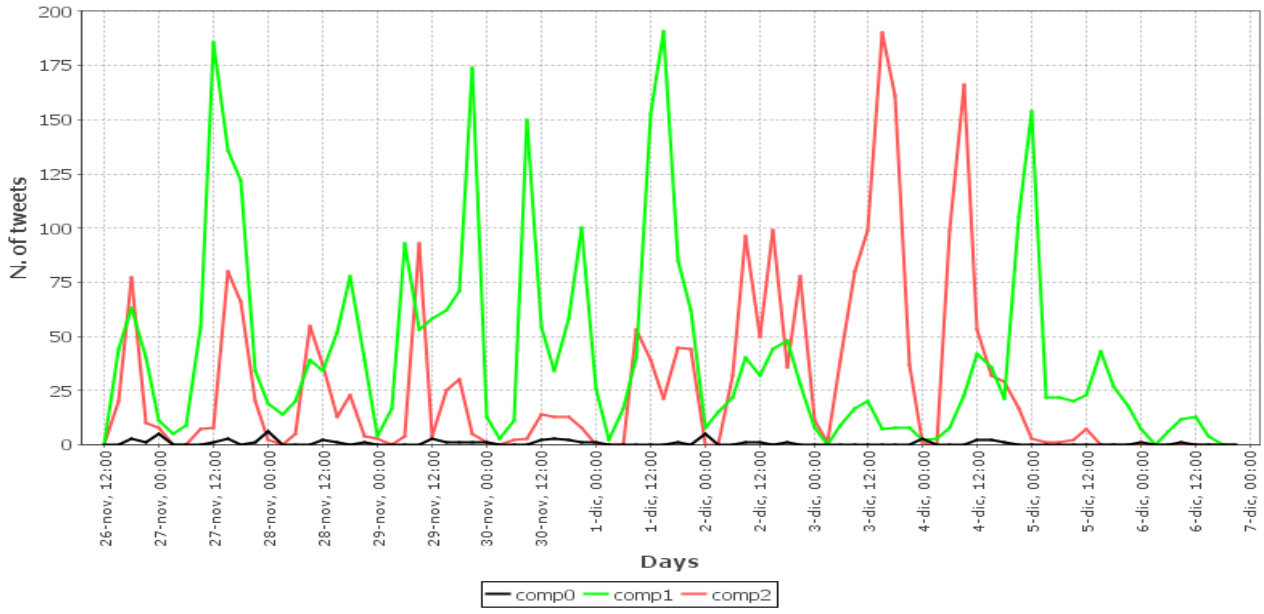

Figure 26: NO cluster18

1.5

I finally searched for hashtags containing "yes" or "no" and then manually selected the relevant ones.

2 Identify mentions of candidates or YES/NO supporter

2.1

From a sampled network of 450193 twitter users we want to find the YES or NO supporters.
For each user I counted the number of mentions of the previously found politicians/journalists for yes or no. Then I made the same search with the number of hashtags, and again with the number of tweets containing all the terms of the previously found groups of tokens. If at least one of this counts overcomes a threshold(=5), I computed the score:
$score = (nhashtags\_y * 2 + npoliticians\_y + ncomponents\_y) - (nhashtags\_n * 2 + npoliticians\_n + ncomponents\_n)$
If the score is bigger than 0, I assigned the user to the yes supporters, if it is smaller I assigned him to the no supporters.

```
M yes:  5737
M no:  10323
N. of yes tweets:  1367825
N. of yes tweets:  2272029
```

No supporters are at least two times of yes supporters.

2.2

Given the subgraph induced by the previously found users (both yes or no supporters), I found the largest connected component and computed HITS algorithm.

```
lcc size:  15075
```

Thanks to the two lists created before, I could classify the hubs and authorities in yes or no supporters.

```
N yes authorities:  179
N no authorities:  821
N yes hubs:  255
N no hubs:  745
```

authorities:

```
name:  pino habitat
screen name:  31239a839db94c9
followers:  1866
in-degree:  438
n mentions:  703
n hashtags-yes:  9
n hashtags-no:  83
n politicians-yes:  14
n politicians-no:  44
n components-yes:  6
n components-no:  11

name:  furio
screen name:  Giangiagainst
followers:  5027
in-degree:  502
n mentions:  682
```

n hashtags-yes: 1
n hashtags-no: 5
n politicians-yes: 2
n politicians-no: 10
n components-yes: 1
n components-no: 1

name: david
screen name: Didi1648
followers: 1188
in-degree: 225
n mentions: 14
n hashtags-yes: 0
n hashtags-no: 8
n politicians-yes: 2
n politicians-no: 2
n components-yes: 1
n components-no: 0

name: domenico formica
screen name: domenicoformic3
followers: 1746
in-degree: 306
n mentions: 26
n hashtags-yes: 1
n hashtags-no: 7
n politicians-yes: 1
n politicians-no: 0
n components-yes: 0
n components-no: 0

name: leonardo
screen name: leonardo1ao
followers: 2193
in-degree: 499
n mentions: 478
n hashtags-yes: 2
n hashtags-no: 26
n politicians-yes: 15
n politicians-no: 14
n components-yes: 1
n components-no: 0

hubs:
name: il fatto quotidiano
screen name: fattoquotidiano
followers: 1654598
in-degree: 55
n mentions: 34749
n hashtags-yes: 1
n hashtags-no: 20
n politicians-yes: 0
n politicians-no: 25

```
n components-yes:  18
n components-no:  17

name:  matteo renzi
screen name:  matteorenzi
followers:  2754423
in-degree:  29
n mentions:  128706
n hashtags-yes:  14
n hashtags-no:  0
n politicians-yes:  5
n politicians-no:  0
n components-yes:  11
n components-no:  1

name:  beppe grillo
screen name:  beppe_grillo
followers:  2244415
in-degree:  19
n mentions:  34293
n hashtags-yes:  3
n hashtags-no:  159
n politicians-yes:  4
n politicians-no:  23
n components-yes:  1
n components-no:  0

name:  la repubblica
screen name:  repubblicait
followers:  2506673
in-degree:  5
n mentions:  81986
n hashtags-yes:  11
n hashtags-no:  11
n politicians-yes:  19
n politicians-no:  8
n components-yes:  160
n components-no:  85

name:  sky tg24
screen name:  SkyTG24
followers:  2636921
in-degree:  10
n mentions:  19808
n hashtags-yes:  7
n hashtags-no:  25
n politicians-yes:  10
n politicians-no:  0
n components-yes:  27
n components-no:  9
```

2.3

I now created a root of 100 users supporting yes and a root of 100 users supporting no.
The yes root is composed by the users with the higher yes score (computed as said before). Given the root, I also included in the graph users in the yes list connected to the root users. I finally computed HITS on this graph.
I did the same for users supporting no.

yes authorities:
name: carla dossi
screen name: DossiCarla
followers: 4229
n hashtags-yes: 79
n hashtags-no: 3
n politicians-yes: 73
n politicians-no: 18
n components-yes: 12
n components-no: 5

name: renzo seccia
screen name: Renzoseccia
followers: 3446
n hashtags-yes: 49
n hashtags-no: 3
n politicians-yes: 52
n politicians-no: 22
n components-yes: 10
n components-no: 4

name: danielecina
screen name: danielecina
followers: 10892
n hashtags-yes: 24
n hashtags-no: 3
n politicians-yes: 56
n politicians-no: 11
n components-yes: 37
n components-no: 5

name: emiliano liberati
screen name: ELiberati
followers: 2287
n hashtags-yes: 94
n hashtags-no: 0
n politicians-yes: 17
n politicians-no: 2
n components-yes: 12
n components-no: 1

name: laura cesaretti
screen name: lauracesaretti1
followers: 6572
n hashtags-yes: 15
n hashtags-no: 2

n politicians-yes: 43
n politicians-no: 9
n components-yes: 7
n components-no: 3

yes hubs:
name: matteo renzi
screen name: matteorenzi
followers: 2754423
n hashtags-yes: 14
n hashtags-no: 0
n politicians-yes: 5
n politicians-no: 0
n components-yes: 11
n components-no: 1

name: partito democratico
screen name: pdnetwork
followers: 200416
n hashtags-yes: 203
n hashtags-no: 1
n politicians-yes: 267
n politicians-no: 4
n components-yes: 46
n components-no: 6

name: la repubblica
screen name: repubblicait
followers: 2506673
n hashtags-yes: 11
n hashtags-no: 11
n politicians-yes: 19
n politicians-no: 8
n components-yes: 160
n components-no: 85

name: deputati pd
screen name: Deputatipd
followers: 57302
n hashtags-yes: 248
n hashtags-no: 0
n politicians-yes: 339
n politicians-no: 6
n components-yes: 40
n components-no: 4

name: graziano delrio
screen name: graziano_delrio
followers: 193037
n hashtags-yes: 14
n hashtags-no: 0
n politicians-yes: 19
n politicians-no: 3

n components-yes: 2
n components-no: 0

no authorities:
name: oinot49
screen name: oinot49
followers: 1634
n hashtags-yes: 25
n hashtags-no: 153
n politicians-yes: 56
n politicians-no: 52
n components-yes: 1
n components-no: 5

name: pino habitat
screen name: 31239a839db94c9
followers: 1866
n hashtags-yes: 9
n hashtags-no: 83
n politicians-yes: 14
n politicians-no: 44
n components-yes: 6
n components-no: 11

name: leonardo
screen name: leonardo1ao
followers: 2193
n hashtags-yes: 2
n hashtags-no: 26
n politicians-yes: 15
n politicians-no: 14
n components-yes: 1
n components-no: 0

name: furio
screen name: Giangiagainst
followers: 5027
n hashtags-yes: 1
n hashtags-no: 5
n politicians-yes: 2
n politicians-no: 10
n components-yes: 1
n components-no: 1

name: sandro&maria
screen name: SandroPop1
followers: 1501
n hashtags-yes: 2
n hashtags-no: 17
n politicians-yes: 2
n politicians-no: 9
n components-yes: 1
n components-no: 0

no hubs:
name: beppe grillo
screen name: beppe_grillo
followers: 2244415
n hashtags-yes: 3
n hashtags-no: 159
n politicians-yes: 4
n politicians-no: 23
n components-yes: 1
n components-no: 0

name: il fatto quotidiano
screen name: fattoquotidiano
followers: 1654598
n hashtags-yes: 1
n hashtags-no: 20
n politicians-yes: 0
n politicians-no: 25
n components-yes: 18
n components-no: 17

name: lello esposito
screen name: LelloEsposito5
followers: 92540
n hashtags-yes: 23
n hashtags-no: 192
n politicians-yes: 125
n politicians-no: 103
n components-yes: 33
n components-no: 18

name: ?.friendsm5s.com
screen name: MPenikas
followers: 46121
n hashtags-yes: 39
n hashtags-no: 922
n politicians-yes: 104
n politicians-no: 692
n components-yes: 62
n components-no: 33

name: luigi di maio
screen name: luigidimaio
followers: 147034
n hashtags-yes: 0
n hashtags-no: 9
n politicians-yes: 0
n politicians-no: 1
n components-yes: 1
n components-no: 0

3 Spread of Influence

Finally we want to estimate the spread of influence of the two parties using the Label Propagation Algorithm. We have to first assign a label to each node of the graph (users):

0: if we do not know the party of the user.
1: if he is a yes supporter.
2: if he is a no supporter.

initList:
list is the list containing all the nodes which we want to change the label.
If the label is different from 0, we already know their label and we do not need to add the node to the list. We do not need to add a node also if we already know that it has no other nodes that can influence it (indegree $= 0$).

bestLabel:
First of all we do not want 0 labels to propagate, so we have to remove them from the array of the neighbors.
If the length of the new array is 0, we can only return the 0 label.
If there is only one element left, it will return it.
Otherwise we can start counting the occurences of the labels.
maxCount contains the label occured more frequently.
If two labels have the same frequency, the algorithm will chose the new best randomly.
If $(maxCount - counter) > (nneighborhood.length - i)$ we can be sure that the last best label is the label that we are searching for.

In the first case, M is divided in 5737 yes supporters and 10323 no supporters. And the results of the LPA are:

```
YES: 357176
NO: 65848
```

Obviously the results can change a little bit with the order of the list, which is random.

In the second case I chose the best 500 hubs for yes supporters and 500 for no supporters. And the results are:

```
YES: 253306
NO: 168739
```