Graph techniques for micro blog analysis

Philip Fraisse

***Abstract*—"what are we talking about on tweeter?". To answer this question we try to provide, a concise information about the topics of tweets, their trends, and their likeliness to become a buzz.**
The semantic and outbreak detection part will be dealing with graph techniques. The results must be made available efficiently. As the volume of data can be hard to work, we will first sample the micro-blogs over the desired period of time.
As the analysis of a full graph built on the index of the corpus can be computationally intense, we filter the dictionary with different methods : Latent Dirichlet Allocation, and "outlying occurrence" during one day of the period. This first filter will be used to create a seed list of words. We then use a word to vector method to create a vector space on the dictionary. The seed list is expanded by taking the most cosine similar words to it. Since we will use unsupervised graph clustering methods to find topics in our micro-blogs, we try here to shape some kind of structure in the data, to make it easier for the clustering algorithm to read it. As there is a lot of noise in micro-blogs, we try here as well to filter all the un-relevant information before providing it to the unsupervised methods. We will then try different measures to look for a word contained in the topic, that is the most likely to resume it.
Further on in a second part, we try to improve the resume of the topics by using an "agnostic" knowledge graph on which we will project our topics and try to bring new external relevant information, to make the topics more understandable. Then, to increase the understanding of the results we will try to plot the trends of these topics to identify the period which this topic is related. Finally we try to tell where in the user graph, those arising trending topics are more likely to emerge.

I. Introduction

Automatic comprehension of large corpus of documents has been the quest of data scientists for years.
The goal to achieve can go from grammatical comprehension, to ontology awareness, event detection, automated and intelligible summaries, topic modeling and more.

Many companies would like to be more aware of how is perceived their brand by the customers. What brand feature is relevant to customers? How did their perception evolve over time. Is it possible to react the sooner as possible to an emerging bad buzz?
Social media offers a wide range of customers expression and feelings. Some of them can be useful to address these questions.
The aim of this project will be to bring a method to provide results in a way that is: Efficient, concise, accessible, understandable by non experts, visual and operational.
The method will use graph theory to structure the underlying semantics of twitter micro-blogs.
We will see how to improve the result by implementing a knowledge graph, to add external descriptive information.
The results will be analysed temporally, to see emerging trends in the topics
Relationships between users could also be used to weight the users actions, and detect potential emerging bad buzz.
The results will provide simple insights on what are the trending topics, potentially related to an event. What are the inner interactions between components of these topics, how do those topic interacts with each other, when are they likely to occur.
Overall, the aim is to outline a general method, which covers all parts from filtering to clustering, in order to be able to visualize a whole process and identify the parts that could be improved or which could be interesting to focus on.

II. Semantic Analysis

A. Methodology

1) Motivation: The aim of the work is to produce concise and relevant topics, from micro-blogs.
We identified two main risks : the fact that the semantic graph based on this data can be challenging to analyse, semantically and computationally, and the fact that since we will build topics by clustering this very noisy graph in an unsupervised way, we have no guarantee on what kind of information will emerge from these clusters.

To address both of these issues, we will pre-filter the data. We will compare two ways of doing it : with a Latent Dirichlet Allocation on the whole period, and with a custom metric to measure how unusual is the term occurrences on a specific day versus the period. Since LDA produces topics that are not exclusive to words, we can not use it directly to build our semantic graph. Nevertheless, we will use it to reduce the size of the graph that will be built on top of it. The aim is to filter noise, and to pre-shape the data we will provide to the unsupervised clustering algorithms to make them more efficient in finding relevant topics. The method based on how unusual is a term, will be used to create a seed list. The word to vector step will be done two times to compare a non-temporal method versus a temporal one

2) Data set: We will use the 2009 data set of twitter micro blogs. The analysis will first be done on a 0.1% sample of June 2009 tweets, and the obtained parameters will be applied to a 0.1% sample of September 2009 tweets to check the generalization of the method. Since the implementation is done in a way that is memory constant, applying it to the hole corpus will only take hours instead of minutes, but does not represent a treat for feasibility.

3) Data preparation: As we focus on graph techniques, and as data preparation of corpus is a topic by itself, we will not explore sophisticated techniques of preparation like lemmatization, or bi-grams, but rather use the most simple and conservative technique
A focus on this sub-part of the whole procedure, could constitute an improvement for the final results in subsequent works.

4) Pre-shaping the data: In this part we will compare two methods. One non-temporal : based on an Latent Dirichlet allocation of the corpus on the whole period. The other will consist building a metric of un-usuallity, that compares the occurrence of terms in each days, to their usual occurrence frequency. The aim is finding terms that ”make a spike” in their daily frequency graph.
The LDA is applied to a sample of the micro-blogs extracted on the desired period of time. The list of words resulting will be used as a seed list. We will try different parameters of number of topics. To compare performance, we will look to the wikipedia page of the concerned month and build a word list that are non-ambiguously related to the events. We will pick the method that produces the most words of expected events.

5) Vector space word representation: We will then use a word to vector technique to produce a vector space based on the entire corpus. We project this seed list and make it grow with the most cosine similar words to each words of the seed. We will compare two methods : a non-temporal one based on the overall co-occurrences, and a temporal one based on the daily frequency correlation.
Now that we have a few relevant words on top of which we grew a kind of semantic tree, we can build a graph based on this data.

6) Graph Clustering: We will then try to cluster this semantic graph to make topics by grouping words. We will try 6 techniques of which : Gaussian mixtures models, agglomerative clustering, K-means, Louvain algorithm, Clauset-Newman-Moore and Girvan-Newman technique. These methods will be compared by measuring the modularity of the resulting graph, and by the sense of the information they provide. To evaluate the sense of the topics created, we will use wikipedia to list the main events on the month concerned. We will look for the presence of these event. We will reiterate the methodology on other months of the data set to see if the approach is relevant.

7) Synthetic word selection: Since the grouped list of words is not very handy to interpret at one glance, we will compare the usefulness of picking one word of each topic, based on graph measures of which : word with the most degree, the most centrality, and the most be betweenness values, and best page rank.

B. Data preparation

We will filter everything that is not an alphabetical character, lower case it, and remove a list of common stop word, of english, french, spanish and german. We also use only the word that are in a frequency range of at least 1/50 the highest frequency in the corpus. We will keep the @ and the # to keep the information of a popular # or a popular user.
To further improve this sub-part of the process, The use of bi-grams can also be very useful, like for proper nouns of celebrities. We also can improve the stop list by adding words that will never improve the sense of results like ”things”, ”today” etc.

C. Pre-filtering : Creating a seed list

1) Latent Dirichlet Allocation:

a) Algorithm description: In this method, it is considered that each document is a mixture of k topics, that each has its word distribution. We will use the gensim implementation of the LDA algorithm.

b) Number of topics selection: There is no perfect solution to evaluate the right number of topics to be found. As a rule of thumb we will pick one non-equivocate highly related word for each event listed in the wikipedia page of the month concerned. We consider the hypothesis that this word is very likely to arise in a tweet talking of this event. If this word is in the seed list, it will have a structure in the graph. Here the aim is to evaluate if the seed list generated contains an important word related to that event, for which parameter does it arises and if it is possible to generalize the approach.

Here we will consider 10 events in June 2009 : The Air France Paris-Rio plane crash (target word : crash), the lift of suspension of Cuba from the American states organisation (target word : cuba), the outbreak of the H1N1 influenza strain (target word : flu), the protest during the Iranian presidential elections (target word : iran), the launch the NASA Lunar Reconnaissance Orbiter (target word : nasa), The death of Neda Agha-Soltan, an Iranian student shot during a protest (target word : neda), the step toward total independence from Denmark of Greenland, (target word : greenland), The death of Michael Jackson (target word : jackson), The Supreme Court of Honduras orders the arrest and exile of President Manuel Zelaya (target word honduras). (the Yemenia airlines plane crash is not considered since it occurred on the last day on June, this event will only be partially present in the data set).
We can see that in June in figure 1 that the optimal number of topics to cover the set of events is 100.

Interestingly we see in figure 2 that the LDA filter focuses on key words with fewer occurrences as we set a higher number of topics.

In order to test the reproducibility of the method, we try the same LDA, with 100 topics as determined as optimal on the June data.
Here we will consider the 5 major events listed in wikipedia

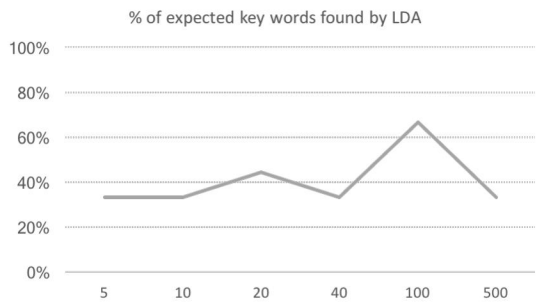

Fig. 1. % of relevant topics found by the LDA filter

: the G20 summit (target words : pittsburgh, summit, economy), the typhoon ketsana in manilia (target words : typhoon, ketsana, manila, philippines), the Guinean military clash (target word : guinea), the Samoa and Sumatra earthquake and tsunami (target words samoa, sumatra, earthquake, tsunami).
Unfortunately, none of the target words where present in the 1000 words resulting of the LDA (100 topics, 10 best words for each). In june we had major events that where highly represented in tweets like the Iranian elections and Michael Jackson's death, which where in the top 100 most frequent terms of the period (non including stop words and blacklisted words). The problem could come from that fact that this is a non temporal method. Since iran and jackson where terms with very high overall occurrence, it could be easier to identify. In September it seems that high occurrence terms over the month are less frequent. Moreover, this method needs inputs like number of topics, number of terms by topics, and is quite intense computationally, with no guaranty on the re-usability of the parameters on other months.

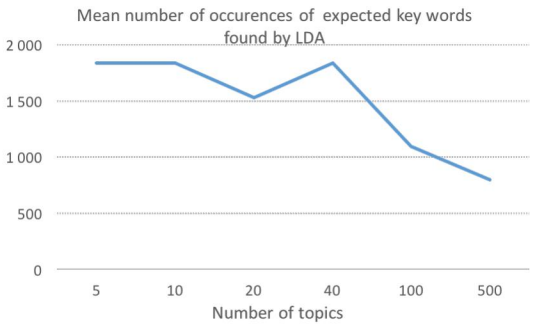

Fig. 2. Mean number of occurences of keyswords found by the LDA filter

2) Outlying frequency detection:

a) Method description: To replace the LDA approach which was non-temporal (taking every words in the period), we will try a more temporal approach based on how un-usual is the term frequency on each day of the period. The hypothesis is that if we make the daily frequency plot of each terms, a buzz word or a major event will create a spike on the corresponding days.
We will use a rather conventional approach looking at maximum seen daily frequency of terms and compare it to the mean daily frequency over the month. We will take care of pre-filtering words with very low occurrences to keep only the relevant terms in the results. For that we will keep the same rule of excluding words that have a frequency less than 1/50 of the maximum frequency

Outlying frequency metric of term i :

$$O_i = \frac{\max f_d i}{\frac{1}{N}\sum_d^N f_d i)}$$

Where N = total number of days, f = frequency of term i on day d

b) Results: Using this method we obtain a bi-modal distribution, separating usual and flat occurrences terms, and emerging ones on the observed month.

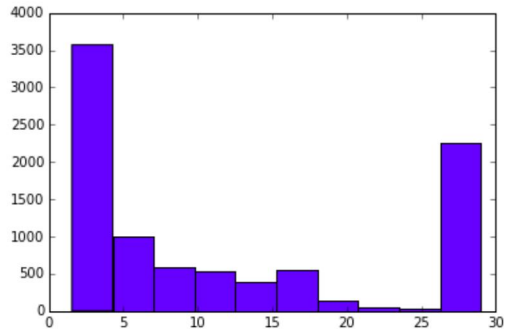

Fig. 3. Distribution of Outlying frequency metric

In order to keep a high level of summarizing, we will keep the seed small. The 100 tops words obtained are the following :
females, ginger, knw, nicely, keisha, guardianship, ig, bet, manufacturing, temple, hittin, processing, daar, skiing, overcome, hecho, breakdown, @spencerpratt, overview, dey, nesta, lmaooooo, persona, bollywood, rea, denmark, widescreen, quiere, @maestro, guten, firefighter, questionnaires, machines, sack, teh, yen, melting, mercado, prop, voucher, churchill, betawards, casey, nov, telesur, steady, lookbook, stickers, classics, shannon, crappy, askin, woulda, faire, discounts, hyped, girlfriends, ganz, ersten, glasgow, claire, dancin, philips, jaime, yaa, aerial, historia, neffe, fleet, muffins, wirklich, zoe, mara, stood, earnings, goode, planes, demitido, variety, harley, buffett, gin, junto, freely, simpsons, andrews, instructor, @repmikecastle, apresenta, randy, murio, @ocriador, molester, problemas, pedophile, spoken, depressed, unesco, jennings, mornings
By first sight, we do not see much of the major events we were expecting.
This might be due to the fact that a word that occurs only one day will always have an outlying measure near to maximum.

Moreover, again we need to give input parameters, like the occurrence bottom limit, which we wouldn't know how well it will generalize on an other months.

3) Frequency spread comparison:

a) Method description: Still following the same principle of finding the spike in the daily occurrences, we will look at the spread between the smallest non-zero daily frequency, and the highest daily frequency. This time there will be no input parameter or filter, since both un-frequent misspellings or very common words will have by construction a low spread and be discarded.

Frequency spread measure :

$$S_i = \frac{\max f_d i}{\min f_d i}$$

b) Results: This time we obtain a highly right skewed distribution of measures.

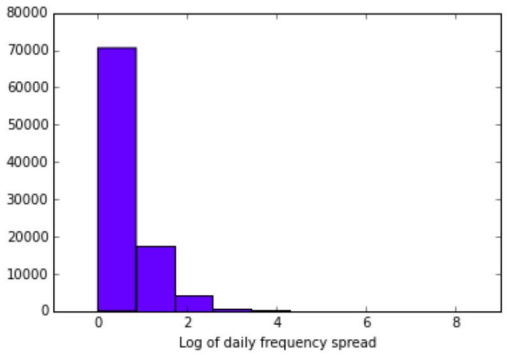

Fig. 4. Distribution of daily frequency spreads

The sorted 100 tops words obtained are the following :
mj, mays, farrah, post, jackson, fathers, lolquiz, ff, iranelection, fawcett, followfriday, sanford, eligible, forasarney, helpiranelection, airline, thriller, mcmahon, @msofficeus, tribute, perez, honduras, outlook, tehran, cnnfail, democracy, kobe, madoff, inaperfectworld, gr, honduras, liver, shaq, protesters, lakers, franken, dontyouhate, render, micheal, neda, stanley, quiz, billy, nkotb, michael, tmz, goldblum, mp, tho, cardiac, aaroncarter, usernames, penguins, proxies, msg, namorados, mcfly, invites, embassies, @jd, twitpocalypse, transplant, standards, ribbon, iran, peta, astley, pandemic, worm, destination, iranians, burton, vanity, dads, @stopahmadi, aaroncartercell, confuse, tilatequilalive, chupa, janet, @iamdiddy, cavs, musicmonday, @persiankiwi, basij, @change, haveyouever, unite, golpe, requesting, functioning, ware, prey, spree, pens,urs, rip, tethering, fisher, @peterfacinelli
We can see number of terms related to Michael Jackson, to Iran, as well we see Honduras, pandemic, airlines which were expected
If we look at the daily frequency of the term 'mj':

We can clearly see a spike starting june 25 and reaching its maximum June 26, following Michael Jackson death on june 25.

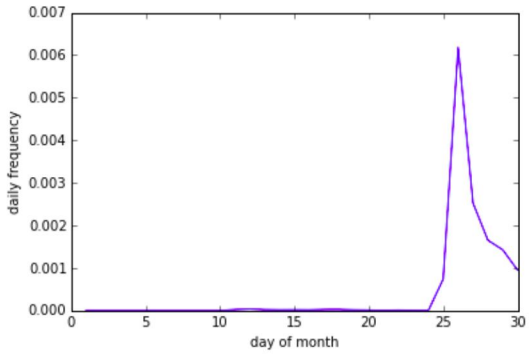

Fig. 5. Daily frequencies of term 'mj' in June 2009

If we look at another, less explicit term like 'sanford' :

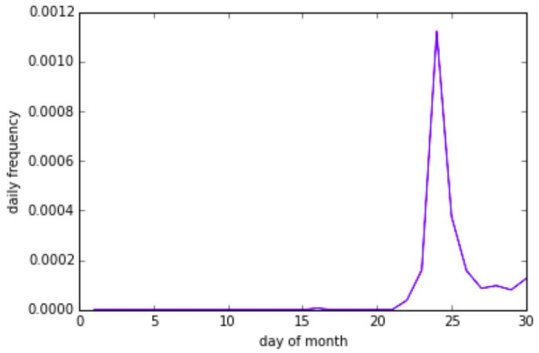

Fig. 6. Daily frequencies of term 'sanford in June 2009'

Again we can clearly see a spike on June 24. By looking at Google what happened that day, we see that it was the press briefing Gov. Mark Sanford of South Carolina addressed about his extramarital affair.

Now by applying the same methodology to the month of September, we find the following top 100 words :

swayze, kanye, vmas, ff, vma, patrick, tsunami, emmy, iwish, whateverhappenedto, musicmonday, emmys, admitit, lilmamais, labor, mayweather, viagra, followfriday, beyonce, jackass, leno, philippines, samoa, iamsinglebecause, serena, marquez, earthquake, pirate, typhoon, phrasesihate, supernatural, marvel, mafiawars, cowboys, wilson, cantlivewithout, @lucasfresno, @expensiveguy, lebaran, aftersex, odst, @thatkevinsmith, ilhabela, segue, uknowwhatilike, remembering, degeneres, cialis, @myfabolouslife, grocery, acorn, mms, gmail, swift, ellen, songthatmademecry, freelance, inmyhood, usc, victims, chatting, @estebantavares, anatomy, steelers, fsu, snl, why, ihatewhen, argentina, @adamlambert, teamtaylor, itmightbeover, pengakuan, mic, btw, dire, maaf, chrisbrownsbowtie, vmas, havn, happybirthdaymikey,

padang, diaries, seguindo, @congjoewilson, raiders, roc, beatles, divas, dontwifeher, worm, roman, gaga, eid, floyd, lite, mayer, luciferiscoming, diapers, thenines

Again we can see expected words like tsunami, typhoon, earthquake.

If we look at the term 'vma' :

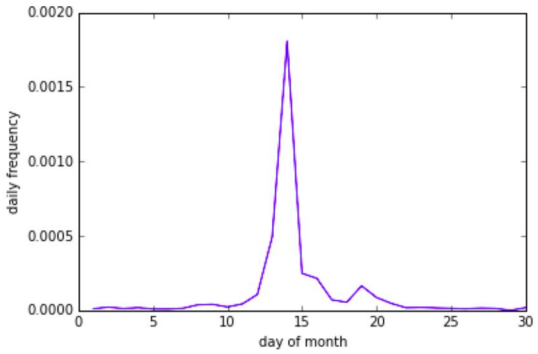

Fig. 7. Daily frequencies of term 'vma'

Searching into Google, we can see that it refers to the MTV video music awards that took place on September 13. Moreover we can see that the winners and most nominated were Beyonce and Lady Gaga which are also terms that appears in the seed list.

If we look at the term 'leno' :

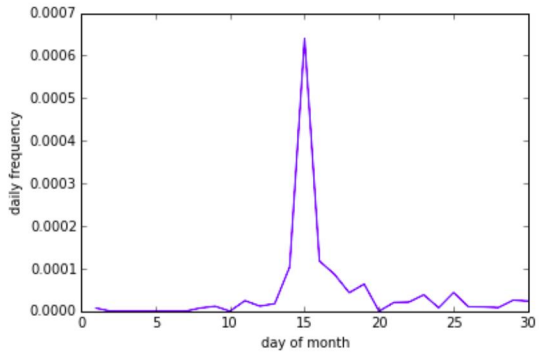

Fig. 8. Daily frequencies of term 'leno'

We can see it refers to the first of 'the Jay Leno show' which started on September 14

It seems that choosing words based on the importance of the spread of their daily frequency could be a solution to create concise seed of key-terms of relevant events. Moreover, this can be done very fast, do not need any parameter input, and seems quite robust.

D. Word to vectors

Now that we have a seed list containing terms about major events, we will try to expand it with other terms contained in the corpus and from which it is close. For that we will first use a word vector representation, then we will try the same seed expansion based on temporal correlation.
To keep the overall result interpretable and rather concise, we will only pick the 5 most related words for each word of the seed list.

1) Word2vec:

a) Algorithm description: We will use the gensim implementation of the word2vec algorithm.
The word2vec algorithm is a 2 layer neural net that takes in input a corpus of texts and produces a vector space in which each word has its vector, such as if 2 words that are used in the same context will be close in the vector space.

b) Results: To have a look at results, we will look at the three examples of the seed : 'mj' and 'sanford' in June and 'vma' and 'leno' in September to see how they expand.

for the term 'mj' (June) we obtain : jacko, #michaeljackson, jackson, michael, billy

for the term 'sanford' (June) we obtain : governor, affair, bush, gop, admits

for the term 'vma' (September) we obtain : vmas, #vmas, mtv, emmy, vh

for the term 'leno' (September) we obtain : cutler, @judy, jay, letterman, @blackmediascoop

We can see that the obtained results are completely understandable, and are quite obviously related

By looking at the cosine similarity distributions, we can see it is rather centered on high similarities. We can see as well that the September distribution is more skewed. We will see it is likely to be linked to the vma and the emmys that took place in September, and creates a cluster of very close terms (persons in fact) like Beyonce, Tayor Swift, Lady Gaga, Kayne West etc ..

While being rather sophisticated, the word2vec transformation is quite efficient - less than 10min on a low profile laptop on 1 million tweets - in providing meaningful results.
The parametrisation itself should be a subject, especially how to control language meaning versus the context meaning : for a word like 'house' how tuning word2vec will influence the similarity towards doctor house or towards doors and windows. This will be part of the possibilities of optimizations to make in further developments of the project.

2) Occurrence correlation: In this section we will try a temporal method to find similarities between words

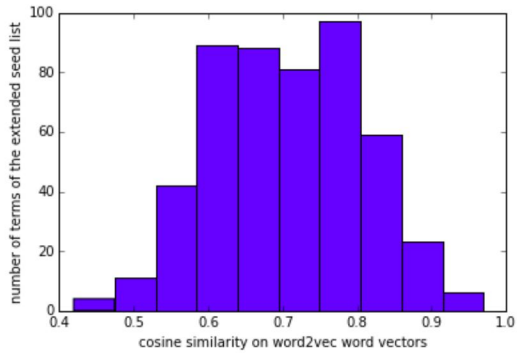

Fig. 9. Distribution of cosine similarities in the extended June seed list

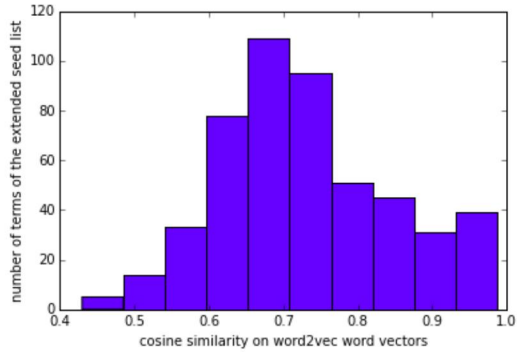

Fig. 10. Distribution of cosine similarities in the extended September seed list

a) Method description: After having tried a non temporal method, we will see if we could benefit using the daily frequencies as a vector space, and using correlation to find similarities between terms

b) Results: We will look at the same terms to see what differences there could be between the two methods :

for the term 'mj' (June) we obtain : thriller, mj, moonwalk, mjs, moonwalker. Which is rather close

for the term 'sanford' (June) we obtain : appalachian, outlook, uncool, oscars, standards. Here the first term seems related but not the others

for the term 'vma' (September) we obtain : vmas, vmas, preshow, vma, beyonce. Which seems quite good.

for the term 'leno' (September) we obtain : patrick, hiermit, verlost, swayze, sevensnap. Here we can see we are not having an accurate measure : while leno refers to a spike on September 14 related to the first diffusion of 'The Jay Leno Show', Swayze and Patrick are related to the death of the actor the same day.

To conclude on the occurrence correlation methodology, we can say that : since the profile of the events we are looking for tends to be 'one spike on one day', two un-related events are too likely to have the same signature, and be highly correlated while not linked.
For this reason we will not use the daily occurrences as a vector space.

E. Graph clustering

At this point we have an extended seed list of 500 terms. We are building a graph with this information : two terms share an edge if they have being linked during the seed list construction. This means we build a graph with respect to the seed structure. This has two advantages :
If we were considering all pairwise similarities, we would need to define a cutoff to determine whether we create an edge or not between two terms. And we aim at a method with less 'user defined' parameters as possible to make more easy a generalization over other periods of time.
Second, we are building a very structured graph - sometimes several isolated clusters of 5 terms - and we ask the clustering algorithm to group them, when they share edges. This will provide more regularly sized' clusters and guaranty good readability of clusters.

In this part we will try different clustering methods to create topics from the filtered semantic graph. To select one method we will use the one that provides the highest modularity of the graph. We will then check the "understandability" of the groups that are formed.
We will see Two kinds of methods : methods that take a given input of number of clusters, and methods that computes the optimal number of clusters.

1) Modularity: Modularity is a graph measure of the quality of groups made in a graph. A grouping is good if there are many edges within each groups, and few edges between groups. In this work, we will use that measure to make topics containing words very linked to each other in a certain context, and less related to other groups.

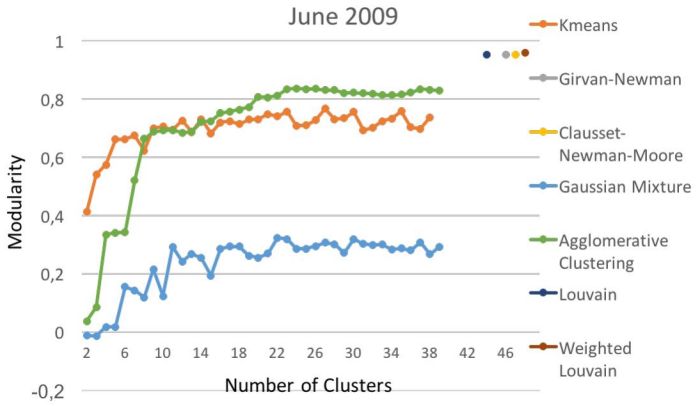

Fig. 11. Modularity results of different graph clusterings (June 2009)

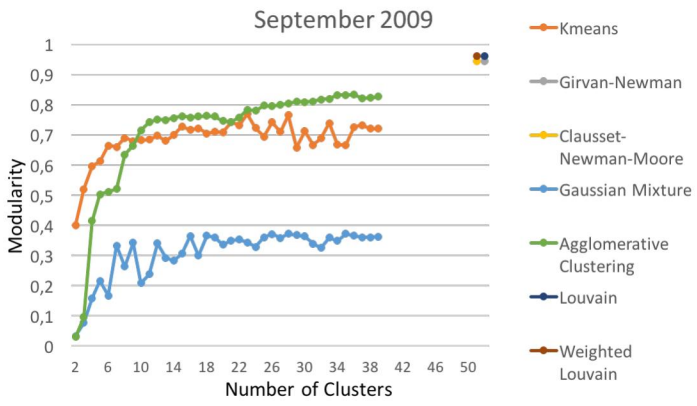

Fig. 12. Modularity results of different graph clusterings (Sept 2009)

2) K-means: This method consists in picking a defined number of centers at random in the vector space, assigning the closest center to each point, re-computing the centers on the mean of the assigned points, until the centers do not move anymore.
We will use the K-means clustering implementation of the NLTK package. The best modularity is 0.76 and is obtained for 27 clusters. (June)

3) Gaussian mixture model: Gaussian mixture models can be thought as a generalisation of the K-means clustering : instead of k center points, we consider the data coming from k Gaussians distributions of which the center are the k center points.
We will use the scikit-learn implementation of gaussian mixture models. The Gaussian mixture model has best modularity 0.32 for 22 clusters. (June)

4) Agglomerative clustering: Agglomerative clustering consist at grouping together the 2 closest 1-point groups, then grouping the 2 other groups that have the second less distance (and which can be part of the precedent grouping), etc until all points are grouped together. The resulting can be seen as a dendogram which is a tree representation of the successive groupings.
We will use the NLTK implementation of agglomerative clustering. The best modularity is 0.83 and is obtained for 24 clusters. (June)

5) Clauset-Newman-Moore: Like Gaussian mixture and K-means, CNM can be seen as an extension of agglomerative clustering, but in this case to groups are joined together if they bring the best increase in modularity.
We will use the SNAP implementation of the CNM algorithm. The obtained value is 0.95 for 46 clusters.(June)

6) Louvain: The Louvain algorithm consists in considering first as many groups than nodes, then for each node is calculated the gain in modularity if it is removed from is group and assigned to another group. This method usually brings very good modularity groupings in fast computations.
We will use the python Louvain package. The algorithm provides a grouping with a modularity of 0.95 with 44 groups.
If we apply the weighted version to our cosine similarity linked nodes, instead of considering a full edge if the similarity is over a threshold, we obtain a modularity of 0.957 with 48 groups.

7) Girvan-Newman: The Girvan-Newman algorithm is based on fact that if clusters in a graph are not well connected, then the few edges that ensure this connection must have high edge betweenness centrality, because many shortest path are forced to pass trough them. So the algorithm removes sequentially the edge with highest betweenness centrality at each time, to make emergent the underlying structure of clusters.
If we apply this method we find a modularity of 0.95 with 46 groups

8) Overall clustering results: At this step we will select the Clauset-Newman-Moore because it is extremely fast, does not need to pre-define a number of expected topics, has high modularity, and provide groups that are rather more logic and intelligible than the other methods.

F. Synthetic word selection

Our group of words can be quite big and are not very understandable at first glance. According to the fact one or two words within each group might resume the group itself very well, we will try different methods based on node measures to pick these nodes. We will look at examples of comparing for given groups, the node that has best degree, best betweenness, and best centrality.

Here we will use degree centrality, since it is likely to work well because, in the way we structured our seed list, we know that the parent node is relevant (it makes a spike on daily graphs) and is linked to all child nodes.

III. Knowledge graph implementation

The purpose of this part is to add external information to the cluster resume.
The aim is to consider the following use case : given the fact we are working for a car company, our goal is to analyse only tweets containing some wanted key words related to cars. Then we can analyse the topics and trend on this given subject. To add external information, we could parse and vectorize, for example, a mechanic documentation, or a marketing documentation, and then try use this to add information to our topic name, about what it is related to on a different given angle.
For the purpose of the exercise, we will use a context agnostic word vector space coming from the Glove 6Bilion-100 dimension dataset (mostly wikipedia), to add information to our context aware topics.

a) Motivation: We will project the terms of each topics we found in the context aware vector space, on this context agnostic space. That is we will pick the most similar words of each word in a topic, and in the same way as we did before, build a graph on these similarities.
Then we will try different centrality measures to pick a word in this new vector space, that could add relevant information to our topic name. We can imagine to characterize flower species or mechanic parts using a specialised vector space, trained on specific documents.
We will try Degree centrality, betweenness centrality, closeness centrality, and page rank.

A. Degree centrality

a) Method description: Here we will pick a term in the new graph that has the most degree centrality, that is the most edges in common with our topic terms.
To make it simple to visualize, we will use a few topics as examples.

b) Results: By applying the above method we find the following most degree central terms for topic "pandemic" and their number of degrees :
('virus', 5), ('outbreak', 4), ('avian', 4), ('h1n1', 4), ('outbreaks', 3), ('h5n1', 3), ('sars', 3), ('smallpox', 2), ('polio', 1), ('hiv', 1), ('epidemics', 1), ('measles', 1), ('epidemic', 1), ('vaccines', 1), ('vaccination', 1), ('vaccinations', 1), ('doses', 1), ('bird', 1)

We can see that the external graph brings usefull new information, principally related to pathologies. 'Virus' is here unsurprisingly the mostlinked term.

B. Betweenness centrality

a) Method description: Here we will try to pick the additional term that has the greater betweenness centrality, that is that has the most shortest paths going trough it.

b) Results: By applying the above method find the following most between central terms for topic "pandemic" and their corresponding betweenness centrality value :
('avian', 9549), ('outbreaks', 9549), ('h1n1', 9549), ('h5n1', 9549), ('polio', 8143), ('vaccination', 8143), ('doses', 8143), ('measles', 8143), ('vaccines', 8143), ('vaccinations', 8143), ('smallpox', 8143), ('epidemic', 5287), ('epidemics', 5287), ('virus', 2443), ('sars', 2443), ('outbreak', 2443)

We can see this metric is a particularly good complement of information, because the flu pandemic we are observing on this period is the avian H1N1 flu. Since there are some specific terms in our topic like swine flu, it is not surprising the term avian is on the most shortest paths of theses terms

C. Closeness centrality

a) Method description: Here we will look at the term which has the best closeness centrality, that is, we will measure the length of the shortest paths going trough it.

b) Results: By computing this metric for the small extended topic graph we built on our new vector space, we find the following results :
('virus', 0.564), ('smallpox', 0.478), ('avian', 0.468), ('h1n1', 0.468), ('outbreak', 0.468), ('outbreaks', 0.431), ('h5n1', 0.431), ('sars', 0.431), ('polio', 0.423), ('hiv', 0.423), ('vaccination', 0.423), ('doses', 0.423), ('vaccines', 0.423), ('vaccinations', 0.423), ('measles', 0.423), ('epidemics', 0.392), ('epidemic', 0.392), ('bird', 0.392)

We can see that this time the term "virus" comes first, which makes sense since, virus is the common denominator for the terms we are using

D. Page rank

a) Method description: Here we will try the page rank computation to compare it to the other results.
This method is based on accumulation of power based on the number of in going and out going edges, like a fluid circulating and accumulating through nodes.
Nevertheless since we are using an undirected graph, this metric will mostly be proportional to the number of degrees.

b) Results: By applying the method we find :

('virus', 0.0486), ('avian', 0.0383), ('h1n1', 0.03834), ('outbreak', 0.0383), ('outbreaks', 0.0297), ('h5n1', 0.02972),('sars', 0.0297), ('smallpox', 0.0254), ('polio', 0.01682),('hiv', 0.0168), ('vaccination', 0.01682),('doses', 0.01682), ('vaccines', 0.0168), ('vaccinations', 0.01682), ('measles', 0.0168), ('epidemics', 0.015139), ('epidemic', 0.01513), ('bird', 0.01443)

Which still is one of the most relevant metric, since the 5 first terms are well related to our topic.

E. Knowledge Graph implementation results

Since we could want to use all the information available, we could pick the best -non already picked- term in each metric to add it to our topic description.

For our pandemic topic we would have :
(22, 'pandemic', u'virus', u'avian', u'smallpox', u'h1n1'): for this topic made of the terms :791: 'flu', 1027: 'swine', 5121: 'pandemic', 8287: 'swineflu', 8332: 'vaccine', 8400: 'influenza'

Which is not bad, excepted for the term smallpox provided by the closeness centrality.

For our michael topic we would have :
(3, 'michael', 'goulder', 'keuning', 'devakula', 'lyons'): for this topic made of the terms 115: 'michael', 2141: 'micheal', 3995: 'janet', 6017: 'jesse', 8553: '#michael',

27412: 'michale', 32795: 'micha', 37086: 'browne', 69451: 'rathbone'

We can see here that our external graph is not sufficiently specialized to be relevant in this case.
Since it is mostly base on wikipedia corpus, our external graph is trying to link name of authors with first name michael, whereas it is a topic of Michael Jackson.

This kind of observation cancels the results we could use here with this method, but nevertheless it has shown as well a potential to add useful information, if the scope of the external graph is oriented towards the subject we are looking. Whereas our additional information was useful about disease, it wasn't for artist, which could be related to the wikipedia source of the information, that could be more useful for scientific subjects.

IV. Temporal visualisation

a) Motivation: In this part we will come back to the frequencies, at the topic level. We will try to represent evolutions of topics, and attempt to rank them by importance

A. Topic overall occurence

Here we will sum the daily terms frequency, to group them at a daily topic level.
Thus we can gather additional insight of what events happened on the period.
By re-using the daily dictionaries we used for computing the "un-usuallity metric" for the seed list, we can compute the overall frequencies for each topic.
For example, for a sample of topic in June and in September we obtain :

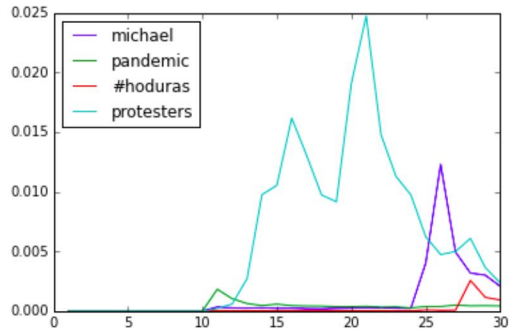

Fig. 13. Topic overall daily frequency in June 2009)

NB :Due to corrupted data in June, we do not have frequencies before June 10th.

In June 2009 :
We can see very clearly that something happened to Michael Jackson on June 26 (actually June 25).
As well we can see the Iranian protest are very well transcribed, with a first spike corresponding to the start of the protest, just after June 13, and the second spike corresponding to the death of Neda, a young student shot during protests on June 20.
June 28 The Supreme Court of Honduras ordered the arrest and exile of President Manuel Zelaya, which is visible on the plot.

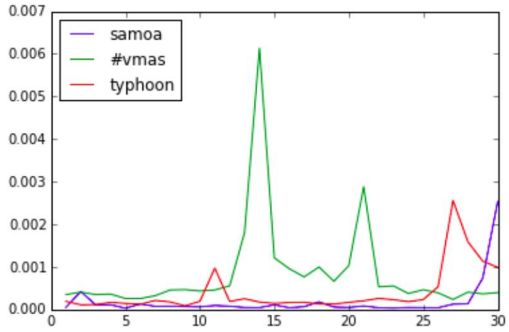

Fig. 14. Topic overall daily frequency in September 2009)

In September 2009:
On September 26th, the typhoon Ketsana caused record of rainfall in Manilia, Philipines
On September 29th the Samoa earthquakes strikes
On September 13th takes place the video music awards, and on September 20th took place the emmy awards.

The entire results of topics formed automatically by the methodology, without any tailored parameter, is available, in the appendix, for the June 2009 and September 2009 topics.

We can dig some very fine results, like the fact that Steve Jobs got a liver transplant on june 23th. It is noticeable in the topic 33 "Madoff" of June because interestingly, besides the judgment of Bernard Madoff on june 24th, a Mr. Madooff is also a transplant receiver, and a transplantation doctor.

B. Topic importance ranking

Here we will apply the same "frequency spread measure" than for the seed list creation, but it will be applied to the overall topic frequencies computed above.

We find the follwing results for june 2009 ((topic num., topic name), spread measure) :
((20, 'honduras'), 426), ((18, u'post'), 236), ((6, u'inaperfectworld'), 192), ((12, u'mp'), 160), ((41, u'forasarney'), 119), ((5, 'protesters'), 114), ((7, '@persiankiwi'), 89), ((16, u'followfriday'), 83), ((37, u'requesting'), 72), ((29, u'namorados'), 67), ((3, 'michael'), 66), ((1, u'mcmahon'), 62), ((45, u'haveyouever'), 54), ((39, u'functioning'), 49), ((28, u'burton'), 48), ((25, u'golpe'), 44), ((2, u'invites'), 42), ((26, u'proxies'), 33), ((32, 'fathers'), 32),

((43, u'perez'), 28), ((36, u'thou'), 22.457084934546131), ((34, u'confuse'), 20), ((10, 'outlook'), 20), ((11, u'franken'), 15), ((8, u'prey'), 12), ((17, u'twitpocalypse'), 12), ((15, u'cardiac'), 10), ((24, u'thriller'), 9), ((9, u'mcfly'), 9), ((14, u'unite'), 9), ((21, u'vanity'), 8), ((33, u'tethering'), 8), ((4, u'@peterfacinelli'), 8), ((23, u'peta'), 8), ((27, u'tmz'), 7), ((0, 'kobe'), 7), ((22, u'pandemic'), 7), ((19, u'musicmonday'), 6), ((31, u'madoff'), 6), ((35, 'tilatequilalive'), 6), ((40, 'spree'), 5), ((30, u'astley'), 4), ((13, u'destination'), 3)

For September 2009 : ((20, u'happybirthdaymikey'), 157.0), ((7, u'thenines'), 145.3), ((28, u'maaf'), 131.3), ((17, u'@estebantavares'), 91.7), ((35, u'mafiawars'), 90.8), ((24, u'eid'), 86.8), ((13, 'samoa'), 81.2), ((26, u'roc'), 78.2), ((22, u'teamtaylor'), 65.2), ((30, u'mayer'), 57.2), ((27, u'worm'), 50.3), ((11, 'dontwifeher'), 46.5), ((12, u'pirate'), 45.1), ((23, u'havn'), 39.1), ((32, u'lebaran'), 33.5), ((21, u'padang'), 32.6), ((8, 'typhoon'), 29.6), ((5, 'vmas'), 26.0), ((41, u'serena'), 22.6), ((19, 'ff'), 22.3), ((16, u'ellen'), 21.7), ((29, u'remembering'), 18.6), ((33, u'freelance'), 18.0), ((9, u'ilhabela'), 15.3), ((1, u'swift'), 14.7), ((25, u'btw'), 13.1), ((31, u'lite'), 12.9), ((46, u'wilson'), 12.9), ((2, 'steelers'), 12.0), ((4, u'why'), 11.7), ((34, u'jackass'), 11.0), ((18, u'diapers'), 11.0), ((44, u'marvel'), 10.7), ((14, u'@congjoewilson'), 9.8), ((6, '@myfabolouslife'), 8.0), ((50, u'leno'), 7.9), ((38, u'roman'), 7.4), ((49, u'odst'), 7.2), ((40, u'argentina'), 7.1), ((42, u'gmail'), 6.5), ((15, 'musicmonday'), 6.5), ((3, u'anatomy'), 6.0), ((43, u'pengakuan'), 5.9), ((37, u'snl'), 5.3), ((10, '@adamlambert'), 5.0), ((36, u'floyd'), 4.4), ((0, 'uknowwhatilike'), 3.4), ((47, u'beatles'), 3.3), ((39, u'@expensiveguy'), 3.1), ((45, u'mic'), 3.0), ((48, u'chatting'), 3.0), ((51, u'mms'), 2.8)

By looking at the results, and for some, the corresponding plots, we might consider using and absolute value of importance rather than a relative one. for example, Honduras doesn't seem to be a very popular subject, which leads to a very important relative spread measure during the presidential events. This could be misleading since, there are other topics that are orders of magnitudes more popular in absolute values.

V. Outbreak detection

The aim of this topic is the answer the following use case : a company wants to monitor its fame on twitter. There are interest to detect the sooner as possible a potential bad buzz. We will look at two ways of doing it here. Unusual spread occurrence, and Unusual spread occurrence reinforced by user importance.

A. Unusual spread occurrence

Here we can apply the exact same method than above for seed term detection.
In this case, we might want to look at frequencies in minutes intervals, and compare it the the min observed frequency over a past period of time.
The threshold of alert could be a simple statistical measure over the past major events, to determine that, past a given spread value, 99% of the time it was related to a major event. This might be a way to monitor a raise alerts very early in the event. Since we would have to simulate a stream of tweets, for the example, and since it is the exact same method than above, we will keep this development for further work in implementing this monitor.

B. Unusual spread occurrence and user importance

Here the aim would be to modify the threshold above, according to the user that tweeted.
The result would be if a defined set of users tweets, we know that the buzz threshold is lower, since these users are influent, they are more likely to start a buzz than others.
We can compute a page rank over the twitter graph and define a set of influent users.
Again, we can look a the past major events to see how they reacted and of how much we need to adjust our buzz threshold.
Since this part is basically using the snap pagerank algorithm on the users of which we used the tweets for the analysis : over the million tweet analysed for each month, there where approximately 500k unique users, of which only 300k where found in the twitter API to retrieve their user id from the provided user name in the tweets.
Then we need to trans-code the tweeter id to the compact graph id, which makes us fall approximately at 250k users with a name, a twitter id and a graph id. then we need to filter the 1.4billion edges of the 2009 twitter graph to keep only the ones of our data-set. Once we have this, we can build a standard directed snap graph, apply the page rank algorithm and retrieve a set of influent user. Since we are encountering technical problems on the penultimate part of filtering the edges on our machine, we might not be able to present a set of influent users in this work. Nevertheless, it does not represent an important result to be used by another method neither a challenging part to realize here.

VI. Conclusion

We saw a simple technique to detect important events trough tweets.
This method has the benefit it has no user parameters, it is computationally quite light, the results are understandable by non experts and they give insights to what happened. Moreover the same approach could help determining what is happening right now.
This first try helps to structure an entire process of data manipulation. It is useful to identify improvements at each steps, noticeably, the use of bi-grams in data preparation, to capture the proper nouns, the tuning of word2vec, and the absolute measure of importance of a topic.
Thank you for having read this.

REFERENCES

[1] Michael I. Jordan David M. Blei, Andrew Y. Ng. *Latent Dirichlet Allocation*. 2003.

[2] Qin Lu Ji Wang Wenjie Li Jintao Tang, Ting Wang. *A Wikipedia Based Semantic Graph Model for Topic Tracking in Blogosphere*. 2011.

[3] Lyle Ungar Dean Foster Joao Sedoc, Jean Gallier. *Semantic Word Clusters Using Signed Normalized Graph Cuts*. 2016.

[4] Natasa Milic-Frayling Jurij Leskovec, Marko Grobelnik. *Learning Substructures of Document Semantic Graphs for Document Summarization*. 2005.

[5] Pedro Szekely Mohsen Taheriyan, Craig A. Knoblock and Jos e Luis Ambite. *A Graph-Based Approach to Learn Semantic Descriptions of Data Sources*. 2013.

[6] Yutaka Matsuo Takeshi Sakaki, Makoto Okazaki. *Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors*. 2016.

[7] Maarten de Rijke Tom Kenter. *Short Text Similarity with Word Embeddings*. 2015.

APPENDIX A
CLUSTERING RESULTS BY CNM, WITH THE HIGHEST DEGREE NODE AS THE DESCRIPTIVE WORD, WITH 100 WORDS IN THE INITAL SEED LIST AND A VECTOR SPACE OF 100 FEATURES

(Topic ID : Topic name (based on the most degree node of the topic)
The result come from the methodology describe above, and tries to resume the initial corpus of 930 000 documents in June and 940 000 in September

June 2009 results :

========== 22 : thriller ==========
drake beyonce vh thriller king jamie rapper mtv rendition rap beatles ciara pop soundtrack neyo keri elvis ========== 11 : confirmed ==========
arrested embassy reported confirmed unconfirmed arrest declared injured azadi ========== 25 : jeff ==========
colbert jeff andrew brett goldblum stephen ========== 6 : quiz ==========
danny nkotb efron quiz jord quiztweet selena niley try describes took demi mcfly saw became spoke jonas lolquiz ========== 10 : firefox ==========
safari browser mozilla beta firefox firefox ========== 41 : protests ==========
elections protesting demonstrations protests riots protestors ========== 2 : jtv ==========
jtv baste meron lng akong panalo ========== 5 : alpha ==========
tweetboard email alpha ware script requesting wolfram invite ========== 1 : mcmahon ==========
dead perform homage mcmahon jacko swells tribute michaeljackson rip dies death farrah killed died mays murder passed jackon tears gove deaths billy jacksons @betawards fawcett vick travalena @bet oxyclean jackson fawcet farah mj ========== 45 : jon ==========
kate lajoie jon letterman stewart gosselin divorce ========== 7 : democracy ==========
meddling @change avatars slachtoffers overlay avatar renewing helpiranelection democracy ribbon ========== 24 : pride ==========
fireworks celebration queer pride stonewall parade ========== 20 : @perezhilton ==========
@perezhilton @jimd hilton unfollowperezhilton @johncmayer hiltons punched unfollowperez perez ========== 34 : fathers ==========
gifts sons daddy daddies fathers @willie independence father mothers dads wife ========== 39 : spain ==========
italy brazil confederations usmnt spain egypt ========== 12 : followfriday ==========
@starlingpoet rt followfriday ff @modelsupplies ========== 46 : ronaldo ==========
cristiano ronaldo madrid kaka benzema torres ========== 0 : cleveland ==========
lebron cleveland cavaliers texas cavs shaq @the howard kobe orlando rt@the dwight suns bryant ========== 3 : iran ==========
iran mosque neda iranian @irannewsnow soltan forces cnnfail regime neda persian silence iranlection gr iranians tehran @elham iranelection tehran iran basiji violence protester @jimsciuttoabc protesters mousavi basij soltani iranelections ========== 19 : poll ==========
council poll congress results voters policy ========== 40 : member ==========
proud nationa members fan member group ========== 14 : pandemic ==========
pandemic swine vaccine influenza flu swineflu ========== 42 : flight ==========
plane continental airlines flight pilot crash ========== 13 : michael ==========
janet michael michael michale micha micheal browne ========== 4 : airline ==========
@gilbirmingham airline peterfacinelli rothbury message orbitz @peterfacinelli tix giving chance ticket eligible @orbitz peterfacinelli tickets carlise enter refund facinelli msg enchanteddandelions ========== 9 : standards ==========
outlook efforts websites tactics standards render @msofficeus emails ========== 27 : @mileycyrus ==========
@ddlovato @nickjonas @tommcfly @selenagomez @mileycyrus @jonasbrothers ========== 29 : sanford ==========
gop sanford governor affair bush admits ========== 16 : @persiankiwi ==========
@tehranbureau @iranriggedelect @iranelection @oxfordgirl @persiankiwi @stopahmadi ========== 15 : post ==========
post entry @thinkbig infoseekchina posting archive ========== 17 : squarespace ==========
squarespace trackle iphone jailbreak @squarespace ========== 23 : username ==========
twitterers account password facebook urls usernames facebook username vanity @eadvocate url arrests ========== 33 : madoff ==========
prison transplant bernie madoff sentencing steve bernard sentenced liver donte stallworth ========== 26 : tweetdeck

==========
ubertwitter twitterfon twitterific tweetdeck seesmic tweetie ========== 32 : gmt ==========
timezone settings gmt zone location confuse ========== 35 : awards ==========
winning awards cmt award performing ========== 43 : thursday ==========
sunday wednesday saturday monday tuesday thursday ========== 36 : forasarney ==========
chupa @fepaesleme forasarney sarney @marcelotas @twpirata ========== 37 : transformers ==========
trailer movie sequel revenge hangover transformers ========== 30 : pack ==========
pack bottle bag custom bags shoe ========== 8 : nba ==========
mlb baseball draft nhl nba title streak chapter lakers laker penguins ncaa championship nfl finals ========== 21 : tiny ==========
thick toya ugly skinny creepy tiny ========== 28 : ahmadinejad ==========
khamenei nejad recount ayatollah mahmoud ahmadinejad ========== 44 : california ==========
california alaska southern county florida michigan ========== 18 : honduras ==========
nicaragua honduras crisishn zelaya honduras barcelona pide mexico argentina manuel colombia

September 2009 results :

========== 18 : roc ==========
roc grandmaster tupac shakur raida travers ========== 44 : havn ==========
havn bullsh mido bient rkei at ========== 14 : @congjoewilson ==========
potus pelosi @congjoewilson gop democrats acorn obama @glennbeck racism ========== 31 : padang ==========
menit bersama naek padang datang euy ========== 27 : chatting ==========
chilling chat chatting buzzing obssesed skyped ========== 39 : mic ==========
ball stage mic door towel bobby ========== 41 : wilson ==========
mccain republican carter joe wilson congressman ========== 45 : thenines ==========
fridaynightcranks deland dayton thenines surf movie cialis hambric viagra @bobbibillard ovi rhsummit soa ========== 7 : @expensiveguy ==========
jk ohhh ahaha ummm ohh @expensiveguy ========== 35 : snl ==========
snl nbc fallon gma maddow comedy ========== 51 : mayer ==========
@bepp mayer apartamento @thelover jay zemayerfacts ========== 13 : pirate ==========
valentines labour labor qods @willie independence mateys arrrrr pirate punctuation ========== 33 : itmightbeover ==========
uknowufrombrooklyn uknowufromla hoe @mredlover itmightbeover wifeher dontcuffhim dontwifeher yourlame uknowufromqueens ========== 28 : beatles ==========
beatles remastered rockband band guitar muse ========== 17 : ilhabela ==========
@talita seguindo @cih sigam @pri indicando ajuda indico galera @localmosa ilhabela segue ========== 11 : floyd ==========
keith floyd billy derek jeremy brian ========== 43 : gmail ==========
error firefox gmail browser tweetdeck server ========== 30 : leno ==========
@judy letterman jay cutler leno ========== 26 : @adamlambert ==========
@davidarchie henrie @philipbloom @mileycyrus @thatkevinsmith @david @selenagomez @adamlambert @ddlovato @thedavidcook @jonasbrothers ========== 24 : maaf ==========
batin mohon maaf bathin minal lahir ========== 36 : jackass ==========
prick politician moron jackass barack liar ========== 48 : @estebantavares ==========
@estebantavares @coelhors @lucasfresno @cesinha @celsoportiolli @himynameisdh @geerocha ========== 38 : happybirthdaymikey ==========
mikeybdayvideobr mcrmybr @simonsaystrd happybirthdaymikey youtube ticketzz ========== 32 : samoa ==========
philippine magnitude @breakingnews sumatra tsunami quake samoa tehran earthquake ========== 25 : btw ==========
btw fdp piraten gegen cdu piratenpartei ========== 42 : ff ==========
ff rts @lele follow shoutout mentions followfriday ========== 19 : argentina ==========
brazil venezuela portugal peru colombia argentina ========== 21 : teamtaylor ==========
@oceanup happybirthdaynickj teamkanye teamtaylor robsten taylor ========== 2 : why ==========
uknowublackwhen why whyyy morningafterquestions ova inmyhood lightskin dese harlem @andybizzo cmon alwayz ihatewhen jerz @giiwiz whut ========== 34 : typhoon ==========
dire waystation ondoy flood ateneo clothng victims donations families philippines philippines floods typhoon ========== 22 : worm ==========
@madyar @mentions worm spreading @mayhemstudios phishing ========== 10 : diapers ==========
huggies grocery diapers bucks luvs certificates groceries ========== 23 : mafiawars ==========
mafiawars @mafiawars mafia mafiawars atk armor ========== 40 : ellen ==========
oprah idol dioguardi ellen kris paula degeneres abdul ========== 16 : anatomy ==========
anatomy greys buffy dexter @superwiki fringe twilight supernatural bonesseason vampire supernatural inkripkewetrust weed pdiddyisscaredofhistv luciferiscoming grey diaries ========== 0 : lite ==========

enabled touchscreen appstore ddr mhz lite ========== 3 : @myfabolouslife ==========
rhianna @hapis lilmamais tpainbetter lmaooooo fabolous fabsteeth @lilduval @bowwow tpain chrisbrownsbowtie maia @myfabolouslife lmfao ========== 1 : musicmonday ==========
mm remix mixtape mariah britneyspears feat @imeem songthatmademecry musicmonday ========== 29 : remembering ==========
remembering mourning tragedy gospel fathers romans ========== 49 : mms ==========
mms iphone tethering iphone android ========== 8 : uknowwhatilike ==========
imsinglebecause wish idgaf inhighschool gnr uknowwhatilike ihatewhengirlssay hoes wishing lmfaoooo phrasesihate admitit lmaoooo lmfaooo fact @shantheprincess lls whateverhappento iamproudof niqqa uknowhowiknowuregay iamsinglebecause lmaoo wishin lmaooo iwish cantlivewithout iminchurch whateverhappenedto duringsex females beforesex loves aftersex ========== 15 : marvel ==========
walt disney sequel pixar galaxy marvel ========== 20 : odst ==========
multiplayer halo xbox odst wii ========== 6 : vmas ==========
vma dollhouse award divas backstage emmys awards vh mtv vmas emmy vmas dwts janet emmys ========== 12 : steelers ==========
usc ufc ucla yankees jets pats osu bears romo fsu steelers colts boxing cowboys favre marquez raiders giants lions titans mayweather ========== 4 : swift ==========
gagas swifts swayzes bey taylor neil micheal patrick keri funeral momsen swift mj rihanna @ladygaga gaga kanya kanye beyonce swayze obama kayne lautner britney ========== 46 : pengakuan ==========
sih pengakuan @radityadika belum banget bener ========== 37 : lebaran ==========
neng selamat taun sudah jumat lebaran ========== 5 : freelance ==========
magento php freelance joomla odesk freelance ========== 50 : serena ==========
robbie williams clijsters oudin federer serena ========== 9 : eid ==========
eid mubarak hashanah fitr rosh kippur ========== 47 : roman ==========
arrest arrested roman alleged polanski murder