

# A Twitter Analytic Tool to Measure Opinion, Influence and Trust

Ahmed Sameh

Prince Sultan University, Riyadh 11586, Saudi Arabia

Email: Sameh.aucegypt.edu@gmail.com

**Abstract-** We are using combination of data-mining and text-mining techniques to measure opinion, influence, and trust spread in Social Networks within the domain of political elections. In particular we study the case of Twitter and focus on three 2012 presidential elections: the American, the French, and the Egyptian. We propose a new real-time analytic tool for elections. The proposed tool collects live tweet streams on frequent time intervals from followers of presidential candidates and computes a mix of qualitative and quantitative indicators to measure our three parameters (metrics dashboard). English, French and Arabic tweets are analyzed taking into consideration important context background information. We focus more on “Arabic,” and present three “Tweet Coloring” algorithms that speculate voters’ inclination and impression about the candidates and the influence of the candidates on the voters. On the other hand, three “Edge Coloring” algorithms are used to speculate on opinion, influence, and trust spread (cascading) through Twitter social network graphs. The algorithms compute inferred value ratings through polling neighbors and weighted averaging. A prototype of the proposed tool is implemented on-top of “Nodexl”; an open source template for Microsoft Excel that allows automated connection to a social network server and import (Using Twitter APIs) any data stream into the usual Excel environment. Tweet coloring and Edge coloring algorithms are implemented as Excel Macros with selective setting to either surface, shallow, or deep analysis parameters. Visualization graphs are provided that allow dynamic filtering, vertex grouping, adjusted appearance (zoom into areas of interest), graph metric calculations, etc.

**Index Terms**—text mining; social networks; opinion; influence; trust; analytic tool; and Q-wordnet

## I. INTRODUCTION

It is well known that currently 20% of people worldwide are using Social Networks. In 2012, there is 1.43 billion social network users, a nearly 20% increase over 2011. One out of every five people worldwide will use a social network this year; one in every four will do so in 2014. This number is projected to rise to 67.6% in 2013 and then to 70.7% in 2014. Most users have multiple accounts on popular Social Networks and spend

fair amount of time reading and writing posts on these networks.

Social Networks Analytic tools are used to monitor and analyze posts and activities on these Networks. Social Networks are the new media that people use for exchanging ideas and opinions about many areas such as arts, politics, religion, science, social, and economic issues. In this paper we are drawing our attention to the area of politics. In particular we are analyzing political posts and studying the three 2012 presidential elections: the American, the French, and the Egyptian over Twitter posts as case studies. We collect live tweet streams on frequent time intervals from followers of several presidential candidates and compute a mix of qualitative and quantitative indicators to measure out three parameters: Opinion, Influence and Trust spread. Opinion of followers about the candidate and his/her stand on issues and how this spreads. Influence of the candidate on his followers and how this spreads. Trust measurement among voters and candidates is also of our interest. We provide three levels of analysis: Surface, Shallow, and Deep depending on analyst wish and the traffic rate of input streams for the purpose of providing real-time mining service. English, French and Arabic tweets are analyzed taking into consideration important context background information. We focus more on “Arabic,” and present three “Tweet Coloring” algorithms that make use of previously developed NLP tools and graph algorithms such as “Arabic WordNet,” “Q-WordNet” ontology for sentiments, “Arabic Lexicons,” “Arabic Tweet Corpus,” “Max Flow Minimum Cut” in order to speculate voters’ inclination and impression about the candidates and the influence of the candidates on the voters. Similar tools and algorithms are used for English and French tweets. On the other hand, three “Edge Coloring” algorithms are used to speculate on opinion, influence, and trust spread (cascading) through Twitter social network graphs. Each one of the three Edge coloring algorithms has a “binary” version (+ve/-ve, Yes/No) and a “continuous” version. The algorithms compute inferred value ratings through polling neighbors and weighted averaging.

The structure of this paper is as follows: in section 2 we present three “Tweet Coloring” algorithms. These algorithms are based on combination of data-mining and text-mining techniques to measure opinion, influence,

---

Manuscript received February 25, 2013; revised April 20, 2013.

Copyrighted, Project number: 5 -2013 Prince merger Research Center, Kingdom of Saudi Arabia

Corresponding author: Ahmed Sameh, Prince Sultan University

and trust. In section 3, we present three “Edge Coloring” algorithms that measure the spread (cascading) of opinion, influence, and trust in a Twitter graph. Section 4 presents the overall analytic tool architecture. And Section 5 presents the implementation details of the system. Two sets of experiments are presented in section 6. Related work is presented in section 7, and finally the conclusion is presented in section 8.

## II. TWEET COLORING AND TEXT

Tweets are text of no more than 140 characters. We use text mining algorithms to discover and extract knowledge implicitly embedded in the body of a Tweet or during an exchange of Tweets of unstructured text. Text mining is based on “Natural Language Processing- NLP” and “Information Retrieval-IR”. On the other hand, an opinion (or regular opinion) is simply a positive or negative sentiment, view, attitude, emotion, or appraisal about an entity or aspect of the entity. Opinion orientation is positive, negative or neutral. The goal of an opinion is either speculative, persuasive, impression, and/or inclination. Opinions matter a great deal in politics. In this work we focus on understanding what voters are thinking, election candidates’ attacks, support, etc. Tweet coloring expands on the idea of opinion orientation, and borrows from Web page ranking. We apply similar ideas using text mining algorithms to color tweets according to the degrees of voting or non-voting for the candidate. We collect live tweet streams on frequent time intervals from followers of presidential candidates and apply these tweet coloring algorithms to color tweets to measure opinion support or non-support for a candidate. We provide three levels of analysis: Surface, Shallow, and Deep depending on analyst wish and the traffic rate of input streams for the purpose of providing real-time mining service. English, French and Arabic tweets are analyzed taking into consideration important context background information (e.g. culture of the country, political background, nature of the election process, etc.).

### A. Surface Analysis

At this level of analysis we apply word-level text mining algorithm called “Word Bag”. In this algorithm we have training and testing phases. In the training phase we analyze a collection of Tweets from the Tweet stream and build two word bags “Positive Words Bag” and a “Negative Words Bag” (sentiment words such as great, excellent, horrible, worst, are used to identify which bag to color the Tweet, etc.). This is kind of training is simply building an “Opinion” corpus. The need for this step is evident since Tweeters’ language is completely different from standard everyday life text language. Tweeters use their own “slang” language and their own special words and symbols. That is why it is important to build a special corpus for this Tweets language, we called it “opinion” corpus. In the first step of the coloring algorithm, Tweet clustering is used to reduce the dimensionality and categorization of the training tweets according to the political domain. Unsupervised machine learning is used at this step. Tweet pre-processing is the next step, where

text cleanup is performed (un-important stop words are deleted), followed by “Tokenization” where roots (stems) of words are produced, followed by POS (Part of Speech Tagging) where syntactic tagging is applied, and finally “word sense disambiguation” is applied. In the training phase a teacher (supervised training) is needed to classify the produced words into one of the two bags: positive and negative. This is called “opinion lexicon generation” stage. Then in the testing phase, tweets are colored positive or negative depending on the number and POS of the +ve/-ve words in the tweets identifies from the two bags. This Tweet surface coloring algorithm has a “binary” version (+ve/-ve, Yes/No) and has no “continuous” version.

### B. Shallow Analysis

This level of analysis extends the previous level of analysis by replacing the “bag of words” with a “feature vector,” where the words produced from the training tweets are inserted into a feature vector in the vector space. In the politics domain we choose the best features that best characterize the domain. Then we use a classifier to automatically generate labels from these feature vectors. These labels are thought of as column heads of a database table. This is a tweet level syntactic analysis where phrases along with words are also considered in the feature vectors. This is a continuous version of the surface analysis algorithm above. Text transformation is a preprocessing stage that prepares for the feature selection process. The prediction model is built in the pattern discovery stage. Then finally the testing mode is performed at the interpretation/evaluation stage.

### C. Deep Analysis

This level uses the attributes identified in the previous algorithm to create a corresponding database. Now the training set of tweets has been converted into a structured database. At this stage we can apply data mining algorithms such as discovering patterns, answer queries. At this level traditional data and visualization tools are used to analyze and visualize the results. Semantic analysis is considered at this level where ontologies such as “Wordnet” and “Q-Wordnet” are used to understand hidden meanings.

Also background context knowledge about the political domain is consulted during the semantic analysis. This is another continuous version of the surface analysis algorithm above. Arabic Wordnet is classified into nouns, verbs, and adjectives. Once a POST (Parts of Speech) analysis of a Tweet is done, nouns, verbs and adjectives can be identified and used as input into their corresponding Arabic Wordnet ontology and consequently deep understanding of the Tweet becomes possible. Background and context can also be used as in our previous work [1] to help understand the Tweets.

Fig. 1 and Fig. 2 show the deep analysis Tweeter coloring algorithm. In this algorithm the following processing stages are applied in order:

- NLP: parser that produces output words.
- Semantic Parser: produces output concepts and

their frequencies. AffectiveSpace produces output vector of eigen-moods of given concepts clustered with respect to the Hourglass model

- Automatically mine opinions with certain confidence level
- First Training mode of the Algorithm: Input set of tagged opinions for the subject matter (say 1000 tweets sampled from a subject tweet site). Identify concepts commonly used in this specific topic tweets and evaluate how important each concept is to a set of opinions (emotions) concerning the same topic through weighting equation
- Second Testing mode: Input new tweets. Use “Spreading Activations” that relates concepts to their related concepts with the same ontology
- Put in a database the posts and their computed opinion index

Algorithm 1: Suggested Algorithm

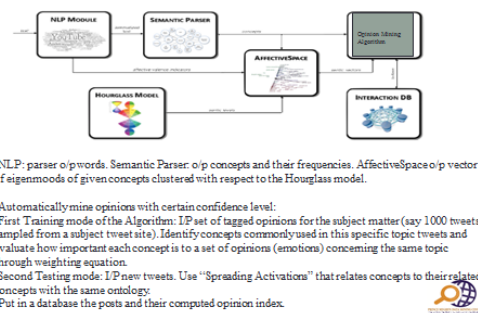


Figure 1. The deep analysis tweeter coloring algorithm

Effective Space: Bottom-Left corner +ve, and Upper-right corner -ve



A multi-dimensional space built by blending ConceptNet (a semantic net of common sense knowledge) and WordNet-Affect (a linguistic resource for the lexical representation of emotions). This alignment build a matrix: rows are concepts and columns are emotions, and whose values indicate truth values of assertions. +ve values for +ve emotions and -ve values for -ve emotions, 0 when nothing is known about the assertion. Dot product is high when two vector concepts are close in the AffectiveSpace above (eigenmoods form the axes of the above effective Space) Similar one for Arabic and Eng-Arabic tweets, threads, and text

Figure 2. The multi-dimensional space of concept net and wordnet affect

A multi-dimensional space built by blending Concept Net (a semantic net of common sense knowledge) and WordNet-Affect (a linguistic resource for the lexical representation of emotions)- See Fig. 2.

- This alignment build a matrix: rows are concepts and columns are emotions, and whose values indicate truth values of assertions. +ve values for +ve emotions and -ve values for -ve emotions, 0 when nothing is known about the assertion. Dot product is high when two vector concepts are

close in the AffectiveSpace above (eigenmoods form the axes of the above effective Space)

- Similar one for Arabic and Eng-Arabic tweets, threads, and text.

This diagram emulates Marvin Minsky’s conception of emotions [2]. 4 dimensions with 6-levels (-3,+3) – sentic levels- each: 1- the user is happy with the candidate, 2- the user is interested in the Tweet supplied, 3- the user is comfortable with the candidate, 4- the user is disposed to the candidate. Sentric vector is of 4 dimensions

### III. EDGE COLORING AND DATA MINING

Twitter followers and following create twitter graphs that can be manipulated using traditional graph theory algorithms to measure the spread (cascading) of opinions, influence and trust in a particular social network graph. We use the following “Edge Coloring” algorithms to discover the spread (cascading) of opinion, influence and trust. They are based on the traditional “Maximum Flow Minimum Cut” family of algorithms. The idea is that we would like to compute the flow of “Opinion,” “Influence,” and “Trust” from one follower to another by analyzing a sequence of Tweet exchange between them.

There is considerable literature on graph theory, network optimization, and the minimum cut set problem. The accomplishments of the work reported here are a) to find and implement a practical way of solving large networks for minimum cut sets and b) to discover that the minimum cut set for a large Twitter network was much smaller than what might have been expected given the number of edges in the network.

#### A. Opinion Spread

Fig. 3 shows an extract from the opinion spread algorithm. After using any of the Tweet coloring algorithms above, an edge between two nodes  $i$  and  $j$  is either incremented or decremented according to the tweet color  $ij$  or  $ji$ . At the end of this stage (several exchange of Tweets) an edge is colored according to the accumulated count over that edge. Large positive corresponds to heavy dark color, large negative corresponds to light blurred color. We use these counts as edge capacities in the well known “Maximum Flow Minimum Cut” problem. Solving for a minimum cut set is a well-known network problem, and existing map data and network solvers could have been used to answer the question posed. We have used a modified network solver algorithm- see Fig. 3. The first step in this algorithm is to determine where, in a general sense, the barrier is to be located. In our case, because we wanted a barrier surrounding the center of the graph (the candidate), we defined the barrier’s general location in terms of its distance from the center of the graph. After determining the general area in which the barrier will be located, the second step is to apply the methodology to find a minimum cut set in that general location. To determine where a minimum cut set is to be located, we drew two concentric circles around a central point in the area in question. The source of “Opinion” is assumed to start at an unknown location outside the outer

circle, with the intention of reaching an unknown location inside the inner circle. The actual path network considered consists of those edges that have at least one node (i.e., endpoint) between these two concentric circles. The second step in our algorithm is to find a minimum cut set for the graph whose segments have one or both nodes between these boundary lines. This is a well-known problem in graph theory, and it might be expected that there would be many solvers that could be used to obtain a solution to it. However, all but one of the solvers we considered could not find minimum cut sets in networks as large as the Twitter network surrounding "Obama". One, the GNET solver, could do so. Several solvers were also Excel-based, and so were restricted by the size limitations in capacity. Others could only accept the problem in the form of a general linear program and their LP-interfaces were unable to handle the problem. As a result, it took significant effort for us to identify just one solver that could find minimum cut sets in networks as large as the Obama's network. That one was the GNET solver [2]. GNET is designed to handle very large networks, and our experience so far is that it can handle networks of over one million arcs. To let the max-flow algorithm find a minimal cut set, the following structure was used. Each of the segments with both nodes inside the ring was given a capacity of counts as described before. An artificial "super-source" node was added outside the outer ring, and an artificial "super-sink" node was added inside the inner ring. The outer node of each segment crossing the outer ring was changed to this super-source node and these segments were given an infinite capacity. Similarly, the inner node of each segment crossing the inner ring was changed to this super-sink node, and these segments were also given an infinite capacity. Finally, an artificial road segment going from the super-sink to the super-source was added, also with an infinite capacity.

We then solved this network for its maximum flow and, hence, for a minimum cut set. This minimum cut set gives the smallest number of users that must be there in order to ensure that any "opinion" attempting to penetrate the inner circle, starting on any node from outside of the outer circle, will necessarily encounter the "Candidate" node. It should be noted that there could be more than one minimum cut set, and if there is more than one, the cut sets may or may not include some of the same nodes.

```

algorithm double_scaling;
begin
  set  $x := 0$ ;  $v := 0$ ;
  set  $\Delta := 2^{\lfloor \log U \rfloor}$ ;
  set  $\epsilon := \frac{1}{n+1}$ ;
  while  $\Delta \geq 1$  do
    augment_flow_double;
    set  $\Delta := \Delta/2$ ;
  end while
  if  $(x, \pi)$  is not  $\epsilon$ -optimal with  $\epsilon < \frac{1}{n}$  then
    reoptimize_flow_double;
  end if
  set  $P := \text{find\_admissible\_path}(G(x), \epsilon, s, t)$ ;
  augment  $\delta := (D - cx)/[\pi(s) - \pi(t)]$  units of flow along  $P$  and along  $x_{ts}$ ;
  update  $x, cx, v$ ;
end

```

Figure 3. An extract from opinion spread algorithm

## B. Influence Spread

Social Networks tools such as Klout [3] measure one's influence on a network of followers based on his/her ability to drive action within a certain exchange. Kred [4] analyze tweets over the last 1000 days of an exchange. It analyzes what others do because of you. Influence increases when others take action because of one's content. Peer Index [5] measure how active, receptive audience within an exchange. Tweepar [6] computes the ratio of number of followers to the number following.

### Algorithm 1: Hybrid maximum flow algorithm

```

1: (Apply preprocessing operations)
2: Label vertices
3: repeat
4:   Depth-restricted flow augmentation
5:   Update vertex labels after  $r$  augmentations
6: until no augmenting path with prescribed length between  $s$  and  $t$  is found
7: Switch to 'double tree' or 'push/relabel' strategy
8: (Undo preprocessing operations (reroute flow))

```

Figure 4. Influence spread algorithm

Fig. 4 shows a pseudo code of the influence algorithm that computes a numeric score for a presidential candidate. It measures the volume of waves (cascading of information flow) that his/her posts generate over the Social Network. The algorithm uses statistical indicators to measure "Influence" of Election Candidates/Voters in Twitter. We compute for each *Candidate* his/her indicators using the following 20 indices:

- 1- No of Tweets he sends a day, 2- No of mentions he receives a day, 3- No of Re-tweets his followers make a day, 4- No of people he follows, 5- Eigen-Factor: Weight of "highest-count-of-followers" person who ever Re-Tweeted one of his Tweets, 6- Rate of Followers growth, 7- Cascade measure: Breadth and Depth of spread of his Tweets (through Re-Tweets), 8- Diffusion measure: Breadth and depth of spread of his name (through mentions), 9- Degree Centrality: no of neighbors in his graph of followers up to 3-levels deep, 10- Betweenness Centrality: Exist on the shortest paths of all other node-pairs of his 3-levels deep neighbors graph, 11- Closeness Centrality: Has the shortest paths to every other node in his 3-levels deep neighbors graph, 12- Engaged Audience measure: Summation of "Direct Messaging"+"Mentions"+"Re-Tweets", 13- How Active measure: Time since he joined Twitter & Time he spent on Twitter a week, 14- Eigenvector Centrality: Higher scoring nodes contributes more to his score, 15- Hub measure: Arabic- English- French hub, 16-Visibility measure: Best visibility of what is happening in the network, 17- Persuasion measure: His convincing power to internalize the argument into his audience and make them adapt it (on exchange of Tweets), 18- Impression measure: His process to influence audience by regulating and controlling the exchange of Tweets (reputation), 19- Gallop measure: His efforts in marketing himself and ideas, 20- Bit-ly URL *interestingness*: How much interesting his URL links

## C. Trust Spread

By comparing the opinion spread graph and the influence spread graph we can produce a trust spread

graph. Fig. 5 shows the “Trust Spread” algorithm. This algorithm has two versions: Binary and Continuous. Its basic structure is: Source polls neighbors for trust value of sink, then the source computes the weighted average of these values to come up with an inferred trust rating; when polled, neighbors return either their direct rating for the sink, or they apply the algorithm themselves to compute a value and return it.

```

CAPACITY-SCALING
1   $x \leftarrow 0$ 
2   $\Delta \leftarrow 2^{\lceil \log_2 U \rceil}$ 
3  while  $\Delta > 0$ 
4    do while in  $G_x$  exists path  $s \rightsquigarrow t$  with the capacity of at least  $\Delta$ 
5      do find an augmenting path  $P$  with the capacity of at least  $\Delta$ 
6         $\delta \leftarrow \min\{r_{ij} : (i, j) \in P\}$ 
7        augment  $\delta$  units of flow along  $P$ 
8        update  $G_x$ 
9       $\Delta \leftarrow \Delta/2$   $\triangleright$  Integer division
10 return  $x$ 

procedure successive shortest path
begin
   $X = \min\{H(X) | X \in \mathbb{Z}^E\}$ 
   $\pi = 0$ 
  while there is a node  $s$  with  $g_s(X) > 0$ 
    compute the reduced costs  $c^\pi(X)$ 
    determine the shortest path distances  $d(i)$  from  $s$  to all other nodes
      with respect to the reduced costs
    choose a node  $t$  with  $g_t(X) < 0$ 
    let  $w(s, t)$  denote the shortest path vector from  $s$  to  $t$  in  $D$ 
     $X = X + w(s, t)$ 
     $\pi = \pi - d$ 
end
end
    
```

Figure 5. The trust spread algorithm

IV. ANALYTIC TOOL ARCHITECTURE

Fig. 6 shows the system architecture of the proposed analytic tool with metrics dashboard.

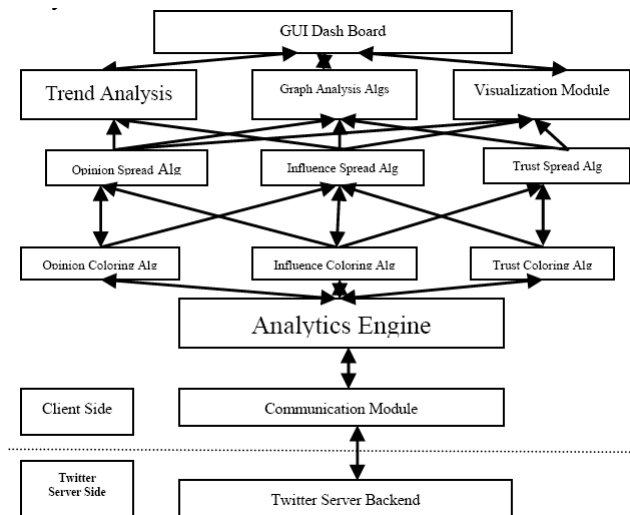


Figure 6. System architecture

The “Analytic Engine” communicates with the Communication Module to call the Twitter Server Backend using Twitter APIs to retrieve qualified Tweets for analysis. The retrieved Tweets are passed by the Engine to appropriate analysis module according to the

End user needs. Built-in analysis algorithms (as described in the previous sections) are called upon to process these Tweets and display their results either through the Visualization Module or the Graph Analysis Algorithms. As shown in the figure, there is interaction among the analysis algorithms and they call each others for intermediate processing. Trend Analysis is a module that accumulates analysis results over a period of time and performs trend analysis for future prediction.

V. PROTOTYPE IMPLEMENTATION

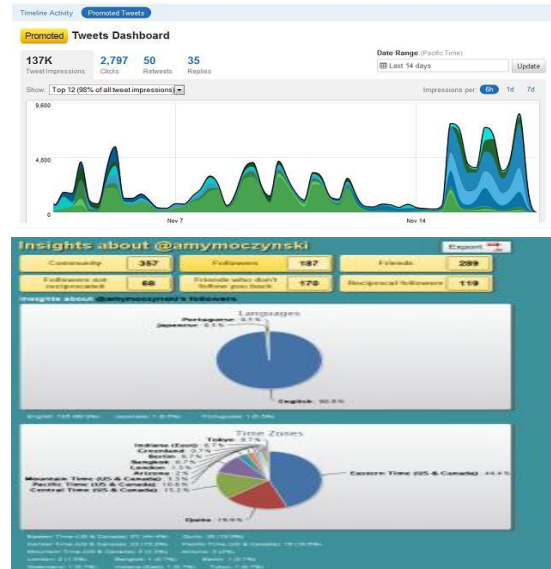


Figure 7. Two screenshots of the analytic tool dashboard

A prototype of the analytic tool with metrics dashboard is implemented on-top of “Nodexl” [7]; an open source template for Microsoft Excel that allows automated connection to a social network server and import (Using Twitter APIs) any data stream into the usual Excel environment. Tweet coloring and Edge coloring algorithms are implemented as Excel Macros with selective setting to either surface, shallow, or deep analysis parameters. Influence is measured through some twenty graph related terms such as: Degree centrality, Betweenness centrality, closeness centrality, Bit-ly interestingness, visibility measure, cascade measure, etc. Visualization graphs are provided that allow dynamic filtering, vertex grouping, adjusted appearance (zoom into areas of interest), graph metric calculations, etc. Each analytic algorithm will give indication(s) of where the candidate polling numbers are heading, based on its crowdsourced data-set and its analytic algorithm. Each analytic algorithm provides sentiment analysis results with human classification of the sentiments expressed in tweets."By illustrating instances when unprompted, natural conversation deviates from responses to specific survey questions, the algorithm helps capture the nuances of public opinion. The Indices are updated five times daily. Fig. 7 shows two screenshots of the implemented prototype.

VI. EXPERIMENTAL SYSTEM



Presidential candidates are heavy weight users of Social Networks. For example, Barak Obama has 14 Million Followers on his Twitter account, and 10 Million likes on his Facebook account. We have applied our analytic tool on the three elections of 2012- the American, the French, and the Egyptian. We have implemented the above Tweet and Edge coloring algorithms are macros in our tool. Fig. 8 shows part of the shallow Tweeter coloring Macro.

Fig. 9 shows part of the surface Tweeter coloring Macro.

Fig. 10 shows part of the "Deep Twitter Coloring" Macro, while Fig. 11 shows part of the opinion spread edge coloring Macro.

<pre>Public Sub Shallow-Tweet-Coloring()     Dim ie As Object     Dim WebPage As String     Dim wb As Workbook     Dim ws As Worksheet     Dim vWord As String     Dim current As Workbook     Dim rng As Range     Dim lrow As Long     Dim Rw_Start As Long     Dim Colored_Text As String     Dim Transl_Text As String     Dim MyText As String     Application.ScreenUpdating = False     Set current = ActiveWorkbook     lrow = current.Sheets(2).UsedRange.Rows.Count     Shallow-Tweet-Coloring_Text = Application.InputBox("Enter the Row no. for Coloring Or 'A' for all.", "Initialization")     If Shallow-Color_Text = "A" Then         Rw_Start = 3     ElseIf Shallow_Text = "" Then         Exit Sub     Else         lrow = CLng(Shallow-Color_Text)         Rw_Start = lrow     End If     Set Coloring_Text = Nothing     End If     For Shallow("fa" in the url = Shallow, and if using Shallow Coloring it will be "iw")         For i = Rw_Start To lrow             MyText = ""             Set ie = CreateObject("InternetExplorer.Application")             vWord = ActiveWorkbook.Sheets(2).Range("AM" &amp; i).Value             WebPage = "http://Shallow-Coloring.com/#falar" &amp; ActiveWorkbook.Sheets(2).Range("AM" &amp; i).Value             ie.Visible = False             ie.Navigate WebPage             Application.Wait (Now + TimeValue("0:00:4"))             Do Until ie.ReadyState = 4 For STATE as COMPLETE                 ie.document.getElementById("result_box").innerHTML             For j = 0 To UBound(Color_Text)                 If InStr(1, color_Text(j), "&lt;", vbTextCompare) &gt; 0 Then                     Transl-shallow-coloring_Text = Split(Translated_Text(j), "&lt;")                     MyText = MyText &amp; " " &amp; Transl_Text(0)                 End If             Next j         Next i     End Sub</pre>	<pre>Public Sub Deep-Color-Tweets()     Dim ie As Object     Dim WebPage As String     Dim wb As Workbook     Dim ws As Worksheet     Dim vWord As String     Dim current As Workbook     Dim rng As Range     Dim lrow As Long     Dim wb As Workbook     Dim ws As Worksheet     Dim i, j As Single     Set Application.ScreenUpdating = False     Set current = ActiveWorkbook     lrow = current.Sheets(2).UsedRange.Rows.Count     Deep-Coloring_Text = Application.InputBox("Enter the Row no. for Deep-Color Or 'A' for all.", "Initialization")     If Colored_Text = "A" Then         Rw_Start = 3     ElseIf Deep_Text = "" Then         Exit Sub     Else         lrow = CLng(Deep-Color_Text)         Sub StopFlash()             Application.OnTime NextTime, "StartFlash", schedule:=False             ActiveWorkbook.Styles("Flashing").Font.ColorIndex = xlAutomatic         End Sub         'Registry API functions         Private Declare Function RegCloseKey Lib "advapi32.dll" (ByVal hKey As Long) As Long         Private Declare Function RegCreateKey Lib "advapi32.dll" Alias "RegCreateKeyA" (ByVal hKey As Long, ByVal lpSubKey As String, phkResult As Long) As Long         Private Declare Function RegOpenKeyEx Lib "advapi32.dll" Alias "RegOpenKeyExA" (ByVal hKey As Long, ByVal lpSubKey As String, ByVal ulOptions As Long, ByVal samDesired As Long, phkResult As Long) As Long         Private Declare Function RegQueryValue Lib "advapi32.dll" Alias "RegQueryValueEx" (ByVal hKey As Long, ByVal lpValueName As String, ByVal lpReserved As Long, lpType As Long, lpData As Any, lpcbData As Long) As Long</pre>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Figure 8. The shallow tweeter coloring Macro

<pre>Public Sub Surface-Color()     Dim ie As Object     Dim WebPage As String     Dim wb As Workbook     Dim ws As Worksheet     Dim vWord As String     Dim current As Workbook     Dim rng As Range     Dim lrow As Long     Sub ProcessColumn()         Do Until ActiveCell.Value = ""             If ActiveCell.Value &gt; 10 Then                 Selection.Interior.Pattern = xlCrissCross             Else                 Selection.Interior.Pattern = xl             End If             Selection.Surface-color.Offset(1, 0).Range("A1").Select         Loop     End Sub     Sub ListWorkbooks()         Dim wb As Workbook         Dim ws As Worksheet         Dim i, j As Single         Set ws = Sheets.Add         For j = 1 To Workbooks.Count             Range("A1").Cells(j, 1) = Workbooks(j).Name             For i = 1 To Workbooks(j).Sheets.Count                 Range("A1").Cells(j, i + 1) = Workbooks(j).Sheets(i).Name             Next i         Next j     End Sub</pre>	<pre>ActiveWorkbook.Sheets(2).Range("AB" &amp; i).Value = MyText Set Surface-Color_Text = Nothing Close_IE Set ie = Nothing Next i End Sub 'Close_IE to close all instances of IE. Function Close_IE()     Dim objWMI As Object, objProcess As Object, objProcesses As Object     Set objWMI = GetObject("winmgmts://.")     Set objProcesses = objWMI.ExecQuery(_ "SELECT * FROM Win32_Process WHERE Name = 'explorer.exe'")     For Each objProcess In objProcesses         Call objProcess.Terminate     Next     Set objProcesses = Nothing: Set objWMI = Nothing     End Function ie.document.getElementById("result_box").innerHTML Colored_Text = Split(Colored_Text, "&gt;") For j = 0 To UBound(Colored_Text)     If InStr(1, Color_Text(j), "&lt;", vbTextCompare) &gt; 0         Public Function EnvironmentVariable(strIn As String) As String             'Returns the environment variable set on the computer             'EnvironmentVariable("windir") 'Windows folder             'EnvironmentVariable("temp") 'Temp folder             'EnvironmentVariable("username") 'Username             'EnvironmentVariable("computername") 'Computer name             EnvironmentVariable = Environ(strIn)         End Function</pre>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Figure 9. The surface tweeter coloring Macro

Fig. 12 shows part of the influence spread edge coloring Macro.

Fig. 13 shows part of the trust spread edge coloring Macro.

Visualization graphs are provided that allow dynamic filtering, vertex grouping, adjusted appearance (zoom into areas of interest), graph metric calculations, etc.

<pre>Public Sub Opinion-Spread()     Dim current As Workbook     Dim rng As Range     Dim lrow As Long     Dim Rw_Start As Long     Dim Opinion_Text As String     Dim Opposite_Text As String     Dim MyText As String     Application.ScreenUpdating = False     Set current = ActiveWorkbook     lrow = current.Sheets(2).UsedRange.Rows.Count     Color_Edge = Application.InputBox("Enter the Row no. for Opinion-Spread Or 'A' for all.", "Initialization")     If Colored_Text = "A" Then         Rw_Start = 3     ElseIf Colored_Text = "" Then         Exit Sub     Else         lrow = CLng(Opinion-Color_Edge)         Public Function CalculatePermutations(IngNumberElements As Long, lngNumberPositions) As Double             'Calculate the number of possible arrangements where the order matters of the number of elements into the number of positions             CalculatePermutations = Application.WorksheetFunction.Permut(IngNumberElements, lngNumberPositions)         End Function         Public Function EnvironmentVariable(strIn As String) As String             'Returns the environment variable set on the computer             'EnvironmentVariable("windir") 'Windows folder             'EnvironmentVariable("temp") 'Temp folder             'EnvironmentVariable("username") 'Username             'EnvironmentVariable("computername") 'Computer name             EnvironmentVariable = Environ(strIn)         End Function</pre>	<pre>'For Find Window and bring to top Private Declare Function AttachThreadInput Lib "user32" (ByVal idAttach As Long, ByVal idAttachTo As Long, ByVal fAttach As Long) As Long Private Declare Function GetForegroundWindow Lib "user32" () As Long Private Declare Function GetWindow Lib "user32" (ByVal hWnd As Long, ByVal wCmd As Long) As Long Private Declare Function GetWindowText Lib "user32" Alias "GetWindowTextA" (ByVal hWnd As Long, ByVal lpString As String, ByVal cch As Long) As Long Private Declare Function GetWindowTextLength Lib "user32" Alias "GetWindowTextLengthA" (ByVal hWnd As Long) As Long Private Declare Function GetWindowThreadProcessId Lib "user32" (ByVal hWnd As Long, lpdwProcessId As Long) As Long Private Declare Function IsIconic Lib "user32" (ByVal hWnd As Long) As Long Private Declare Function SetForegroundWindow Lib "user32" (ByVal hWnd As Long) As Long Private Declare Function ShowWindow Lib "user32" (ByVal hWnd As Long, ByVal nCmdShow As Long) As Long 'Used by the ShowWindowAfterDelay sub Dim strFindWindowTitle As String Dim intFindExitExcel As String Public Function CalculateCombinations(IngNumberElements As Long, lngNumberPositions) As Double     'Calculate the number of possible arrangements where the order does not matter of the number of elements into the number of positions     CalculateCombinations = Application.WorksheetFunction.Combin(IngNumberElements, lngNumberPositions)     End Function Public Function CalculateInterestPayment(dblInterestRate As Single, lngPaymentsPerYear As Long, lngNumPayments As Long, sglCurrentValue As Single) As Single     'Calculates the interest payments on a given principal     CalculateInterestPayment = Abs(Application.WorksheetFunction.Pmt(dblInterestRate / lngPaymentsPerYear, lngNumPayments, sglCurrentValue))     End Function</pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Figure 10. Shows part of the deep Tweeter coloring Macro.

<pre>Public Sub Opinion-Spread()     Dim current As Workbook     Dim rng As Range     Dim lrow As Long     Dim Rw_Start As Long     Dim Opinion_Text As String     Dim Opposite_Text As String     Dim MyText As String     Application.ScreenUpdating = False     Set current = ActiveWorkbook     lrow = current.Sheets(2).UsedRange.Rows.Count     Color_Edge = Application.InputBox("Enter the Row no. for Opinion-Spread Or 'A' for all.", "Initialization")     If Colored_Text = "A" Then         Rw_Start = 3     ElseIf Colored_Text = "" Then         Exit Sub     Else         lrow = CLng(Opinion-Color_Edge)         Public Function CalculatePermutations(IngNumberElements As Long, lngNumberPositions) As Double             'Calculate the number of possible arrangements where the order matters of the number of elements into the number of positions             CalculatePermutations = Application.WorksheetFunction.Permut(IngNumberElements, lngNumberPositions)         End Function         Public Function EnvironmentVariable(strIn As String) As String             'Returns the environment variable set on the computer             'EnvironmentVariable("windir") 'Windows folder             'EnvironmentVariable("temp") 'Temp folder             'EnvironmentVariable("username") 'Username             'EnvironmentVariable("computername") 'Computer name             EnvironmentVariable = Environ(strIn)         End Function</pre>	<pre>'For Find Window and bring to top Private Declare Function AttachThreadInput Lib "user32" (ByVal idAttach As Long, ByVal idAttachTo As Long, ByVal fAttach As Long) As Long Private Declare Function GetForegroundWindow Lib "user32" () As Long Private Declare Function GetWindow Lib "user32" (ByVal hWnd As Long, ByVal wCmd As Long) As Long Private Declare Function GetWindowText Lib "user32" Alias "GetWindowTextA" (ByVal hWnd As Long, ByVal lpString As String, ByVal cch As Long) As Long Private Declare Function GetWindowTextLength Lib "user32" Alias "GetWindowTextLengthA" (ByVal hWnd As Long) As Long Private Declare Function GetWindowThreadProcessId Lib "user32" (ByVal hWnd As Long, lpdwProcessId As Long) As Long Private Declare Function IsIconic Lib "user32" (ByVal hWnd As Long) As Long Private Declare Function SetForegroundWindow Lib "user32" (ByVal hWnd As Long) As Long Private Declare Function ShowWindow Lib "user32" (ByVal hWnd As Long, ByVal nCmdShow As Long) As Long 'Used by the ShowWindowAfterDelay sub Dim strFindWindowTitle As String Dim intFindExitExcel As String Public Function CalculateCombinations(IngNumberElements As Long, lngNumberPositions) As Double     'Calculate the number of possible arrangements where the order does not matter of the number of elements into the number of positions     CalculateCombinations = Application.WorksheetFunction.Combin(IngNumberElements, lngNumberPositions)     End Function Public Function CalculateInterestPayment(dblInterestRate As Single, lngPaymentsPerYear As Long, lngNumPayments As Long, sglCurrentValue As Single) As Single     'Calculates the interest payments on a given principal     CalculateInterestPayment = Abs(Application.WorksheetFunction.Pmt(dblInterestRate / lngPaymentsPerYear, lngNumPayments, sglCurrentValue))     End Function</pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Figure 11. Shows part of the opinion spread edge coloring Macro.

```

Registry API functions
Private Declare Function Influence-Spread-
RegCloseKey Lib "advapi32.dll" (ByVal hKey
As Long) As Long
Private Declare Function RegCreateKey Lib
"advapi32.dll" Alias "RegCreateKeyA" (ByVal
hKey As Long, ByVal lpSubKey As String,
phkResult As Long) As Long
Private Declare Function RegOpenKeyEx Lib
"advapi32.dll" Alias "RegOpenKeyExA"
(ByVal hKey As Long, ByVal lpSubKey As
String, ByVal ulOptions As Long, ByVal
samDesired As Long, phkResult As Long) As
Long
Private Declare Function RegQueryValueEx Lib
"advapi32.dll" Alias "RegQueryValueExA"
(ByVal hKey As Long, ByVal lpValueName As
String, ByVal lpReserved As Long, lpType As
Long, lpData As Any, lpcbData As Long) As
Long
Private Declare Function RegSetValueEx Lib
"advapi32.dll" Alias "RegSetValueExA" (ByVal
hKey As Long, ByVal lpValueName As String,
ByVal lpReserved As Long, ByVal dwType As
Long, lpData As Any, ByVal cbData As Long)
As Long
'Note that if you declare the lpData
parameter as String, you must pass it By Value.

INI API functions
Public Declare Function GetPrivateProfileString
Lib "kernel32" Alias "GetPrivateProfileStringA"
(ByVal lpApplicationName As String, ByVal
lpKeyName As Any, ByVal lpDefault As String,
ByVal lpReturnedString As String, ByVal nSize
As Long, ByVal lpFileName As String) As Long
Public Declare Function
WritePrivateProfileString Lib "kernel32" Alias
"WritePrivateProfileStringA" (ByVal
lpApplicationName As String, ByVal
lpKeyName As Any, ByVal lpString As Any,
ByVal lpFileName As String) As Long

'For Find Window and bring to top
Private Declare Function AttachThreadInput Lib
"user32" (ByVal idAttach As Long, ByVal
idAttachTo As Long, ByVal fAttach As Long) As
Long
Private Declare Function GetForegroundWindow
Lib "user32" () As Long
Private Declare Function GetWindow Lib
"user32" (ByVal hWnd As Long, ByVal wCmd
As Long) As Long
Private Declare Function GetWindowText Lib
"user32" Alias "GetWindowTextA" (ByVal hWnd
As Long, ByVal lpString As String, ByVal cch
As Long) As Long
Private Declare Function GetWindowTextLengthA
(ByVal hWnd As Long) As Long
Private Declare Function
'Used by the Show-Influence-Spread-
WindowAfterDelay sub
Dim strFindWindowTitle As String
Dim intFindExitExcel As String

Public Function
CalculateCombinations(IngNumberElements As
Long, IngNumberPositions) As Double
'Calculate the number of possible arrangements
where the order does not matter of the number of
elements into the number of positions
CalculateCombinations =
Application.WorksheetFunction.Combin(IngNumb
erElements, IngNumberPositions)
End Function

Public Function
CalculateInterestPayment(IngNumberElements As
Long, IngNumberPositions) As Double
'Calculate the number of possible arrangements
where the order matters of the number of elements
into the number of positions
CalculatePermutations =
Application.WorksheetFunction.Permut(IngNumb
erElements, IngNumberPositions)
End Function

Public Function
CalculateInterestPayment =
Abs(Application.WorksheetFunction.Pmt(dblInter
estRate / IngPaymentsPerYear, IngNumPayments,
sglCurrentValue))
End Function

Public Function
CalculatePermutations(IngNumberElements As
Long, IngNumberPositions) As Double
'Calculate the number of possible arrangements
where the order matters of the number of elements
into the number of positions
CalculatePermutations =
Application.WorksheetFunction.Permut(IngNumb
erElements, IngNumberPositions)
End Function
    
```

Figure 12. The influence spread edge coloring Macro

```

Public Function Trust-Speard-Edge-
Coloring-EnvironmentVariable(strIn
As String) As String
'Returns the environment variable set
on the computer
'EnvironmentVariable("windir")
'Windows folder
'EnvironmentVariable("temp")
'Temp folder
'EnvironmentVariable("username")
'Username

EnvironmentVariable("computername")
'Computer name
EnvironmentVariable =
Environment(strIn)
End Function

Private Function ForceForeground-
Color-Window(ByVal hWnd As Long)
As Long
Dim lngThreadID1 As Long
Dim lngThreadID2 As Long
Dim lngResult As Long

If hWnd = GetForeground-Color-
Window() Then
ForceForegroundWindow = True
Else
lngThreadID1 =
GetWindowThreadProcessId(GetForeg
roundWindow, ByVal 0&)
lngThreadID2 =
GetWindowThreadProcessId(hWnd,
ByVal 0&)

If lngThreadID1 <> lngThreadID2
Then
Call
AttachThreadInput(lngThreadID1,
lngThreadID2, True)
lngResult =
SetForegroundWindow(hWnd)
Call
AttachThreadInput(lngThreadID1,
lngThreadID2, False)
Else
lngResult =
SetForegroundWindow(hWnd)
End If

If IsIconic(hWnd) Then
Call ShowWindow(hWnd,
SW_RESTORE)
Else
Call ShowWindow(hWnd,
SW_SHOW)
    
```

Figure 13. The Trust spread edge coloring Macro

A. Experiment I

We have applied our analytic tool on the three elections of 2012- the America, the French, and the Egyptian. Fig. 14 shows the Egyptian Election case.



Figure 14. The Egyptian election case study

B. Experiment II

Fig. 15 shows the American Election case.

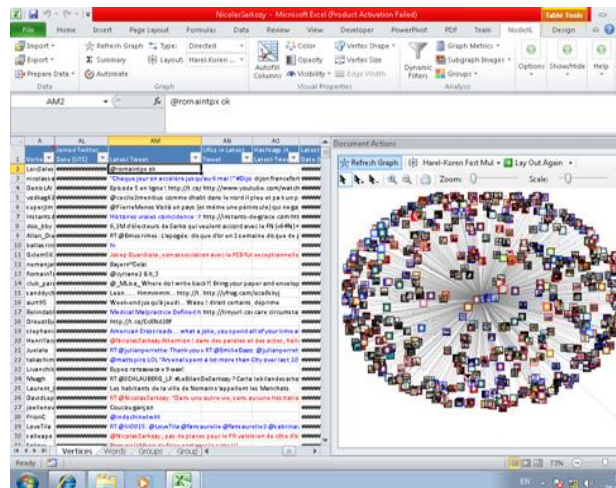


Figure 15. The American Election Case Study

C. Experiment III

Fig. 16 shows the French Election case. The proposed analytics polls tweets that mention the candidates, his followers, and their followers and analyze them. It tracks enthusiasm levels for both. Similar to an approval rating from an opinion poll, the experiment indices run from zero to 100; a 50 rating is neutral. The experiments are automatically run once a day to update the indices [8].

VII. RELATED WORK

Twitter has gotten into the candidate sentiment business, parsing the content of several million tweets a week to capture what is called the "nuances of public opinion" about the presidential hopefuls. The newly launched Twitter Political Index [9] offers a daily tally of



Twitter users' "feelings" about President Obama and his presumptive Republican opponent, Mitt Romney.

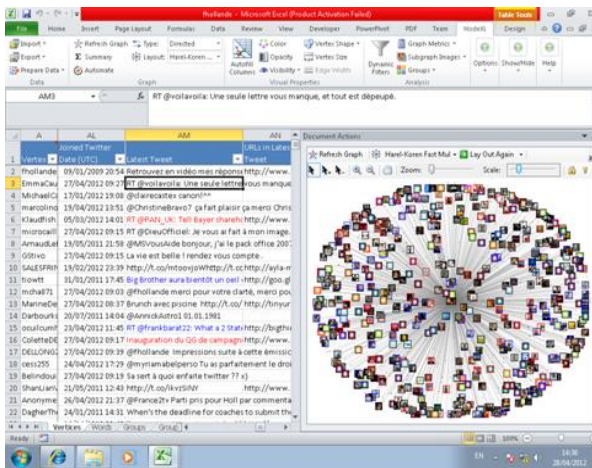


Figure 16. The French Election Case Study

Each day, the Index evaluates and weighs the sentiment of Tweets mentioning Obama or Romney relative to the more than 400 million Tweets sent on all other topics. For example, a score of 73 for a candidate indicates that Tweets containing their name or account name are on average more positive than 73 percent of all Tweets. It lends new insight into the feelings of the electorate, but is not intended to replace traditional polling -- rather, it reinforces it. Twitter expects that the Index will give an indication of where the candidate polling numbers are heading, based on its crowd sourced data-set. Twitter is working with the social-data analysis company Topsy and two polling firms, Mellman Group and North Star Opinion Research [10], to calculate the Index sentiment results. Topsy provides the sentiment analysis technology, which the company said delivers 90 percent concordance with human classification of the sentiments expressed in tweets. By illustrating instances when unprompted, natural conversation deviates from responses to specific survey questions, the Twitter Political Index helps capture the nuances of public opinion. The Index is updated daily at 5 p.m. PT.

In the following we analyze a number of most recent Twitter analytic tools. Twitter tools frequently come and go, especially the free ones, so a survey like this is always evolving. But here is a small collection of tools that several Twitter users like to use.

Twitter has created its own free analytics tool, however it is still in beta and not yet available to everyone. The data is fairly basic. It is not particularly useful yet. But it is always good to get data straight from the source and the toolset will continue to evolve. Topsy's Social Analytics tool [10] offers a good way to trend and compare mentions and replies for up to three profiles. In addition one can track links to a domain or mentions of a particular keyword. Topsy also provides data on the top links in the past 24 hours for each profile. Twitter Counter lets one compare the growth of followers, followings and tweet volume for up to three profiles. TweetStats [9] provides a nice overview of overall activity including tweet volume and density plus

aggregate daily and hourly tweets. It takes a little while for it to compile the data but it's worth the wait. TweetEffect [9] provides some interesting figures on how many followers were gained and lost after the most recent tweets. In addition to providing the numbers in aggregate it allows one to see the follower change after each individual tweet. Follower Wonk [9] provides some comparative analysis between profiles. In particular the Venn diagram that illustrates unique and shared relationships. Basic use of the tool is free but larger requests require the purchase of credits. Larger profiles tend to trigger the credit requirement fairly easily. The pros and cons of Klout have been widely covered [3]. But it does provide a simple way to benchmark a profile and compare it to others. Kred [4] is the new kid on the block that's been getting a lot of attention lately. Klout has a pretty big head start but Kred is positioning itself to become another popular option in this area. Twitalyzer [9] changed up its offering and now provides an overall summary of a profile (including data from other sources) plus a variety of performance scores via its metrics dashboard. TweetLevel from Edelman [9] has been around for quite a while. It offers another way to do some basic benchmarking. WordCURL [9] is a new tool that provides a series of nice charts for things like tweets per hour and day, most used words and most mentioned profiles, influence, popularity, engagement, trust, among others.

## VIII. CONCLUSION

The proposed analytic tool can serve as a Gallop for any "Ad Hoc" queries by incorporating relevant color-related algorithms to color code the answers in the Tweets and the edges in the graphs. It is also applicable to other domains of applications such as "Stock Market Predication," "Public Opinion," "Customer Voice," "Service Benchmarking," and "Blog Analysis". Our future work will add translation-to-Arabic capability from any language to Arabic, archive posts over a longer period of time for better history (finger print), and will tackle other social networks such as Facebook, Flickr, Youtube, etc.

## ACKNOWLEDGEMENT

This paper contains studies and results from the research project sponsored by Prince Megren Center at Prince Sultan University. The author would like to record for Megren Research Center his acknowledgement for supporting this research project.

## REFERENCES

- [1] A. Sameh, "A new twitter client for participating-in and mining social networks of other Languages/Cultures," in *Proc. IDA*, UK October 17-19, 2013.
- [2] Y. Kodama, T. Kudoh, and R. Takano, "GNET gigabit Ethernet Network Testbed," in *Proc. IEEE International Conference on Cluster Computing*, 2004.
- [3] D. Boyd, S. Golder, and G. Lotan, "Tweet, Tweet, Retweet: conversational aspects of retweeting on twitter," in *Proc. IEEE Hawaii International Conference on System. Sciences*, January 2010.



- [4] M. Cha, H. Haddadi, F. Benevenuto, and K. P. Gummadi, "Measuring user influence in Twitter: The million follower fallacy," in *Proc. International AAAI Conference on Weblogs and Social*, 2010.
- [5] D. M. Romero, W. Galuba, S. Asur, and B. A. Huberman, "Influence and passivity in social media," in *Proc. 20th ACM International World Wide Web Conference*, 2010.
- [6] P. S. Dodds and D. J. Watts, "A generalized model of social and biological contagion," *Journal of Theoretical Biology*, vol. 232, no. 4, pp. 587–604, 2005.
- [7] J. Forgas, *On Being Mood but Influential: The Role of Affect in Social Influence Strategies, Social Influence, Direct and Indirect Processes*, A. Sameh, A New Twitter Analytics Tool, Work in Progress, 2001
- [8] A. Gelman and J. Hill, *Data Analysis using Regression and Multilevel/Hierarchical Models*, Cambridge University Press, 2006.
- [9] A. Gill, S. Nowson, and J. Oberlander, "What are they blogging about? Personality, topic and motivation in blogs," in *Proc International Conference on Weblogs and Social*, 2009.



**Ahmed Sameh** Born in Alexandria November 9, 1957. Has B.Sc. M.Sc., and Ph.D. all in Computer Science from University of Alberta Canada 1979, 1985, and 1989 respectively. Major fields of interest are Data Mining, Parallel Computing, Neural Networks, Mobile Computing and Wireless Communication. Currently with Prince Sultan University in Riyadh- Saudi Arabia

as a Professor of Information Systems and Computer Science since 2009, on Sabbatical Leave from the American University in Cairo, also worked at George Washington University, University of Kuwait and Lewis and Clark College in Oregon USA, and the University of Alberta Canada. He has supervised more than 30 Master and Ph.D. students, published 15 Book chapters, 45 Journal papers, and more than 200 Conference papers.

Professor Sameh is a member of major Computer Science related societies such as IEEE, ACM, CPM, ISD, etc. He earned "Best Researcher" Google award for his lifelong contribution to the field of Computing in 2006. He can be reached using the following emails: [asameh@cis.psu.edu.sa](mailto:asameh@cis.psu.edu.sa) and [sameh.aucegypt.edu@gmail.com](mailto:sameh.aucegypt.edu@gmail.com)