Paper 324-2011

%GetTweet: A New SAS® Macro to Fetch and Summarize Tweets

Satish Garla and Goutam Chakraborty, Oklahoma State University, Stillwater, OK

ABSTRACT

The role of Twitter as a source of valuable information for spotting trends has been much talked about in the popular press. The open API in Twitter makes it one of the most sought after platforms for textual data analysis. While SAS® Text Miner provides a robust method for analyzing textual data, the challenge remains to fetch customized Tweets and clean textual data before any text mining.

This paper develops and discusses a new SAS® macro that can be used easily to fetch the Tweets a researcher wants from Twitter. The macro uses the search API in Twitter and the HTTP procedure in SAS to create a data set of Tweets that are customized using parameters such as combination of keywords, exact phrases, omission of specific words, and so on. The macro also purges terms such as http tags and URLs that might create problems in text mining. This paper also shows a visual analysis of "retweets" to identify influencers via network visualization graphs that are available in SAS/GRAPH® software.

INTRODUCTION

Social Media has gained attention in the past decade as a valuable source for information for businesses, governments, and nonprofit organizations across the world. The rate of growth in the numbers of users of social media sites is in the hundreds of thousands every day. The latest Twitter release claims 1,000 TPS (Tweets/Sec) and 12,000 QPS (Queries per sec), which equals over one billion queries a day! [1] Every second, someone is talking about a company or a brand or about the latest buzz in an industry. Thus, a treasure-trove of potential information is available from social media if the textual data can be easily accessed and cleaned. The cleaned textual data can then be used for sentiment analysis, cluster analysis, classification analysis and concept trending. [2]

In Global Forum 2010, Russell, Richard, and Ravi discuss a way to collect Tweets via specifying a hash word to a SAS data set using the HTTP procedure in SAS®. [3] They present an excellent methodology of exploring Tweets using SAS® Text Miner and other visualization tools available in SAS. In his blog, "The SAS Dummy," Chris Hemedinger also discusses a basic approach to collect Tweets on a specific topic as a SAS data set and run some basic analysis to quantify the responses. [3] We build on the work of these researchers to create the %GetTweet macro.

Before attempting to search for information on Twitter, it is essential to understand the Twitter language. The Tweet posted by a user could be: (1) a new Tweet, (2) a reply to someone else's Tweet, or (3) a re-posting (often called Retweet) of a Tweet posted by someone else. If a user wishes to group posts by type or topic, he can use hashtags (prefix the topic with #). This method comes in handy given the 140 character limit on a Twitter post. However, this limit also encourages users to use shorthand, chat slang, symbols, URLs, and abbreviations that can cause major problems in text-mining. [3] Here is a sample Tweet from SAS Analytics.

#SAS #Analytics customer uncover ways 2 improve quality of care & reduce health care costs. http://bit.ly/d57Pkq (health care providers)

A retweet of the above Tweet by another user looks like the following.

RT @SASAnalytics: #SAS #Analytics customer uncover ways 2 improve quality of care & reduce health care costs. http://bit.ly/d57Pkq (health care providers)

Users often search postings using Twitter's advanced search page. Figure 1 shows search options provided by Twitter, which are currently captured in the %GetTweet macro.

%GETTWEET MACRO

This macro allows anyone to collect customized data from Twitter based on combinations of options as shown below. All the Tweets in the last week that match the search conditions are downloaded into a SAS data set. These conditions are specified as the keyword parameters in the Macro. The full Macro code is reported in the Appendix.

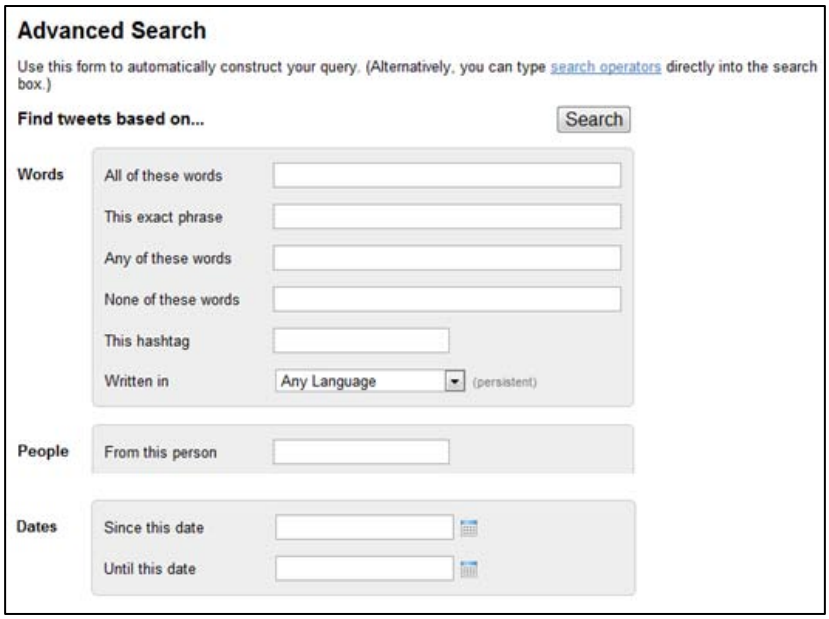


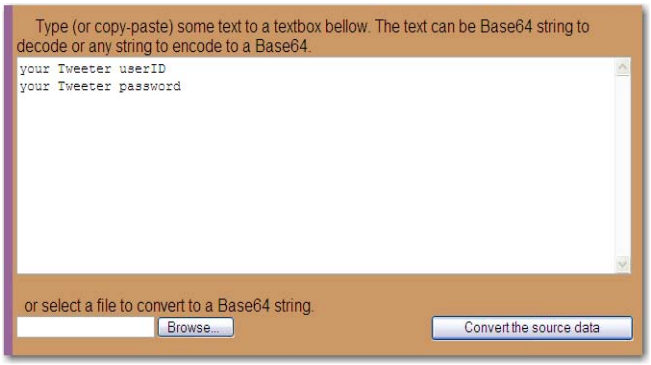
Figure 1. Advanced Search options available on Twitter that are captured in GetTweet Macro

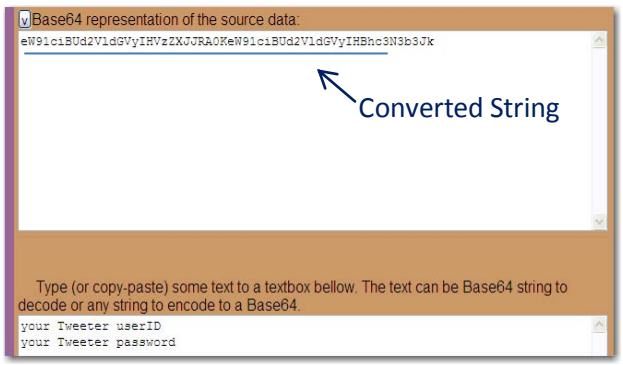
%GetTweet(WORDS=,PHRASE=,ANY=,NONE=,HASH=,FROM=,TO=,SINCE=,UNTIL=,QUESTION=,
CODE=,PATH=);
Where

WORDS=	(Mention all of the words you want to search. This is an AND condition)
SINCE=	(Enter From Date in the format: YYYY-MM-DD)
UNTIL=	(Enter To Date in the format: YYYY-MM-DD)
PHRASE=	(Enter Exact Phrase you want to search)
ANY=	(Enter Any of the words you want to search)
NONE=	(Enter the words you do not want to be in search results)
HASH=	(Enter the hash tag that you want in your results)
FROM=	(Enter name of the person who is Tweeting. Multiple names not allowed)
QUESTION=	(Enter 1 if you want only the Tweets with a Question Mark)
CODE=	(Enter base64 encoded string of Twitter login – explained below)
PATH=	(Directory where fetched data sets will be saved)

In general, Twitter returns up to a week of Tweets with a maximum of about 1,500 Tweets. If search results exceed 1,500 Tweets, only the most recent 1,500 will be kept. To collect Tweets for a complete week on terms that may exceed the 1,500 Tweets limit, one can use the SINCE and UNTIL parameters.

Many websites, including Twitter, require basic authentication to access their database. While there are multiple ways to do this, the macro discussed in this paper uses a base64 encoded Twitter username and password that is passed on as a part of the HTTP header in the PROC HTTP procedure. The base64 encoded string is usually in the form "dXhsdgfhsd….". To find the base64 encoded string for your own Twitter login and password, go to the website: http://www.motobit.com/util/ base64-decoder-encoder.asp , type your Twitter ID and Password in the box, and click "convert the source data" button as shown below.





To use the macro, you need to enter the converted string as one of the parameters. The following five examples demonstrate the utility of this macro.

EXAMPLE 1: Collecting Tweets that Mention the Word “Cancer”

%GetTweet (WORDS=Cancer, CODE= &authorization, PATH=&path);
The CODE= parameter is defined as a macro variable in the string format as shown below:
%let authorization=%nrstr("Authorization: c2F0aXNoZ2Fybx==");
The result of this search is shown in Figure 2.

Support Breast Cancer Survivors, add a #twibbon to your avatar now! - http://twibbon.com/join/Breast-Cancer
The cancer statistics are a lie: Cancer is becoming epidemic around the world http://j.mp/aMcBAE
RT @LoveScopes: The #Water signs #Scorpio #Pisces & #Cancer are motivated & stimulated by feelings,
Support Breast Cancer Survivors, add a #twibbon to your avatar now! - http://twibbon.com/join/Breast-Cancer

Figure 2. Partial results of Tweets with the word 'Cancer'

The search was intended to give results for the word “Cancer” referring to the disease. However, in Figure 2, results also contain the word “Cancer” referring to the astrological sign. Filtering out all irrelevant Tweets that do not meet requirements can be done by using the NONE option in the macro, as shown below. Thus, we recommend that users of this macro follow a two-step strategy. Collect initial Tweets using WORDS in step 1. Look at your results and determine unwanted terms. Call the macro again in step 2 and use the unwanted terms in NONE as shown below.

EXAMPLE 2: Collecting Tweets that Do Not Contain a Specific Set of Words

Repeat the same search for the word “Cancer” as in Example 1 and specify excluding Tweets that contain any of the words “Zodiac,” “Astrology,” “Scorpio,” or “Pisces.”

%GetTweet (WORDS=Cancer, NONE=Zodiac Scorpio Pisces Astrology, CODE=&authorization, PATH=&path);

Support Breast Cancer Awareness, add a #twibbon to your avatar now! - http://bit.ly/4iTMiW
The cancer statistics are a lie: Cancer is becoming epidemic around the world http://j.mp/aMcBAE
Support Breast Cancer Survivors, add a #twibbon to your avatar now! - http://twibbon.com/join/Breast-Cancer
@liprap That's funny--like when people say they "support breast cancer." And I'm like, "not exactly."

Figure 3. Partial Results of Tweets with word 'Cancer' used in the context of a disease

EXAMPLE 3: Collecting Tweets that Mention the Word “Cancer” on a Specific Date, 3rd Oct 2010

%GetTweet (WORDS=Cancer, SINCE=2010-10-03, UNTIL=2010-10-03, CODE= &authorization, PATH=&path);

The SINCE= and UNTIL= parameters help to restrict the search results based on a specified time period. The SINCE date value can go back as much as a week from the date for which you are searching. The UNTIL date cannot be in the future, and it must be greater than the SINCE date. The values should be entered in the format: YYYY-MM-DD.

2010-10-03T23:48:01	@stevereads I was thinking that if with discounting then breast cancer has many generation
2010-10-03T23:47:59	Insurance, Race and Poverty Affect #Cancer Care, Researchers Report - http://newzfor.me/?
2010-10-03T23:47:58	Support Breast Cancer Awareness, add a #twibbon to your avatar now! - http://twibbon.com

Figure 4. Partial Results of Tweets with word Cancer Tweeted on 3rd October 2010

EXAMPLE 4: Collecting Tweets in which Either the Words “SAS” or “Analytics” are mentioned

The use of an OR operation in a search can be accomplished using the parameter ANY=. Mention all words to be searched separated by spaces. Using the two words “SAS” and “Analytics” returns all the Tweets that mention either of these words.

%GetTweet(ANY=SAS Analytics, CODE= &authorization, PATH=&path);

The results in Figure 5 show Tweets that contain the word SAS referring to the context of SAS Airlines or the Radisson Blu (SAS) Strand Hotel in Stockholm. Again, these results can be filtered appropriately using the NONE parameter in the macro.

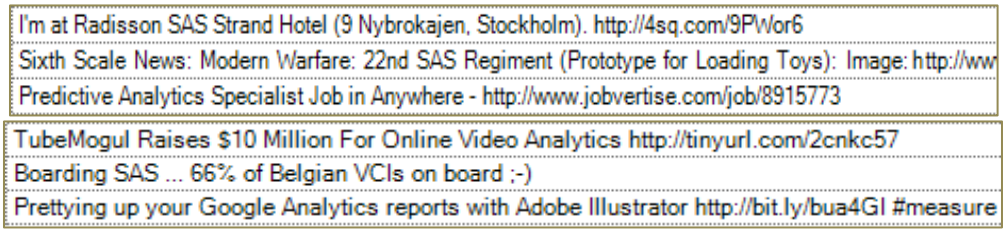


Figure 5. Partial Results of Tweets that contain either 'SAS' or 'Analytics'

Similarly, for collecting Tweets that have both the words “SAS” and “Analytics,” use the WORDS= and HASH= parameters with words separated by spaces as shown below.

%GetTweet (WORDS=SAS Analytics, HASH=SAS Analytics, CODE=&authorization, PATH=&path);

EXAMPLE 5: Collecting Tweets from a Specific User

To collect Tweets from a person or a user, specify the username of the Tweeter using the keyword FROM=. Figure 6 shows Tweets from BevBrown, Social Media Manager for SAS, in the month of October. You can fetch Tweets only for one user at a time.

%GetTweet(FROM=BevBrown, CODE= &authorization, PATH=&path);

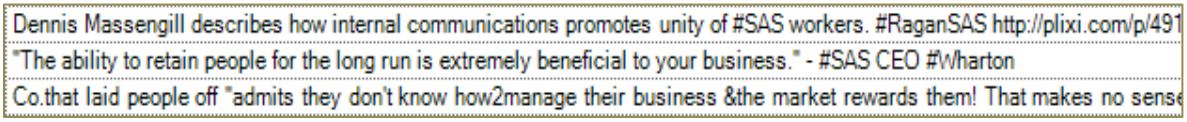


Figure 6. Partial Results of Tweets from a specific user

The keyword parameter QUESTION= can be used to fetch only those Tweets that contain a question mark. Combinations of all of the parameters in the macro create unlimited options to get the right kind of textual data for analysis.

TWEETS vs. RETWEETS

Once the Tweets are available as SAS data sets, all the tools and methodologies in SAS can be used for analysis. Russell, Richard, and Ravi discuss methods to analyze the Tweet data using frequency analysis, tag clouds, and monitoring frequency of Tweets over a period. [3] However, using the Tweet data directly in SAS® Text Miner may not reveal important patterns. Most of the methods used in SAS® Text Miner give weights to terms based on the term frequency in a single document and in all the documents. If a particular term appears more often in multiple documents, it is given a lower weight. This will likely cause problems when same Tweets are retweeted many times.

For example, suppose a Tweet “Rigorous exercise would help in curing breast cancer” is retweeted multiple times. Due to the high frequency of the term “exercise,” the algorithm will likely give the term low importance in associating it with cancer. Hence, retweets may need to be filtered out before use in Text Mining if the goal is to find patterns and associations among terms in documents. However, much valuable information can be obtained by analyzing retweets.

Word-of-mouth (or “word-of-mouse”) is considered a very powerful marketing tool; it has been used by many companies as one source to promote products. Any statement made by an opinion leader attracts attention and is often considered reliable. This message is spread widely and quickly across networks. The same phenomenon can be found on Twitter with users having a large number of followers. Tweets from this user have a high chance of being retweeted. The same principle continues with the followers of the followers. Careful examination of these retweets and understanding influencers and their networks will certainly help marketers with valuable insights. The strength of influence can be measured by the number of times a particular user’s Tweet is retweeted. [3] Here is an example.

Tweet: Monday Keynote Update ~ Bret Michaels: Music Icon, Actor, 'Celebrity Apprentice' Winner Keynotes #DMA2010 - http://bit.ly/bswBgX

Retweet: RT@DMA_USA Monday Keynote Update ~ Bret Michaels: Music Icon, Actor, 'Celebrity Apprentice' Winner Keynotes #DMA2010 - http://bit.ly/bswBgX

Re-Retweet: RT @neolane: RT@DMA_USA Monday Keynote Update ~ Bret Michaels: Music Icon, Actor, 'Celebrity Apprentice' Winner Keynotes #DMA2010 - http://bit.ly/bswBgX

And this can go on.

The above Tweets were collected using the search term "#DMA2010" on the Direct Marketing Association tradeshow in October 2010. The organizers used Twitter as one of the important communication platforms to inform event attendees about changes in the schedules, hall directions, etc.

The GetTweet macro does a basic data cleaning such as removing "http" tags, URLs, and stripping of characters in ID variables and creates two sets of data for analysis. One data set consists of all Tweets; the other only has retweets. Using this Tweet data set in SAS® Text Miner for Text Mining would yield better results compared to the actual data set. The retweet data can be used for analyzing influencers and for finding the frequently retweeted messages.

The macro also generates a two-page PDF report (Tweet Report) when Tweets are collected using any of the parameters in the macro except the FROM= parameter, i.e., no report is generated if the macro is used to fetch Tweets posted by a specific user since the charts on the report are irrelevant in this case.

Figure 7 shows the first page in the report generated for Tweets on the topic "#DMA2010". The report includes the following four graphs.

1. Top Ten Tweeters based on number of Tweets
2. Top Ten sources (Websites) used for posting Tweets
3. Top Tweeters whose Tweets were retweeted highest number of times (Top Influencers)
4. Number of Tweets posted on a daily basis

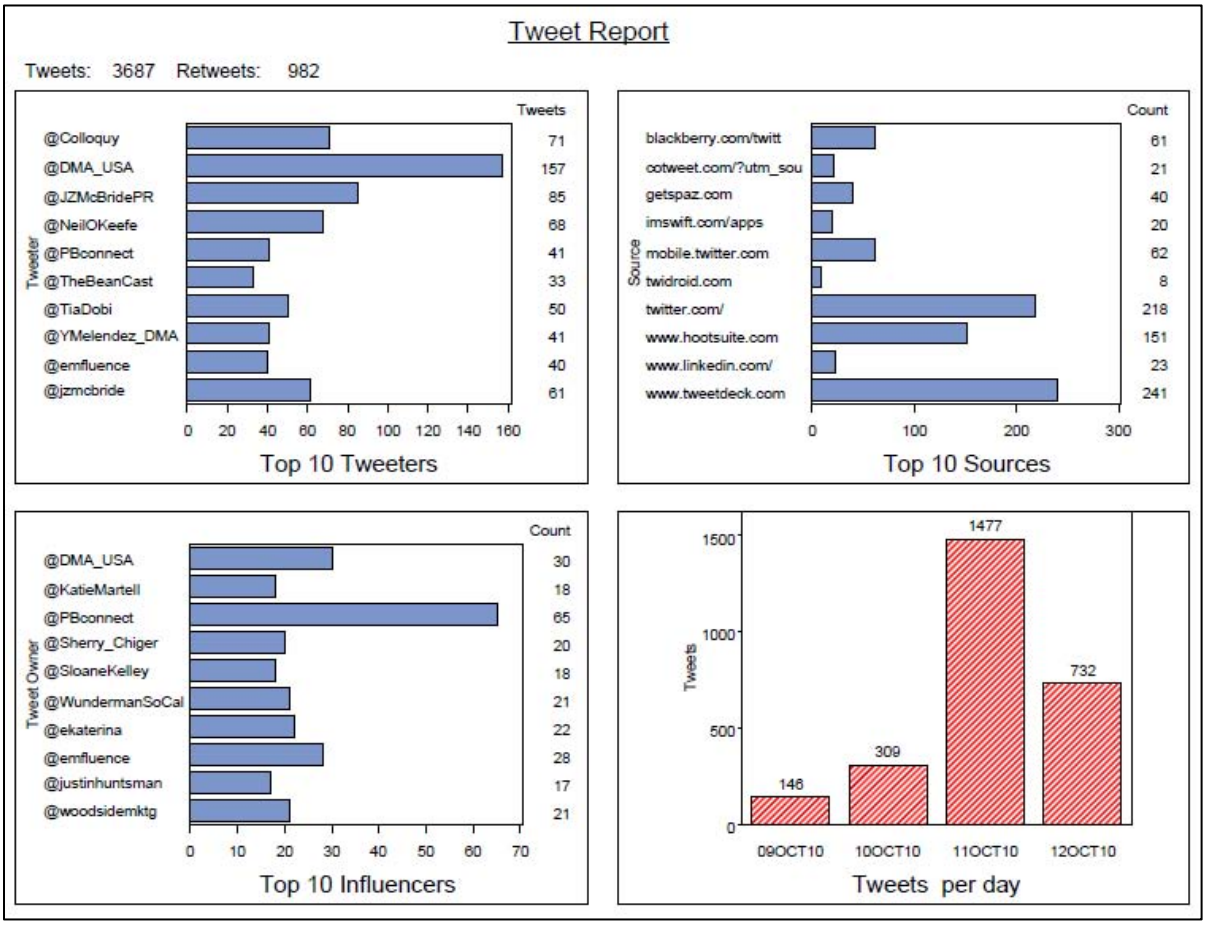


Figure 7. First Page in Summary Report on the topic "#DMA2010"

NOTE: For demonstration purposes, we used Tweets from multiple days in Figure 7. However, the macro generates a report for only up to 1,500 Tweets at a time.

In addition to identifying the most active Tweeters, any marketer would want to track what is being said and shared repeatedly about their brand or promotion. Page 2 of the Tweet Report would help to identify the Tweets that were retweeted highest number of times. Figure 8 shows a part of the top retweets report on the topic "#DMA2010".

Tweet	Retweets
Nearly 65 million users "Like" something online every day. #dma2010	29
"The only thing worse than not doing social marketing is doing it badly" - Brian Sheehan #DMA2010	13
You know you've got a good idea when people don't like it. - Steve Stoute (Best quote of the day) #dma2010	9
E-mail messages that drive all links to a single site have a 48% higher conversion rate. #dma2010	9
76% of online purchases are influenced by direct mail. #DMA2010 #DirectMarketing	9
Brands need to look beyond followers as the metric of social media success. #dma2010	9
Emails delivered at 7:00am had highest open rate - Jay #DMA2010	8
#DMA2010 Kimmel keynote- "we live in the age of marketing longtail..the characterization of behavioural targeting is pers	7

Figure 8. Partial "Top Retweets Report" on the topic #DMA2010

Insights on influencers can be extracted by visually exploring the Tweeter network. While there are many tools for network visualization, SAS/GRAPH® NV Workshop has excellent capabilities for exploring networks or links.

Figure 9 shows a Multilevel Force Visualization Network for DMA2010 data. In-depth analysis can be performed by highlighting a portion of the graph and using Zoom and Label features.

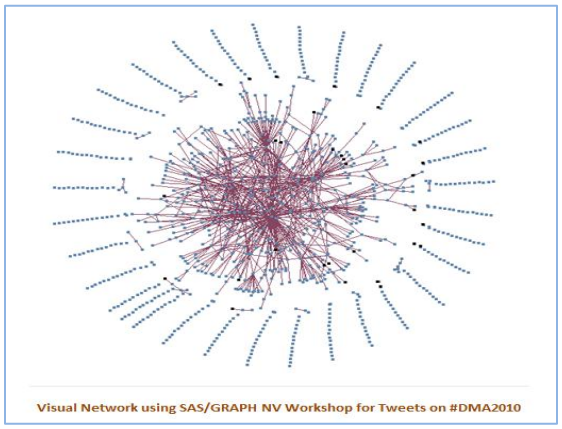


Figure 9. Multi-Level Force Network of Tweeters for #DMA2010

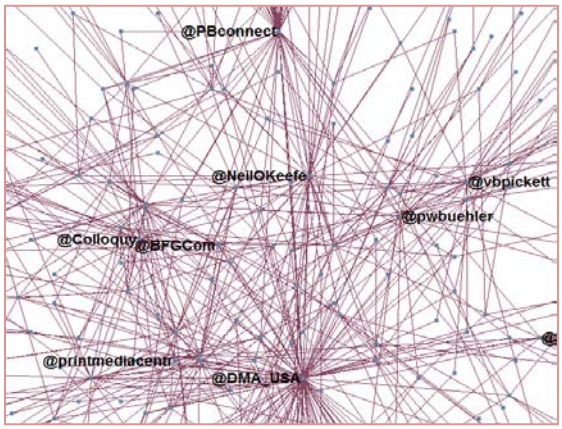


Figure 10. Highlighted Portion of the Network in Figure 9

The number of spokes originating from a node represents the number of times a particular user's Tweets were retweeted. Figure 10 shows an example of a highlighted portion of Figure 9. We can see that users @DMA_USA, @PBconnect, @vbpickett, and @NielOKeffe show strong activity in Tweeting about DMA2010. At this level, we can identify all major hubs in the network. These hubs can be more closely analyzed to further explore a user's network. The SAS/GRAPH® NV Workshop tool provides various valuable graphs and visualization networks that can be used in conducting in-depth analysis to better understand networks and links.

CONCLUSION

Twitter has become a significant data source for businesses. PROC HTTP and Twitter API can be used to tap this data and convert it to useful information via simple reports or text analytics. The difficulty lies in getting customized and cleaned data as input to text analytics. The GetTweet macro attempts to solve these problems by fetching specific Tweets using keywords, cleaning the Tweets of hash tags, URLs, etc. and generating summary reports. The output from this macro is a cleaned data set of Tweets that can be analyzed effectively via text mining. This macro can be further developed to include other search criteria available on Twitter Advanced Search page.

In addition, retweets data created by this GetTweet macro can be analyzed separately for better insights into the influence patterns of Tweeters. As we have demonstrated, SAS/GRAPH® NV Workshop provides excellent tools to visualize the network of such influencers. This helps any marketer to quickly identify Tweeters responsible for spreading information about their brands and their organizations.

REFERENCES

[1] "Twitter's New Search Architecture," Twitter Engineering, http://engineering.twitter.com/2010/10/ twitters-new-search-architecture.html, October 6th, 2010.

[2] "Concept Trending: A Glimpse into the Future?" Blog: Life Analytics-Practical Applications of Data Mining, Text Mining and Information Extraction <http://lifeanalytics.blogspot.com/search/label/ text%20mining>.

[3] Russell Albright, Richard Foley, and Ravi Devarajan, April 2010, "Listening to the Twitter Conversation," Proceedings SAS Global Forum 2010.

[4] "Mining the VP debate results according to Twitter -- with SAS," http://blogs.sas.com/sasdummy/ index.php?/archives/55-Mining-the-VP-debate-results-according-to-Twitter-with-SAS.html, October 3rd, 2008.

[5] "SAS and Twitter–How to Harness SAS to Grab Data from Twitter in 2 Easy Steps",Blog: The Business Intelligence Guru, http://bzintelguru.com/blog/sas-and-twitter-how-to-harness-sas-to-grab-data-from-twitter-in-2-easy-steps/comment-page-1/#comment-582, August 12th, 2010.

[6] Jennifer S. Harper, April 2009, "Mission Possible: Putting a Table and Multiple Graphs on a Single-Page PDF with ODS and Basic GOPTIONS", Proceedings SAS Global Forum 2009.

[7] "Exporting SAS/GRAPH Output to PDF Files from Release 8.2 and higher", TS-659, SAS Knowledge Base – Papers, http://support.sas.com/techsup/technote/ts659/ts659.html, March 18th, 2004.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors at:

Satish Garla, Oklahoma State University, Stillwater OK, Email: satish.garla@okstate.edu

Satish Garla is a Master's student in Management Information Systems at Oklahoma State University. He has three years of professional experience as Oracle CRM Consultant. He is SAS Certified Advanced Programmer for SAS® 9 and Certified Predictive Modeler Using SAS® Enterprise Miner 6.1

Dr. Goutam Chakraborty, Oklahoma State University, Stillwater OK, Email: goutam.chakraborty@ okstate.edu

Goutam Chakraborty is a professor of marketing and founder of SAS and OSU data mining certificate program at Oklahoma State University. He has published in many journals such as *Journal of Interactive Marketing*, *Journal of Advertising Research*, *Journal of Advertising*, *Journal of Business Research*, etc. He chaired the national conference for direct marketing educators in 2004 and 2005 and co-chaired the M2007 data mining conference. He is also a Business Knowledge Series instructor for SAS.

APPENDIX

```
/*************************************************************************************
Macro Name: GetTweet

Purpose: Macro to fetch customized Tweets from Twitter using Keywords

How it works:
       This Macro uses PROC HTTP to communicate with Twitter API. Twitter returns a
       maximum of roughly 1,500 tweets available in a week's time. The search method
       'atom' is used and so we get results in XML format. A XML Mapper code is used
       to create SAS data sets from fetched XML results.

Required:
       XML Mapper code is required to use this macro. You can use SAS XML Mapper tool
       to create one. The XML Map file should be available in the Directory that is
       provided for PATH= parameter

Parameters:
       The macro uses keyword parameters. Multiple words need to be separated by a
       Single Blank

       WORDS=  (Mention All of the words to search. This is an AND condition)
       SINCE=  (Enter From Date in the format: YYYY-MM-DD)
       UNTIL=  (Enter To Date in the format: YYYY-MM-DD)
       PHRASE= (Enter Exact Phrase you want to search)
       ANY=    (Enter Any of the words you want to search)
       NONE=   (Enter the words you do not want to be in search results)
       HASH=    (Enter the hash tag that you want in your results)
       FROM=    (Enter name of the person who is Tweeting. Only one at a time)
       QUESTION=Enter 1 if you want only the tweets with a Question Mark)
       CODE=    (Enter base64 encoded string of Twitter login)
       PATH=    (Directory where fetched data sets will be saved)
 *************************************************************************************/

/*Define macro variable to capture Base64 encoded string */
%let authorization=%nrstr("Authorization:Basic NoZbGFAZa29tOnNhdHRpQDc1OA==");
%let path=%nrstr(C:SGF11 Paper\GetTweet); /*Enter your Directory path*/

/*Macro Definition*/
%macro gettweet(WORDS=,PHRASE=,ANY=,NONE=,HASH=,FROM=,TO=,SINCE=,
                 UNTIL=,QUESTION=,CODE=,PATH=);

libname Twit "&PATH";
%let dataset=Tweets; /*Give a name for Destination Data set*/

filename httpOut "&PATH\twitterOutput.xml";
filename hOut "&PATH\httpOutputHeaders.txt";
filename hIn temp;
filename httpreq temp;

/*Add "+" between keyword parameters when Multiple words are used in the search*/

%let WORDS=%sysfunc(translate(%sysfunc(strip(&WORDS)),"+"," "));
%let PHRASE=%sysfunc(translate(%sysfunc(strip(&PHRASE)),"+"," "));
%let ANY=%sysfunc(translate(%sysfunc(strip(&ANY)),"+"," "));
%let NONE=%sysfunc(translate(%sysfunc(strip(&NONE)),"+"," "));
%let HASH=%sysfunc(translate(%sysfunc(strip(&HASH)),"+"," "));
%let FROM=%sysfunc(translate(%sysfunc(strip(&FROM)),"+"," "));
%let TO=%sysfunc(translate(%sysfunc(strip(&TO)),"+"," "));

%if &QUESTION=1
%then
       %let QUESTION=%nrstr(&tude[]=%3F);
%else
       %let QUESTION=%nrstr(&tude[]=);
```

```
/*Define a macro variable "search" to create the search string from parameters*/
%let search=%nrstr(q=&ands=)&WORDS%nrstr(&phrase=)&PHRASE%nrstr(&ors=)
&ANY%nrstr(&nots=)&NONE%nrstr(&tag=)&HASH%nrstr(&lang=en)%nrstr(&from=)&FROM%nrstr(&to
=)&TO%nrstr(&since=)&SINCE%nrstr(&until=)&UNTIL&QUESTION;

/*Create a Temp file used in PROC HTTP headerin option to hold base64 encode*/
data _null_;
       file hIn;
       put &code;
run;

/*Create Destination Data set*/
proc sql;
create table TWIT.&dataset
(
  id char(39) format=$39. informat=$39.,
  published num format=IS8601DT19. informat=IS8601DT19.,
  title char(159) format=$159. informat=$159.,
  updated num format=IS8601DT19. informat=IS8601DT19.,
  twitter_source char(100) format=$100. informat=$100.,
  twitter_lang char(2) format=$2. informat=$2.,
  uri char(50) format=$50. informat=$50.,
  content char(2600) format=$2600. informat=$2600.
); quit;

/*Initialize and increment Page Number*/
%let pageno=1;
%StartLoop:

/*Define a variable for the number of tweets per page. The maximum allowed is 100 */
%let pagerate=%nrstr(&rpp=100&page=);
/*Combine Search String, Page rate and Page Number Macro variables*/
%let searchstring="&search&pagerate&pageno";

/*Create a Temp file used in PROC HTTP IN= option */
data _null_;
       file httpreq;
       put &searchstring;
run;

proc http
       in=httpReq
       out=httpOut
       headerin=hIn
       headerout=hOut
       url="http://search.twitter.com/search.atom"
       method="get"
       ct="application/x-www-form-urlencoded";
run;

/*Define XML Mapper and XML Library*/
filename SXMLMAP "&PATH\TwitterSearch.map";
filename XMLLib "&PATH\twitterOutput.xml";
libname XMLLib xml xmlmap=SXMLMAP ACCESS=readonly;

/*Concatenate the XML Results in 'Entry' data set and destination data sets*/
data twit.&dataset;
       set twit.&dataset XMLLib.entry;
run;


/*Query the count of Tweets returned, into the Macro Variable 'obscount'*/
proc sql noprint;
       select count(*) into :obscount from XMLLib.entry;
quit;
```

```
%let pageno=%eval(&pageno+1); /*Increment Page Number*/

/**The Loop terminates if it is Page Fifteen or If it is the Last Page (<15) and has
less than 100 tweets. We can fetch a maximum of 1500 tweets at a time. If the tweets
available are less than 1500 the loop is terminated else the tweets from the last
fetched page keep writing to the Data set**/
%if %eval(&pageno)=16 or %eval(&obscount)<100
%then %goto EndLoop;
%else %goto StartLoop;

%EndLoop:

/*Summarize Tweets*/

proc sort data=twit.&dataset out=&dataset._temp nodupkey;
by title uri; run;    /*Delete duplicates*/

/*Below DATA step cleans tweets and creates two data sets (one with all the tweets and
the other only with retweets) in the work library*/
data &dataset (KEEP= id pubdate text author source retweet)
       &dataset._rt (KEEP= id pubdate title text author source tweet_owner);
       set &dataset._temp;
length text $ 159 tweet_owner $ 20 source $ 20;
format pubdate date7.;
retweet=0;
text=title;
author=tranwrd(uri,'http://twitter.com/','@');
id=substr(id,29);
pubdate=datepart(published);

if substr(text,1,3)='RT ' then do;
       retweet=1;
       tweet_owner=tranwrd(scan(text,2),':','');
       call scan(text, 3, position, length);
       text=substr(text,position);
end;

/*Define PERL regular expressions to replace http/www/RT text from tweets*/
if _n_=1 then do;
       retain pattern pattern2;
       pattern = PRXPARSE ("s/(RT @.[^ ]*)|((http|www)(\d|\D)[^ ]*)|(@.[^ ]*)//i");
       pattern2 = PRXPARSE ('/"(\w|\W)[^ ]*"/');
end;
call prxchange(pattern,-1,text);
text=strip(text);
if prxmatch(pattern2, twitter_source) then do;
       call prxposn(pattern2, 0, position, length);
       source = strip(substr(twitter_source, position+8, length-9));
end;
if retweet=1 then output &dataset._rt;
output &dataset;
run;

/*Calcualte Total Number of records collected from twitter */
proc sql noprint;
       select count(*) into :twtcount from twit.&dataset;
quit;
proc sql noprint;
       select count(*) into :retwtcount from &dataset._rt;
quit;

/*Terminate macro execution if usernames are specified in FROM= parameter.
No Tweet report is generated. Only Data sets are created */
%if %length(&FROM) ne 0 %then %goto EndMacro;
```

```
/*Create Data sets for generating Tweet Report*/
proc sql outobs=10;
create table toptweeters as
select author 'Tweeter',count(author) 'Tweets'
from &dataset
group by author
order by 2 desc;
quit;

proc sql outobs=10;
create table topsources as
select source 'Source',count(source) 'Count'
from &dataset._rt
group by source
order by 2 desc;
quit;

proc sql outobs=10;
create table topowners as
select tweet_owner 'Tweeter',count(tweet_owner) 'Count'
from &dataset._rt
group by tweet owner
order by 2 desc;
quit;

/*Define ODS Layout and generate PDF Report*/
options orientation=landscape;
goptions reset=all dev=sasprtc ftext="Helvetica";
ods listing close;
ods pdf file="&PATH\TweetReport_&dataset..pdf" STARTPAGE=NO BOOKMARKGEN=NO;
ods layout start;
ods region x=0 in y=0 in height=8.5 in width=11 in;
proc gslide;
title1 h=17pt j=Center underlin=1 'Tweet Report'  lspace=.1in;
title2 h=12pt j=Left "            Tweets:&twtcount" "       Retweets:&retwtcount"
lspace=.1in;
run;quit;

goptions border;
ods region x=0.25 in y=0.3 in height=3.5 in width=5 in;
title1;
axis1 label=(angle=90 "Tweeter")  minor=none;
axis2 label=(height=15pt "Top 10 Tweeters") minor=none;
proc gchart data=toptweeters;
hbar author/ sumvar= _TEMA001 maxis=axis1 raxis=axis2;
run;quit;

axis1 label=(angle=90 "Source")  minor=none;
axis2 label=(height=15pt "Top 10 Sources") minor=none;
ods region x=5.5 in y=0.3 in height=3.5 in width=5 in;
proc gchart data=topsources;
hbar source/ sumvar= _TEMA001 maxis=axis1 raxis=axis2;
run;quit;

axis1 label=(angle=90 "Tweet Owner")  minor=none;
axis2 label=(height=15pt "Top 10 Influencers") minor=none;
ods region x=0.25 in y=4.75 in height=3.5 in width=5 in;
proc gchart data=topowners;
hbar tweet_owner/ sumvar= _TEMA001 maxis=axis1 raxis=axis2;
run;quit;


axis1 label=(height=15pt "Tweets  per day")  minor=none ;
axis2 label=( angle=90 "Tweets") minor=none major=(n=4);
ods region x=5.5 in y=4.75 in height=3.5 in width=5 in;
proc gchart data=&dataset;
```

```
pattern1 color=red value=r3;
vbar pubdate /discrete outside=freq maxis=axis1 raxis=axis2 ;
run;quit;

ods layout end;
ods pdf STARTPAGE=NOW;
ods layout start;

ods region x=0 in y=0 in height=8.5 in width=11 in;
proc gslide;
title1 h=17pt j=left '            Tweet Report-Top Retweets' lspace=.25in;
run;quit;

ods region x=0 in y=0.3 in height=7.5 in width=10 in; title1;
proc sql outobs=20;
select Text 'Tweet',count(*) 'Retweets'
from &dataset._rt
group by Text
order by 2 desc;
quit;

ods layout end;
ods pdf close;
goptions reset=all;
ods listing;

%EndMacro:

%put Collected Tweets:&twtcount, Collected Retweets:&retwtcount;

%mend gettweet;


%gettweet(HASH=DMA2010, CODE=&authorization, PATH=&path);
```