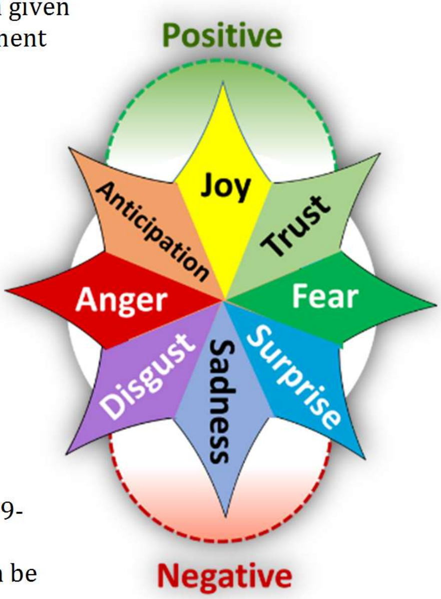
Sentiment Analysis

Sentiment analysis aims to measure the attitude of a writer's words within a given text. It is an application of Text Mining and frequently used to extract sentiment from social media sources including survey results, Twitter tweets and Facebook comments.



This analytic uses R's tidytext[1] package to analyze text for sentiments by associating each word with classifications from two lexicons:

- NRC Word Emotion Association Lexicon[2]: A list of over 14,000 English words and their associations with
  - Eight basic emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust), and
  - Two sentiments (negative and positive).

  The annotations were manually done by crowdsourcing.
- AFINN Lexicon[3]: A list of over 2,400 English words, each word is scored for sentiment with an integer between
  - -5 (extremely negative) and +5 (extremely positive).

  The words have been manually labeled by Finn Årup Nielsen in 2009-2011.

This analytic returns the sentiment analyses as 14 different results, each can be represented by a MicroStrategy metric.

In addition to these results which are returned "in-band" to MicroStrategy metrics in a report, dashboard or document, this analytic optionally can persist output "out-of-band" to the file system, including a result table as a comma-separated-value file, a word cloud and a sentiment score histogram.

For more information, see the Outputs section below.

How It Works:

While Sentiment Analysis is considered an advanced analytic, the approach taken here is deceptively simple. Deployed as a MicroStrategy Metric, this analytic has a single input, a vector of text elements. Each text element is processed following these steps:

1. Breaking, or tokenizing[4], each text element into the words it contains.
2. Removing stop words[5] such as "the", "of", "a" and other common parts of speech that irrelevant when it comes to analyzing text for sentiment.
3. Depending on the source of the text, perform special handling of words embedding technology-related features such as URLs, Hashtags (begin with #) and Users (begin with @).
4. The remaining words are joined to the words in each lexicon, capturing the sentiments and emotions they tend to be associated with.
5. The result for each text element is determined by summing the sentiments and emotions associated with the words it contains.

[1] See https://cran.r-project.org/web/packages/tidytext/vignettes/tidytext.html
[2] See http://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm
[3] See http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=6010
[4] See https://en.wikipedia.org/wiki/Lexical_analysis#Tokenization
[5] See https://en.wikipedia.org/wiki/Stop_words

Example of Sentiment Analysis in Action:

To illustrate the process, let’s perform Sentiment Analysis of the Gettysburg Address delivered by President Abraham Lincoln in November 1863, at the dedication of the National Cemetery in Gettysburg, Pennsylvania. It is iconic for its brevity as well as its use of language to frame the struggles of the Civil War and proclaim the principals of human equality. Lincoln delivered the ten sentences of his address in just over two minutes.

Table 1 below represents the internal data structure of the Sentiment Analysis. For each sentence, the words are matched against each lexicon and their sentiments (words matching those in the lexicons are highlighted in **red-bold** text). Those word-by-word results are summed for each text element (sentence).

ID	Text	Word	Anger	Anticipation	Disgust	Fear	Joy	Negative	Positive	Sadness	Surprise	Trust	Score
1	Four **score** and seven years ago our fathers brought forth on this continent, a new **nation**, conceived in **Liberty,** and dedicated to the **proposition** that all men are created equal.	**score**	0	1	0	0	1	0	1	0	1	0	0
		nation	0	0	0	0	0	0	0	0	0	1	0
		liberty	0	1	0	0	1	0	1	0	1	1	0
		proposition	0	0	0	0	0	0	1	0	0	0	0
		TOTAL	**0**	**2**	**0**	**0**	**2**	**0**	**3**	**0**	**2**	**2**	**0**
2	Now we are **engaged** in a great **civil war,** testing whether that **nation**, or any **nation** so conceived and so dedicated, can long **endure.**	**engaged**	0	1	0	0	1	0	1	0	0	1	0
		civil	0	0	0	0	0	0	1	0	0	0	0
		war	0	0	0	1	0	1	0	0	0	0	-2
		nation	0	0	0	0	0	0	0	0	0	1	0
		nation	0	0	0	0	0	0	0	0	0	1	0
		endure	0	0	0	0	0	0	1	0	0	0	0
		TOTAL	**0**	**1**	**0**	**1**	**1**	**1**	**3**	**0**	**0**	**3**	**-2**
3	We are met on a great **battle**-field of that **war**.	**battle**	1	0	0	0	0	1	0	0	0	0	-1
		war	0	0	0	1	0	1	0	0	0	0	-2
		TOTAL	**1**	**0**	**0**	**1**	**0**	**2**	**0**	**0**	**0**	**0**	**-3**
4	We have come to dedicate a portion of that field, as a final resting place for those who here gave their lives that that **nation** might live.	**nation**	0	0	0	0	0	0	0	0	0	1	0
		TOTAL	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**
5	It is altogether **fitting** and **proper** that we should do this.	**fitting**	0	1	0	0	1	0	1	0	0	1	0
		proper	0	0	0	0	0	0	1	0	0	0	0
		TOTAL	**0**	**1**	**0**	**0**	**1**	**0**	**2**	**0**	**0**	**1**	**0**
6	But, in a **larger sense,** we can not dedicate-we can not consecrate-we can not hallow-this **ground.**	**larger**	0	0	1	0	0	0	0	0	1	1	0
		sense	0	0	0	0	0	0	1	0	0	0	0
		ground	0	0	0	0	0	0	0	0	0	1	0
		TOTAL	0	0	1	0	0	0	1	0	1	2	0
7	The brave men, living and dead, who struggled here, have consecrated it, far above our poor power to add or **detract**.	**detract**	1	0	0	0	0	1	0	0	0	0	0
		TOTAL	**1**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**
8	The world will little note, nor long remember what we say here, but it can never **forget** what they did here.	**forget**	0	0	0	0	0	1	0	0	0	0	-1
		TOTAL	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**-1**
9	It is for us the living, rather, to be dedicated here to the **unfinished** work which they who fought here have thus far so nobly **advanced**.	**unfinished**	0	0	0	0	0	1	0	0	0	0	0
		advanced	0	0	0	0	0	0	1	0	0	0	1
		TOTAL	**0**	**0**	**0**	**0**	**0**	**1**	**1**	**0**	**0**	**0**	**1**
10	It is rather for us to be here dedicated to the great **task** remaining before us-that from these honored dead we take increased devotion to that cause for which they gave the last full **measure** of devotion-that we here highly resolve that these dead shall not have died in vain-that this nation, under **God**, shall have a new **birth** of **freedom**-and that **government** of the people, by the people, for the people, shall not **perish** from the earth	**task**	0	0	0	0	0	0	1	0	0	0	0
		measure	0	0	0	0	0	0	0	0	0	1	0
		nation	0	0	0	0	0	0	0	0	0	1	0
		god	0	1	0	1	1	0	1	0	0	1	1
		birth	0	1	0	1	1	0	1	0	0	1	0
		freedom	0	0	0	0	1	0	1	0	0	1	2
		government	0	0	0	1	0	1	0	0	0	0	0
		perish	0	0	0	1	0	1	0	1	0	0	0
		TOTAL	**0**	**2**	**0**	**4**	**3**	**2**	**4**	**1**	**0**	**5**	**3**

Table 1: Sentiment Analysis of Lincoln's Gettysburg Address (internal dataset representation during analytic execution)

Table 1 represents the internal dataset for Sentiment Analysis with a sentence-word per row to illustrate how the calculation is performed.

Table 2 shows how results returned from the Sentiment Analysis analytic are typically deployed, where each row has a text element to be analyzed. While the text elements in this example are sentences from a speech, they could easily be responses from a survey, comments from FaceBook post, or tweet messages from Twitter. The metrics on columns contain 13 sentiment analysis results, all available from this analytic:

- Eight basic emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust), and
- Two sentiments (negative and positive).
- A numeric Positive/Negative Score on a scale from -5 (extremely negative) and +5 (extremely positive).
- An English text Grade corresponding to the numeric Positive/Negative Score.
- A word count that tallies how many words were attempted to be analyzed for sentiments.
- (a 14th result is not shown, the TotalWordCount aggregation of the WordCount summed from each text element)

ID	Text	Anger	Anticipation	Disgust	Fear	Joy	Negative	Positive	Sadness	Surprise	Trust	Score	Grade	WordCount
1	Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and dedicated to the proposition that all men are created equal.	0	2	0	0	2	0	3	0	2	2	0	Neutral	12
2	Now we are engaged in a great civil war, testing whether that nation, or any nation so conceived and so dedicated, can long endure.	0	1	0	1	1	1	3	0	0	3	-2	Somewhat Negative	9
3	We are met on a great battle-field of that war.	1	0	0	1	0	2	0	0	0	0	-3	Negative	4
4	We have come to dedicate a portion of that field, as a final resting place for those who here gave their lives that that nation might live.	0	0	0	0	0	0	0	0	0	1	0	Neutral	8
5	It is altogether fitting and proper that we should do this.	0	1	0	0	1	0	2	0	0	1	0	Neutral	3
6	But, in a larger sense, we can not dedicate-we can not consecrate-we can not hallow-this ground.	0	0	1	0	0	0	1	0	1	2	0	Neutral	6
7	The brave men, living and dead, who struggled here, have consecrated it, far above our poor power to add or detract.	1	0	0	0	0	1	0	0	0	0	0	Neutral	9
8	The world will little note, nor long remember what we say here, but it can never forget what they did here.	0	0	0	0	0	1	0	0	0	0	-1	Slightly Negative	4
9	It is for us the living, rather, to be dedicated here to the unfinished work which they who fought here have thus far so nobly advanced.	0	0	0	0	0	1	1	0	0	0	1	Slightly Positive	6
10	It is rather for us to be here dedicated to the great task remaining before us-that from these honored dead we take increased devotion to that cause for which they gave the last full measure of devotion-that we here highly resolve that these dead shall not have died in vain-that this nation, under God, shall have a new birth of freedom-and that government of the people, by the people, for the people, shall not perish from the earth.	0	2	0	4	3	2	4	1	0	5	3	Positive	24

Table 2: Sentence-by-Sentence Sentiment Analysis of Lincoln's Gettysburg Address (results returned from this analytic)

This sentiment analysis shows how Lincoln's comments begin neutral, turn negative when describing the battlefield and the impacts of the Civil War, but conclude positively at the end thanks to the cathartic tone delivered by Lincoln's words.

How to Deploy to MicroStrategy:

Prerequisite: Please follow the instructions in the R Integration Pack User Guide[6] for configuring your MicroStrategy environment with R and that the R Script functions have been installed in your MicroStrategy project(s).

1) Save the script to the RScripts folder where the R Integration Pack is installed:
    a. Windows: Intelligence Server and Developer Client:
        - Typically, C:\Program Files (x86)\R Integration Pack\RScripts
    b. Windows: Desktop and Workstation Clients:
        - Typically, C:\Program Files\R Integration Pack\RScripts
    c. Linux: Intelligence Server:
        - Typically, /opt/mstr/MicroStrategy/install/IntelligenceServer/RIntegrationPack/RScripts
    d. Mac: Desktop and Workstation Clients:
        - Typically, ______________________________.
2) From the R console, run the SentimentAnalysis.R script to verify the script runs correctly. For details, see the "**Running from the R Console**" section below.
3) Cut-and-paste the metric expressions below in any metric editor. These metrics return the sentiment analysis results.
4) Use the new metric in reports, dashboards and documents.

Metric Expressions:

1) **Score**: Returns the numeric Positive/Negative sentiment score:

```
RScriptU<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="Score",
_Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE, RemoveRetweets=TRUE,
SaveCSV=TRUE">(Text)
```

2) **Grade**: Returns the English string corresponding to the Positive/Negative sentiment score:

```
RScriptU<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="Grade",
_Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE, RemoveRetweets=TRUE,
SaveCSV=TRUE">(Text)
```

3) **Anger**: Returns the count of words associated with the sentiment Anger:

```
RScriptU<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="Anger",
_Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE, RemoveRetweets=TRUE,
SaveCSV=TRUE">(Text)
```

4) **Anticipation**: Returns the count of words associated with the emotion Anticipation:

```
RScriptU<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="Anticipation",
_Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE, RemoveRetweets=TRUE,
SaveCSV=TRUE">(Text)
```

5) **Disgust**: Returns the count of words associated with the emotion Disgust:

```
RScriptU<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="Disgust",
_Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE, RemoveRetweets=TRUE,
SaveCSV=TRUE">(Text)
```

6) **Fear**: Returns the count of words associated with the emotion Fear:

[6] Use this URL to download the R Integration Pack User Guide, https://github.com/MicroStrategy/RIntegrationPack/blob/master/docs/Documentation_RIntegrationPackUserGuide.pdf

```
RScriptU<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="Fear",
_Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE, RemoveRetweets=TRUE,
SaveCSV=TRUE">(Text)
```

7) **Joy**: Returns the count of words associated with the emotion Joy:
```
RScriptU<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="Joy",
_Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE, RemoveRetweets=TRUE,
SaveCSV=TRUE">(Text)
```

8) **Negative**: Returns the count of words associated with the sentiment Negative:
```
RScriptU<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="Negative",
_Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE, RemoveRetweets=TRUE,
SaveCSV=TRUE">(Text)
```

9) **Positive**: Returns the count of words associated with the sentiment Positive:
```
RScriptU<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="Positive",
_Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE, RemoveRetweets=TRUE,
SaveCSV=TRUE">(Text)
```

10) **Sadness**: Returns the count of words associated with the emotion Sadness:
```
RScriptU<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="Sadness",
_Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE, RemoveRetweets=TRUE,
SaveCSV=TRUE">(Text)
```

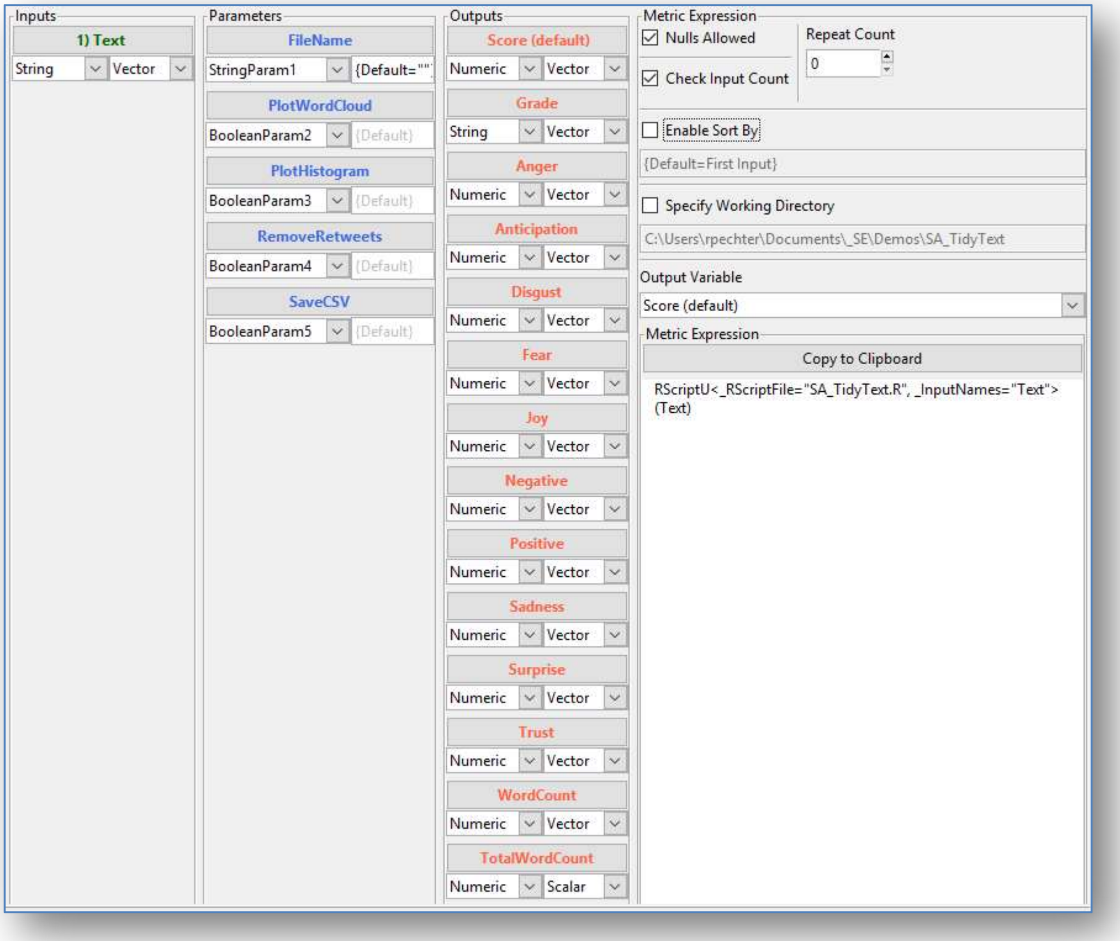
11) **Surprise**: Returns the count of words associated with the emotion Surprise:
```
RScriptU<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="Surprise",
_Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE, RemoveRetweets=TRUE,
SaveCSV=TRUE">(Text)
```

12) **Trust**: Returns the count of words associated with the emotion Trust:
```
RScriptU<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="Trust",
_Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE, RemoveRetweets=TRUE,
SaveCSV=TRUE">(Text)
```

13) **WordCount**: Returns the count of words all words considered for sentiment analysis (i.e., after removal of stop words and any other text items like URLs, hashtags and user handles):
```
RScript<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="WordCount",
_Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE, RemoveRetweets=TRUE,
SaveCSV=TRUE">(Text)
```

14) **TotalWordCount**: An aggregation across all of the text elements of the counts of words considered for sentiment analysis:
```
RScriptAgg<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="TotalWordCount",
_Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE, RemoveRetweets=TRUE,
SaveCSV=TRUE">(Text) RScript<_RScriptFile="SentimentAnalysis.R", _InputNames="Text",
StringParam1="MyWordCloud" >(Text)
```

Analytic Signature:



Inputs

1) Text: String, Vector

Parameters

FileName: StringParam1, {Default=""

PlotWordCloud: BooleanParam2, {Default}

PlotHistogram: BooleanParam3, {Default}

RemoveRetweets: BooleanParam4, {Default}

SaveCSV: BooleanParam5, {Default}

Outputs

Score (default): Numeric, Vector

Grade: String, Vector

Anger: Numeric, Vector

Anticipation: Numeric, Vector

Disgust: Numeric, Vector

Fear: Numeric, Vector

Joy: Numeric, Vector

Negative: Numeric, Vector

Positive: Numeric, Vector

Sadness: Numeric, Vector

Surprise: Numeric, Vector

Trust: Numeric, Vector

WordCount: Numeric, Vector

TotalWordCount: Numeric, Scalar

Metric Expression

☑ Nulls Allowed

☑ Check Input Count

Repeat Count: 0

☐ Enable Sort By

{Default=First Input}

☐ Specify Working Directory

C:\Users\rpechter\Documents_SE\Demos\SA_TidyText

Output Variable: Score (default)

Metric Expression

Copy to Clipboard

```
RScriptU<_RScriptFile="SA_TidyText.R", _InputNames="Text">
(Text)
```

Input:

Text: A vector of text (ex. Survey results, Facebook comments, Twitter Messages, etc.). The metric will analyze sentiment for every text element.

> NOTE: If **Text** comes from a MicroStrategy attribute, it must be converted to a metric for input to this analytic, since all MicroStrategy functions take metrics as inputs (metrics have the required aggregation function and level which attributes alone lack).
>
> To convert the attribute **Text** to a metric, simply wrap the attribute in an aggregation function at the report level:
>
> ```
> Max(Text){~}
> ```
>
> To avoid creating a metric, simply use this metric expression as the input to the Sentiment Analysis metric expressions above. For example:
>
> ```
> RScriptU<[_RScriptFile]="SentimentAnalysis.R", [_OutputVar]="Score",
> _Params="FileName='SA_mstr', PlotWordCloud=TRUE, PlotHistogram=TRUE,
> RemoveRetweets=TRUE, SaveCSV=TRUE">(Max(Text){~})
> ```

Parameters:

FileName: A string parameter containing the name to be used for any files persisted to the file system during execution of the R script. Customize this parameter for each application so that results from one application do not overwrite the results from different applications using the same R analytic. If this parameter is missing from the metric expression, the default is an empty string.

PlotWordCloud: A Boolean parameter that, when set to TRUE, uses R's **wordcloud** package to generate a WordCloud plot of all the words in all the elements passed into the analytic. This plot will be saved to the file system using the **FileName** parameter as FileName_WordCloud.jpg. If this parameter is missing from the metric expression, the default is FALSE and no plot is generated.

PlotHistogram: A Boolean parameter that, when set to TRUE, uses R's **hist** function to generate a histogram of the **Score** output for all elements passed into the analytic. This plot will be saved to the file system using the **FileName** parameter as FileName_Histsogram.jpg. If this parameter is missing from the metric expression, the default is FALSE and no plot is generated.

RemoveRetweets: A Boolean parameter that, when set to TRUE, will eliminate any text elements that appear to be Twitter retweets. This feature can be used when performing sentiment analysis of a Twitter account to exclude any sentiments from retweets (since those sentiments are from different accounts). The sentiment results for these elements will all be 0/Neutral. A text element is considered a retweet if either of these criteria are met:

1) The text begins with a double quote
2) The text begins with "RT", "retweet", "from", or "via" followed by a twitter handle and colon ("@....:")

If this parameter is missing from the metric expression, the default is FALSE and retweets will not be eliminated.

SaveCSV: A Boolean parameter that, when set to TRUE, saves a table of results as a comma-separated-value file using the **FileName** parameter as FileName.csv. If this parameter is missing from the metric expression, the default is FALSE and no file is generated.

Outputs Returned ("In-Band"):

1) **Score**: Returns the numeric Positive/Negative sentiment score.
2) **Grade**: Returns the English string corresponding to the Positive/Negative sentiment score.
3) **Anger**: Returns the count of words associated with the sentiment Anger.
4) **Anticipation**: Returns the count of words associated with the emotion Anticipation.
5) **Disgust**: Returns the count of words associated with the emotion Disgust.
6) **Fear**: Returns the count of words associated with the emotion Fear.
7) **Joy**: Returns the count of words associated with the emotion Joy.
8) **Negative**: Returns the count of words associated with the sentiment Negative.
9) **Positive**: Returns the count of words associated with the sentiment Positive.
10) **Sadness**: Returns the count of words associated with the emotion Sadness.
11) **Surprise**: Returns the count of words associated with the emotion Surprise.
12) **Trust**: Returns the count of words associated with the emotion Trust.
13) **WordCount**: Returns the count of words all words considered for sentiment analysis (i.e., after removal of stop words and any other text items like URLs, hashtags and user handles).
14) **TotalWordCount**: An aggregation across all of the text elements of the counts of words considered for sentiment analysis.

Additional Results Generated by the R Script ("Out-of-Band"):

WordCloud: This optional feature is controlled by the **PlotWordCloud** parameter. If TRUE, a plot is generated of all the words in all the elements passed into the analytic and saved to the file system using the **FileName** parameter as FileName_WordCloud.jpg.

Histogram: This optional feature is controlled by the **PlotWordCloud** parameter. If TRUE, a histogram is generated of the **Score** output for all elements passed into the analytic and saved to the file system using the **FileName** parameter as FileName_Histsogram.jpg.

Result Table: This optional feature is controlled by the **SaveCSV** parameter. If TRUE, a table of results is saved as a comma-separated-value file using the **FileName** parameter as FileName.csv.

.Rdata File: This file contains the complete R environment at the conclusion of the R script execution.

- If the R Script completed without error, this file will be saved to the file system using the **FileName** parameter as FileName_FINAL.Rdata.
- If the R Script execution encountered an error, this file will be saved to the file system using the **FileName** parameter as FileName_ERROR.Rdata.

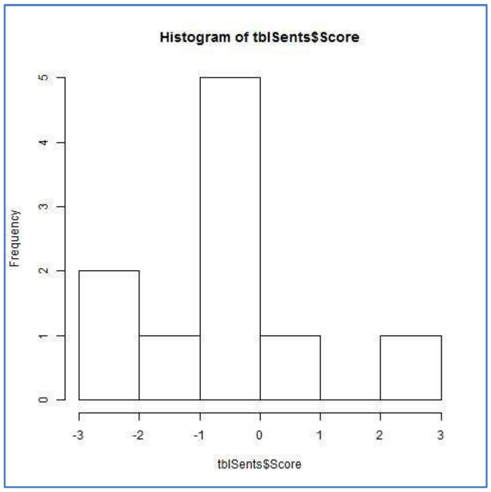
Running from the R Console:

In addition to processing data from MicroStrategy during execution of a report or dashboard, the R script is also configured to run from the R console. Running the script for the R Console verifies that the script is functioning as expected, a good practice when initially deploying this analytic to a new system (for more details, see "Configuring dual execution modes" in the R Integration Pack User Guide).

When run from the R Console, the R Script performs sentiment analysis on the Gettysburg Address (see example above) and proper execution will generate the results in Table 2 above and these two plots:

dead
field
war
living
conceived
dedicated
people
devotion
dedicate
nation

Gettysburg Address Word Cloud



Gettysburg Address Sentiment Histogram

Troubleshooting:

This section covers certain situations you might encounter but it's not intended as a comprehensive list of possible errors.

1) If an error occurs, the report may fail with an error message, or nulls returned as the output. In these cases, please refer to the RScriptErrors.log file generated for further guidance and the DSSErrors.log. Please consult the User Guide [1] and the R documentation for additional guidance.
2) The script will attempt to install the required R packages when executed on systems with ability to access the R-Project's CRAN repositories on the Internet. If the packages are not successfully installed, you can install using the R console using the command:

```
install.pacakges(c("tidytext", "wordcloud", "dplyr", "stringr", "tidyr"),
repos="http://cran.rstudio.com/")
```