Real-Time Sentimental Polarity Classification on Live Social-Media

Dr. Khalid N. Alhayyan
Assistant Professor and Faculty Member in IT Sector
Institute of Public Administration, P.O. Box 205, Riyadh 11141, Saudi Arabia
alhayyan@ipa.edu.sa

and

Dr. Imran Ahmad
Data Scientist, Immigration, Refugees and Citizenship Canada / Government of Canada
180 Kent Street, Ottawa
imran.ahmad@cic.gc.ca

ABSTRACT

In recent years, the popularity of social media networks has attracted the attention of researchers, government agencies, politicians and business world alike, as a powerful platform to explore real-time trends. The data generated by these networks offers an opportunity to investigate people's behaviors and activities, but the high velocity and low quality of this data poses some unique challenges. Twitter, an example of social media networks, is particularly popular for this purpose due to its easily accessible API that is open to use for research purposes. Different techniques can be used to analyze patterns from available data. One of such techniques for extracting subjective information from any text such as opinions on various topics is Sentiment Polarity Classification, which quantifies emotions embedded in texts and classifies them as positive, negative or neutral. The focus of this paper is on preparing and analyzing real-time twitter streams to detect real-time trends on a particular topic using Sentiment Polarity Classification. We have used StreamSensing approach and have performed a supervised machine learning on real-time high velocity data using Apache Spark micro-batching technology to classify the opinions and feelings of people in real-time. Appropriate experiments for processing high rate of incoming streams have been carefully designed and conducted on live twitter data. The outcomes of these experiments were analyzed and presented. The findings of this paper fell into two perspectives: theoretical and practical. The theoretical perspective is seen in testing and confirming the validity of StreamSensing approach as well as the introduction of a sentimental polarity algorithm, while practically; this approach can be employed to perform trend analyses on any real-time streams related to live events.

Keywords: StreamSensing, Real-time Trends, Sentiment Analysis, Supervised Method, Pattern Analysis, Polarity Classification.

1. INTRODUCTION

Social media play important roles in today's world. They are influential in a variety of social phenomena, including economic exchanges, political processes, and sport events (Le et al., 2015). Real-time social media, such as Facebook, Twitter, and Instagram have become a significant source of valuable data. This data can be used for analyses based on which important decisions can be made. With the rapid growth of engagement in social media, analytics becomes attractive to various fields such as marketing, sociology, and information systems for many reasons. Among which is discovering emerging patterns in real-time (Alhayyan and Ahmad, 2017), predicting the performance of financial markets (Bollen et al., 2011; Mittal et al., 2012), identifying relevant events (Becker et al., 2011), building content-based recommender systems (Chen et al., 2010), detecting emerging security threats (Fire et al., 2014), and improving decision making and business intelligence (Farzindar 2012).

Users of social media can write blogs and reviews, post messages on discussion platforms, and publish their opinions in a moment. This phenomenon leads to a continuous flow of a huge amount of data, containing traces of valuable information, such as people's sentiment with respect to products, brands, and events. As estimates indicate, there are around 2.6 million blog posts written per day (WorldPress, Nov 2017), and approximately 600 million tweets per day (Twitter Inc., 2017). The abundance of user-generated content published through such social media renders automated information monitoring tools crucial for today's businesses. Sentiment analysis (SA), also sometimes called opinion mining, comes to answer this need. The access to real-time data on social networks is an opportunity for researchers to extract and study various patterns out of users' opinions. Twitter is an excellent example of social networks to observe these opinions for its short and shared messages. An aspect of this observation is to employ sentiment analysis to extract, classify, and aggregate these opinions. SA has been one of the most active research areas in natural language processing since early 2000. It refers to a broad area of natural language processing, computational linguistics, and text mining (Hogenboom et al., 2013). SA attempts to understand the sentimental polarity of the web comments by classifying comments into positive, negative, and neutral categories (Cai et al., 2008). When SA is employed for real-time analytics in Twitter, number of challenges need to be considered. First, it is impossible to store instances of data, and therefore high-speed analytical algorithms should be utilized. Second, computing resources, such as CPU and memory, are expected to be highly consumed. Therefore, pre-processing of data should be performed in a way that only a short summary of stream is stored in main memory. Third, due to high speed of arrival, average processing time for each instance of data should be in such a way that incoming instances are not lost without being captured. Fourth, the learner needs to provide high analytical accuracy measures (Hossein et al., 2016). Beside these challenges, inherently tweets usually have low quality data, with fixed length text 240-characters, written in a casual language. In addition, in many cases the message may contain text like usernames, links, repeated letters, and emoticons that are irrelevant to sentimental analysis. Last, for training a supervised model, tweeter data is needed with known sentiment polarity. The quality of the trained model depends on the accurate polarity of the training dataset.

The model, is in fact, formalized patterns found in the training dataset.

In this paper, we follow and track the spread of real-time opinions on Twitter, use a fast in-memory processing system, called Apache Spark, and perform a supervised learning approach via implementing a staged methodology appropriate for analyzing and discovering real-time noisy streams, called StreamSensing (Alhayyan and Ahmad, 2017). The results of such implementation are dynamic and dependent on the real-time views of social media at that instant of time. Such analysis is bound to be a snapshot that captures the summary of views at a particular pre-defined window of time. The possible applications of such a system are numerous, among of which is the example of monitoring opinions in relation to election candidates during a televised debate, and then tracking the changes in opinions in real-time. Moreover, it would be interesting for some to observe the impacts of companies' announcements or news events on traders' behaviors during trading hours. Additionally, it can be used to gauge the live conversations relative to sport events, such as soccer games, summarizing the sentimental of the large crowd of fan during the game (Le et al., 2015).

This paper is presented into five sections. The introduction is in section 1, while the second section presents the literature review. The employed methodology is in section 3, and the experimental results and model evaluation is in section 4. The paper is concluded in section 5.

2. LITERATURE REVIEWS

Our efforts on conducting the literature reviews fall into two main areas. First, we targeted the area of analyzing real-time streams in Twitter. Second, we consider the specific area of employing real-time SA in Twitter.

Twitter streams are different from other streams in number of ways. First, Twitter messages are restricted in length (240 characters) and written by anyone, while most media messages are well written, structured, and edited. Therefore, tweets may include large amounts of informal, irregular, and abbreviated words, large number of spelling and grammatical errors, and improper sentence structures and mixed languages. Also Twitter streams contain large amounts of meaningless messages (Hurlock and Wilson 2011), polluted content (Lee et al. 2011), and rumors (Castillo et al. 2011), which negatively affect the performance of the detection algorithms (Atefeh et al., 2013). Prior researches have proposed various techniques for Twitter stream analysis and discovery. Depending on the discovery method, the presented techniques can be categorized into supervised and unsupervised (or a combination of both) approaches. In a supervised approach, the aim is to perform a mapping from the input to an output whose correct values are provided by a trained supervisor model, while in an unsupervised approach, there is no such supervisor and we only have an input data, within which patterns occurring more often than others are extracted, with no prior knowledge involved. Some techniques for noisy Twitter streams rely on clustering approaches, which are naturally suitable for because they are most likely unsupervised in that they require no labeled data for training. However, these clustering approaches must be efficient and highly scalable, and they should not require any prior knowledge such as the number of clusters (Atefeh et al., 2013). On the other hand, supervised models, if simple, reduce biases and variances. In addition, they are proven to be fast and more accurate. While there are many distinct techniques on stream analysis and discovery, we present four of them as they are more related to our work. For example, Alhayyan and Ahmad (2017) introduced an approach, called StreamSensing, to perform trend analysis on any real-time stream data. StreamSensing consists of six stages: tokenization, stop words removal, stemming, filtering, conversion into Term Document Matrix (TDM), and finally pattern analysis. Maynard et al. (2017) present a tested framework for collecting and analyzing real-time social media contents. This framework consists of four main steps: data collection, queueing, processing, and presenting results. Becker et al. (2011a) used RW-Event classifier to identify real-world event contents on Twitter. Cordeiro (2012) combined wavelet analysis and topic inference summarization to detect events that are happening at a given time. Long et al. (2011) employed a hierarchical divisive clustering approach to divide topical words into event clusters.

We now consider the specific area of using real-time SA in Twitter. Four studies were targeted for their highly relevance. First, through dynamic graphical representations updated in real-time, Azzouza et al. (2017) implemented a learning technique to analyze opinions and detect tweets polarity, by which relevant keywords regarding the main topic of interest can be recommended. In conjunction with the Auto-Regressive Sentiment-Aware (ARSA) model, Liu et al. (2010) applied an adaptive S-PLSA+ model, which is capable of incrementally updating its parameters and automatically down-dating old information when new review data become available, to predict sales performance. Quanzeng You (2016) proposed joint visual-textual sentiment analysis model, and compared the performance of neural networks versus classifiers that use predefined low-level or mid-level features attributes. While the results show that the proposed model has significantly improved the performance of sentiment analysis on several datasets, it is not clear how to deal with the challenge of fast incoming data streams. Le et al. (2015) developed a machine-learning system, based on natural language processing and opinion mining. This system leverages social media streams to automatically identify and predict the outcomes of soccer matches. While their system outcomes were deemed promising, their method of data collection was based on storing data first, and then conducting the analysis in a different time. These studies and others are viewed to be limited in their applied methodologies for only performing SA in an off-line manner on a sample of stored stream data. While these methods can work well in some cases, they may not be applicable in real-time fashions. Additionally, real-time SA tools such as MOA, and RapidMiner exist, however they are uniprocessor solutions and they cannot be scaled for an efficient usage in a network nor a cluster. As a result, processing time per instance of data becomes higher and instances get lost in a stream. This affects the learning curve and accuracy measures due to less available data for training and can introduce high costs to such solutions. For this reason, our approach employs Apache Spark, which is capable of processing high rate of incoming streams, to overcome these challenges

Drawing upon the previous research efforts and synthesizing the different approaches and techniques employed on discovering and analyzing real-time SA, we experimentally test the approach of Streamsensing (Alhayyan and Ahmad, 2017) and perform a supervised machine learning on real-time high velocity data using Apache Spark, and the results of such implementation are reported.

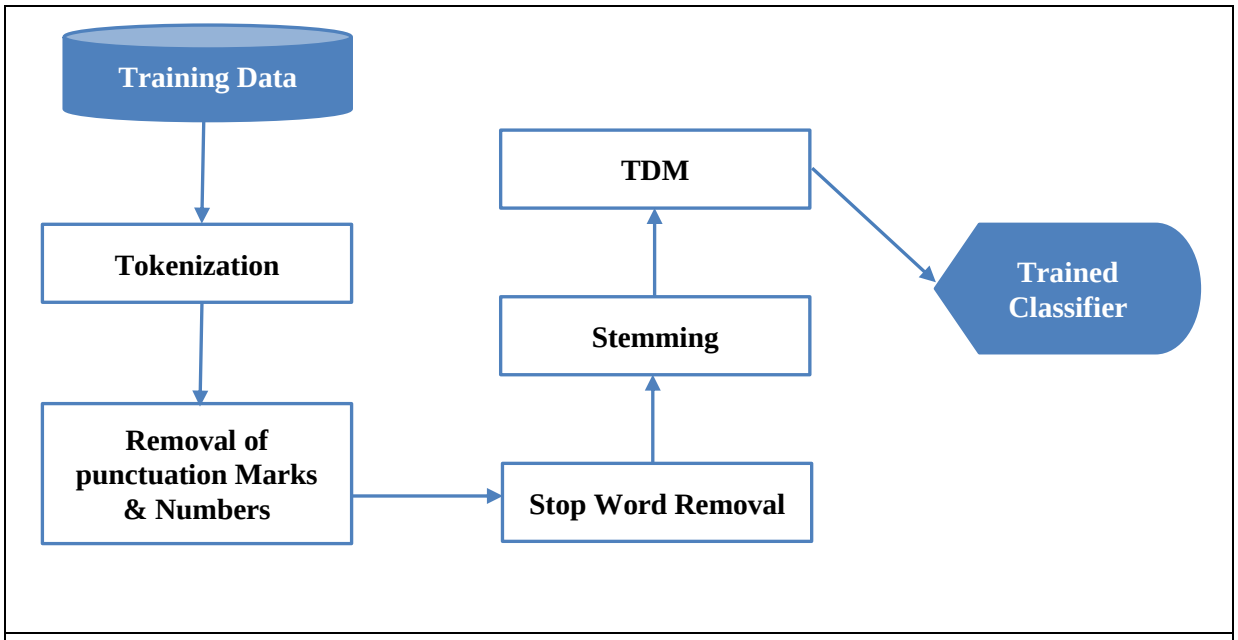

Figure 1: Adapting StreamSensing (Alhayyan and Ahmad, 2017) for training classifier

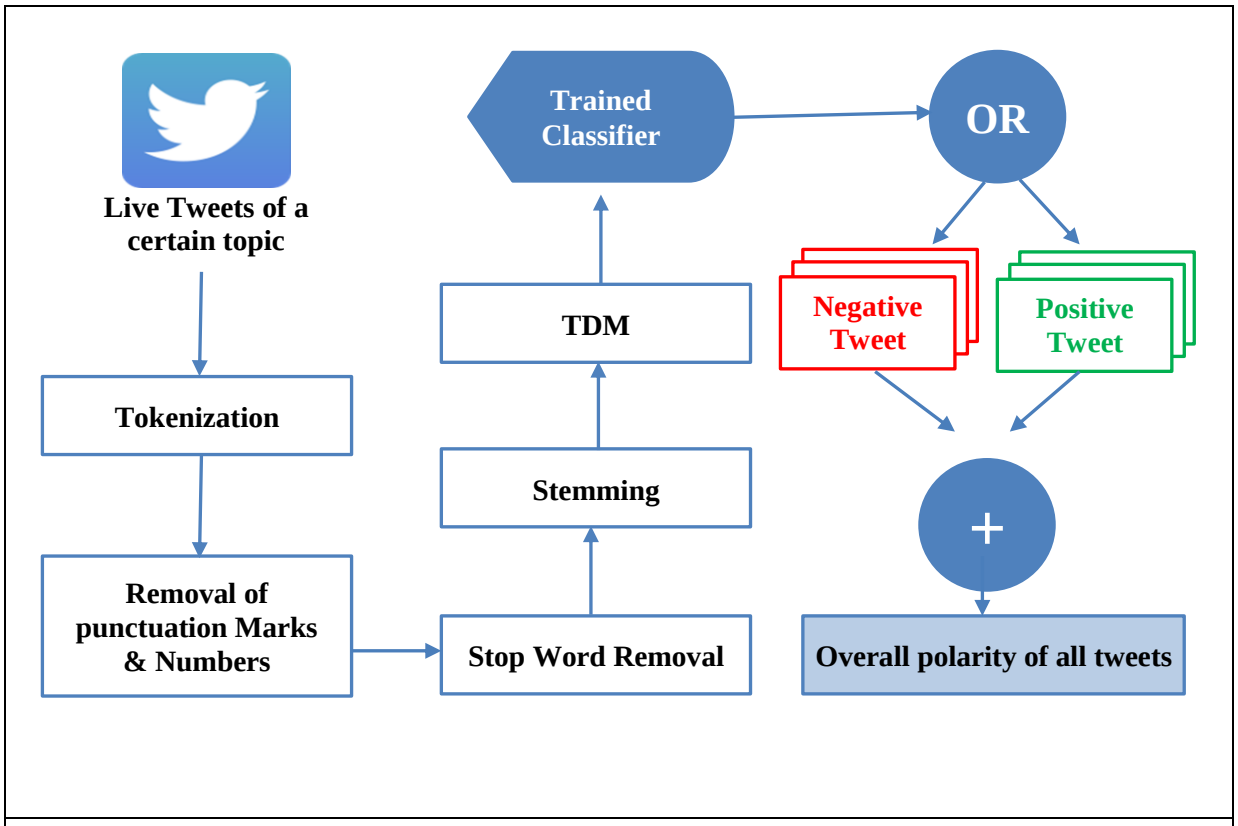

Figure 2: Real-time sentimental polarity classification on Twitter

**** Overall Polarity of all tweets Module ****

```
void findOverallPolarity (String [ ] AllTweets)
{
   int overAllScore = 0;
   String overAllPolarity = "";
   for (int i = 0; I < AllTweets.length; i++) {
      overAllScore += findTweetPolarity(AllTweets[i]);
    }
    if (overAllScore > 0 )
        overAllPolarity = "positive";
    else if (overAllScore < 0 )
        overAllPolarity = "negative";
    else
        overAllPolarity = "neutral";
}
```

**** Single Tweet Polarity Module ****

```
int findTweetPolarity (String tweet)
{
    String [ ] splitTweet = StreamSensing(tweet);
    int tweetScore = 0;
    for (int i = 0; I < splitTweet.length; i++) {
        if (splitTweet[i] is in positiveList)
            tweetScore ++ ;
        else if (splitTweet[i] is in negativeList)
            tweetScore -- ;
    }
    return tweetScore;
}
```

Figure 3: Sentimental polarity algorithm

3. METHODOLOGY

The approach for conducting this research was designed based on classifying high rate of incoming stream tweets. The approach is to extract the embedded sentiments within tweets about a chosen topic. The sentiment classification quantifies the polarity in each tweet in real-time, and then aggregate the total sentiments from all tweets to capture the overall sentiments about the chosen topic.

As shown in Figure 1, the first step is to train the classifier. As deemed to be the leading methodology used by industry for data mining, the CRISP-DM (Cross-industry standard process for data mining) is considered for training the classifier. This process consists of six man phases: *business understanding, data understanding, data preparation, modeling, evaluation, and deployment.*

1- The *business understanding* is driven by the problem statement that specifies the need for sentimental polarity of a stream of tweets related to a chosen topic.

2- For *data understanding*, the velocity and the quality of tweets related to the chosen topics were analyzed to make sure that the right processing techniques has been selected that can prepare the data for the machine learning technique in real-time.

3- The *data preparation* step involved in this phase are shown in Figure 1. The tweets are first split into individual words called tokens (tokenization). The output from tokenization creates a bag-of-words, which is a collection of individual words in the text. These tweets are further filtered by removing numbers, punctuations, and stop words (Stop Word Removal). Stop words are words that are extremely common like "is", "am", "are", and "the". These words, as they hold no additional information, are thus removed. Additionally, non-alphabetical characters, symbols such as "#@" and numbers, are removed using pattern matching, as they hold no relevance in the case of sentiment analysis. Regular expressions are used to match alphabetical characters only and the rest are ignored. This helps to reduce the clutter from the twitter stream. The outcomes of the prior phase is taken to the phase of stemming. In this phase, the derived words are reduced to their roots. Example includes words like "fish" which has same roots as "fishing" and "fishes". We used the library of Stanford NLP, which provides various algorithms such as porter stemming. Once the data is processed, it is converted into a structure called Term Document Matrix (TDM). TDM represents the term and frequency of each work in the filtered corpus. The output of the *data preparation* stage is TDM and that is used as the input to the machine-learning algorithm. StreamSensing is used in this stage to provide the necessary capacity to prepare real-time tweets.

4- Forth stage is the *Modeling or training a classifier*. A trained classifier needs to be plugged with the generated TDM to determine the polarity of each of the tweets (positive, negative, or neutral), followed by the aggregation and determination of the overall polarity of all tweets about a certain topic (see Figure 2). The classifier need to be trained with labelled data with known values of the target variable. To ensure the capability of processing high rate of incoming streams, Apache Spark is used as the compute engine that provides the processing power necessary for classifying incoming tweets in real-time.

In order to train the classifier, we needed an already-prepared dataset that has historical tweeter data and follows the patterns and trends of the real-time data. Therefore, we used the dataset from the website (www.sentiment140.com), which comes with a human-labeled corpus (a large collection of texts upon which analysis is based) with over 1.6 million tweets. The tweets within this dataset has been labelled with one of three polarities; 0 for negative, 2 for neutral, and 4 for positive. In addition to the tweet text, the corpus provides the tweet id, date, flag, and user who tweeted. The process of training the classifier starts with plugging the chosen dataset with the StreamSensing approach. From TDM, we calculated the Sentimental Polarity Importance (SPI) of each word based on its occurrence patterns. SPI is a number that ranges

from -5 to 5. The positive or negative sign specifies the type of emotions represented by that particular word, and its magnitude represents the strength of sentiment.

Once the classifier is trained, it becomes ready to process live tweets about a chosen topic (see Figure 2). To retrieve the real-time raw tweets, we used the Scala library "Twitter4j", a Java library that provides a package for real-time twitter streaming API. The API requires the user to register a developer account with Twitter, and fill in some authentication parameters. This API allows either getting all random tweets, or filtering tweets using chosen keywords. We used filters to retrieve tweets related to our chosen topic. Each tweet needs to pass through all StreamSensing phases, which converts the tweets into a TDM. The trained classifier then classifies each of the tweet into one of three classes: positive, negative, or neutral based on its sentimental polarity score. While classifying tweets, an overall score is maintained, based on which the overall polarity of all tweets about a certain topic is determined.

To discover the polarity, we used an algorithm for counting positive and negative words in each tweet. For the two classes (positive and negative words), two different lists were made by the trained classifier. Every word in a tweet is compared against the two lists. If the current word matches a word in the positive list, a score of 1 is incremented, while if a negative word is found then it is decremented. More words that are positive lead to higher sentiment score, while more words that are negative lead to lower sentiment score. The overall polarity score is aggregated based on the polarities of all tweets. Figure 3 shows the employed sentimental polarity algorithm.

5- The final stage is the *model evaluation and deployment*. Once the model is trained, it is evaluated and then used in production. The results of model evaluation are discussed in the next section.

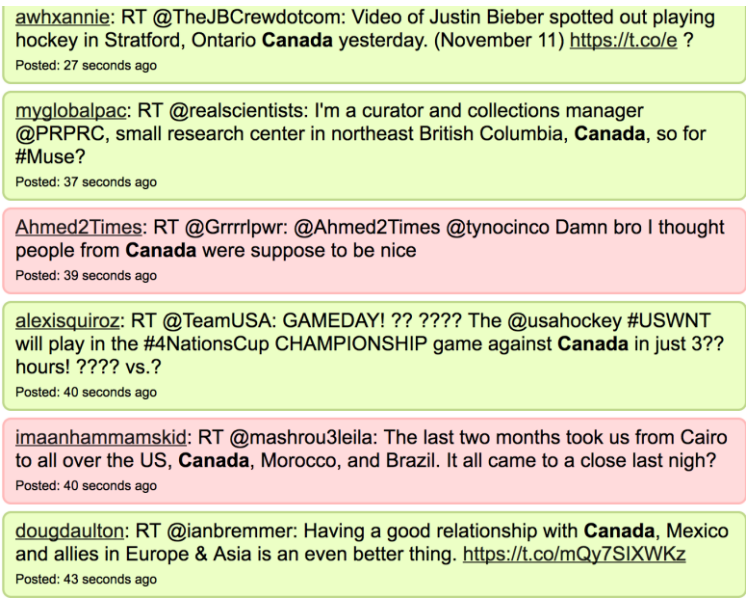

Figure 4: Color-coded (green and red) real-time tweets about the keyword "Canada"

4. EXPERIMENTAL RESULTS AND MODEL EVALUATION

For evaluating the model, three keywords were chosen: "Canada", "iPhoneX", and "United Airlines". Three experiments were conducted on November 12, 2017, with one experiment for each chosen keyword. Table 1 shows some basic information about these experiments.

Figure 4 shows a sample of the real-time tweets about the keyword "Canada". In this figure, each of the individual tweets are color-coded, according to the sentiment they carry. The red color indicates tweets that have negative polarity (the third and fifth tweets in Figure 4), while the green color identifies tweets with positive polarity (the first, second, forth, and sixth tweets in Figure 4). The tweets relevant to each of these keywords were collected for a window of 5 minutes and the resultant TDM was fed to the trained classifier. Each experiment was repeated 6 times for each of the three keywords and the results are shown in Figure 5. Table 2 shows the sentimental polarity classification of the three experiments corresponding to the three chosen keywords. Each tweet is tokenized, processed and converted into a TDM. The TDM is then fed to the trained classifier. The classifier determines the polarity of each of the tweets by calculating the polarity of the individual words.

Table 1: Some statistics about the three experiments

	Keyword	Number of tweets	Duration
1	Canada	350	30 mins.
2	iPhoneX	240	30 mins.
3	United Airlines	100	30 mins.

Table 2: Sentimental polarity classification of the three experiments

	Keyword	Sentiment by %		Sentiment by count	
		Positive	Negative	Positive	Negative
1	Canada	71%	29%	250	100
2	iPhoneX	83%	17%	200	40
3	United Airlines	20%	80%	20	80

Figure 5 shows aggregated results of each of the three stream gathered for 30 minutes, the product of 5 minutes times 6 repetitions. The histogram representation of sentiment polarity shows the average number of tweets in 30 minutes. It can be observed from Table 2 and Figure 5 that sentimental polarity of keyword "Canada" is overall positive (71% vs 29%), where 250 tweets were classified as positive tweets, while 100 tweets were classified as negative tweets, making the overall score of the sentimental polarity equals to +150. This positive number results through implementing the sentimental polarity algorithm, shown in Figure 3, in this order: (+250 – 100 = +150). For the keyword "iPhoneX" the sentiment polarity is evaluated by the model as positive in overall (83% vs 17%). For keyword "United Airlines", the sentimental polarity is mostly negative (80% vs 20%). These real-time sentiment dashboards may be used to summarize the public opinions of people about a certain topic in certain instance of time.

5. CONCLUSION

In this paper, we followed and tracked the spread of real-time opinions on Twitter, used a fast in-memory processing system, called Apache Spark, and performed a supervised learning approach via implementing a staged methodology appropriate for analyzing and discovering real-time noisy streams, called StreamSensing (Alhayyan and Ahmad, 2017). The results of such implementation were dynamic and time-related. The findings of this paper fall into two perspectives: theoretical and practical. The theoretical perspective is viewed in testing and confirming the validity of StreamSensing approach as well as the introduction of sentimental polarity algorithm. Practically, this approach can be extended and used for performing trend analyses on any real-time streams related to live events, such as monitoring live opinions about election candidates, observing live thoughts about stocks' announcements or news events during trading hours, and exploring live conversations during sport events.

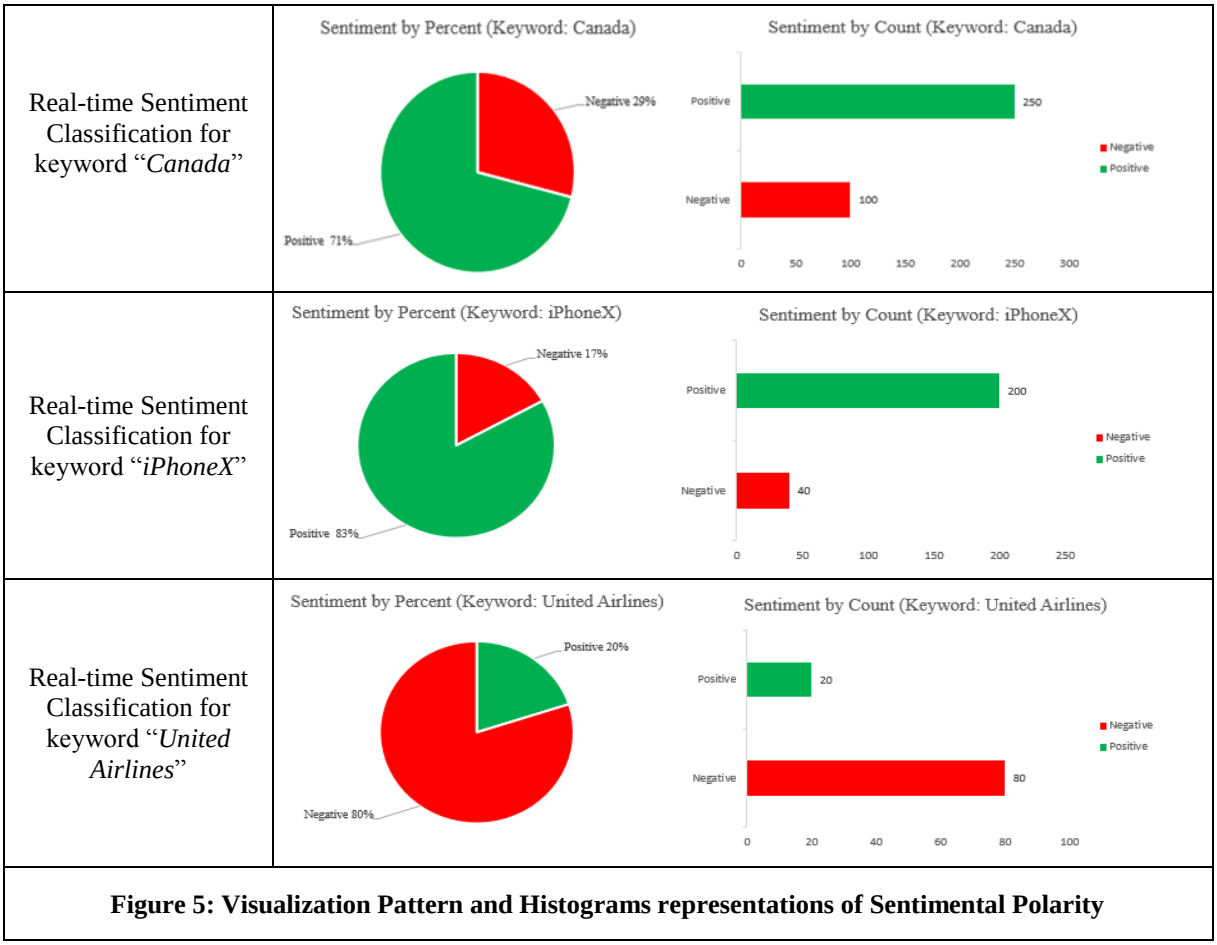

Figure 5: Visualization Pattern and Histograms representations of Sentimental Polarity

6. REFERENCES

[1] A. Farzindar, (2012). Industrial perspectives on social networks. In EACL 2012 - Workshop on Semantic Analysis in Social Media.

[2] A. Go, R. Bhayani., L. Huang,. (2009). Twitter Sentiment Classification using Distant Supervision. In (http://www.stanford.edu/~alecmgo/papers /TwitterDistantSupervision09.pdf).

[3] A. Hogenboom, D. Bal, and F. Frasincar. Exploiting Emoticons in Sentiment Analysis. ACM, 978-1-4503-1656-9/13/03, 2013.

[4] A. Hossein, and A. Rahnama, (2016). Distributed Real-Time Sentiment Analysis for Big Data Social Streams. IEEE International Conference on Control, Decision and Information Technologies. arXiv:1612.08543.

[5] A. Mittal, and A. Goel, (2012). Stock Prediction Using Twitter Sentiment Analysis. Working Paper Stanford University CS 229.

[6] B. Jansen, M. Zhang, K. Sobel, and A. Chowdury. Twitter Power: Tweets as Electronic Word of Mouth. Journal of the American Society for Information Science and Technology, 60(11):2169{2188, 2009.

[7] C. Castillo, M. Mendoza, and B. Poblete. (2011). Information credibility on Twitter. In Proceedings of the 20th International Conference on World Wide Web, WWW '11, ACM, New York, NY, pp. 675–684.

[8] C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, and D. McClosky. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 55-60.

[9] D. Maynard, I. Roberts, M. Greenwood, D. Rout, and K. Bontcheva, (2017). A Framework for Real-time Semantic Social Media Analysis. Web Semantics: Science, Services and Agents on the World Wide Web, Volume 44, May 2017, Pages 75-88.

[10] F. Atefeh, and W. Khreich, (2013). A Survey of Techniques for Event Detection in Twitter. In Journal Computational Intelligence, Volume 31 Issue 1, Pages 132-164.

[11] H. Becker, M. Naaman, and L. Gravano, (2011). Beyond Trending Topics: Real-World Event Identification on Twitter, In Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media. Pages 438-441.

[12] H. Becker, M. Naaman, and L. Gravano. (2011a). Beyond trending topics: Real-world event identification on Twitter. In ICWSM, Barcelona, Spain.

[13] J. Bollen, H. Mao, and X. Zeng, (2010). Twitter mood as a stock market predictor. IEEE Computer, 44(10). Pages 91–94.

[14] J. Chen, R. Nairan, L. Nelson, M. Bernstein, and E. Chi, (2010). Short and tweet: experiments on recommending content from information streams. In CHI '10 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. Pages 1185-1194.

[15] J. Hurlock, and M. Wilson. (2011). Searching Twitter: separating the tweet from the chaff. In International AAAI Conference on Weblogs and Social Media, Barcelona, Spain.

[16] K. Alhayyan, and I. Ahmad. Discovering and Analyzing Important Real-Time Trends in Noisy Twitter Streams. Journal of Systemics, Cybernetics and Informatics (JSCI), Vol. 2 – No. 2 – 2017, pp. 25-31, ISSN: 1690-4524 (Online).

[17] K. Cai, S. Spangler, Y. Chen, and L. Zhang. Leveraging Sentiment Analysis for Topic Detection. IEEE Computer Society, 978-0-7695-3496-1, 2008.

[18] K. Lee, B. Eoff, and J. Caverlee. (2011). Seven months with the devils: A long-term study of content polluters on Twitter. In International AAAI Conference on Weblogs and Social Media, Barcelona, Spain.

[19] L. Le, E. Ferrara, and and A. Flammini. On Predictability of Rare Events Leveraging Social Media: A Machine Learning Perspective. In Proceedings of the 2015 ACM on Conference on Online Social Networks (pp. 3-13). ACM. 2015.

[20] M. Cordeiro, (2012). Twitter event detection: Combining wavelet analysis and topic inference summarization. In Doctoral Symposium on Informatics Engineering, DSIE'2012.

[21] M. Fire, M. Goldschmidt, and Y. Elovici, (2014). Online Social Networks: Threats and Solutions. IEEE Communications Surveys & Tutorials (Volume: 16, Issue: 4, Fourthquarter 2014).

[22] M. Taboada, K. Voll, and J. Brooke. Extracting Sentiment as a Function of Discourse Structure and Topicality. Technical Report 20, Simon Fraser University, 2008. Available online, http://www.cs.sfu.ca/research/ publications/techreports/#2008.

[23] N. Azzouza, K. Akli-Astouati, A. Oussalah, and S. Ait Bachir. (2017). A Real-time Twitter Sentiment Analysis using an unsupervised method, In Proceedings of WIMS '17, Amantea, Italy, June 19-22, 2017, 10 pages. DOI: 10.1145/3102254.3102282.

[24] O. Phelan, K. McCarthy, and B. Smyth. (2009). Using Twitter to recommend real-time topical news. In Proceedings of the 2009 ACM Conference on Recommender Systems, 2009, pp. 385–388.

[25] P. Melville, V. Sindhwani, and R. Lawrence. Social Media Analytics: Channeling the Power of the Blogosphere for Marketing Insight. In 1st Workshop on Information in Networks (WIN 2009), 2009.

[26] Q. You. (2016). Sentiment and Emotion Analysis for Social Multimedia: Methodologies and Applications. In Proceedings MM '16 Proceedings of the 2016 ACM on Multimedia Conference. Pages 1445-1449.

[27] R. Long, H. Wang, Y. Chen, O. Jin, and Y. Yu. (2011). Towards effective event detection, tracking and summarization on microblog data. InWeb-Age Information Management, Vol. 6897 of Lecture Notes in Computer Science. Edited by WANG, H., S. LI, S. OYAMA, X. HU, and T. QIAN. Springer: Berlin/Heidelberg, pp. 652–663.

[28] Y. Liu, X. Yu, X. Huang, and A. An. (2017). S-PLASA+: adaptive sentiment analysis with application to sales performance prediction, In Proceeding SIGIR '10 Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval, Pages 873-874.