International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:9, No:1, 2015

What the Future Holds for Social Media Data Analysis

P. Wlodarczak, J. Soar, M. Ally

***Abstract*—**The dramatic rise in the use of Social Media (SM) platforms such as Facebook and Twitter provide access to an unprecedented amount of user data. Users may post reviews on products and services they bought, write about their interests, share ideas or give their opinions and views on political issues. There is a growing interest in the analysis of SM data from organisations for detecting new trends, obtaining user opinions on their products and services or finding out about their online reputations. A recent research trend in SM analysis is making predictions based on sentiment analysis of SM. Often indicators of historic SM data are represented as time series and correlated with a variety of real world phenomena like the outcome of elections, the development of financial indicators, box office revenue and disease outbreaks. This paper examines the current state of research in the area of SM mining and predictive analysis and gives an overview of the analysis methods using opinion mining and machine learning techniques.

***Keywords*—**Social Media, text mining, knowledge discovery, predictive analysis, machine learning.

I. Introduction

OPINIONS are central to almost all human activities because they are key influences of our behaviours [1]. Whenever we have to make a decision such as buying a car or selecting a holiday destination we often want to know other people's opinions. In the past when an organization needed public opinions on their products or services it conducted surveys, consumer focus groups or other information gathering approaches. With the advent of Web 2.0 technologies people have been sharing opinions and views on the Internet at an unprecedented rate. Social Media has become a premium site for sharing opinions, ideas and views. There has been unprecedented interest in Social Media as a category of online discourse where people create content, share, bookmark and network at a prodigious rate [2].

With the massive growth in the utilization of SM such as Facebook or Twitter individuals and organizations are increasingly using the content for decision making. Positive reviews can make an impact on the propensity of customers to buy products or influence the choices for holiday destinations [15]. A presumption is that behavior is an indication of future decisions and consequently, SM is seen to have the potential to predict future behavior. Organizations have realized the potential of SM data mining and are increasingly using it in their decision making processes [16].

Not surprisingly a new area of research has emerged called *predictive analytics*. Predictive analytics (PA) refers to "Technology that learns from experience (data) to predict the future behaviour of individuals in order to drive better decisions" [3].

Predictive analysis research using SM has been used in various domains. The increase in the use of social media has led many social scientists to examine whether extracting specific patterns in SM might be able to predict real-world outcomes [17]. An increasingly popular way of extracting useful information from social network platforms is to build indicators, often in the form of a time series, of general public mood by means of sentiment analysis [4]. The time series are then compared against real-world data to find correlations. If a correlation could be detected and the same pattern occurs in the future, it might suggest the same real-world phenomenon will occur. These methods have been used to predict elections [5], financial indicators [6], box office revenue [2], disease outbreaks [7], and natural disasters [8].

This paper describes the research methods used for SM mining and predictive analysis applied in recent studies. SM is analyzed using *text mining* to automatically extract actionable patterns from SM content. *Machine learning* (ML) systems can learn from data to make better decisions in the future. There are other text mining techniques that do not use ML but employ statistical methods or lexicon-based sentiment analysis [9], [13]. This paper focuses on ML using Twitter data. ML systems are trained using historic SM data. Once the system has been trained, it can be used to find correlations and determine if real-world predictions can be made.

II. Research Methodology

Predictive analysis of SM data comprises two phases, a data conditioning phase and a predictive analysis phase. In the conditioning phase the data is collected and preprocessed for analysis. In the analysis phase the data is mined for actionable patterns and correlations are searched for.

A. Data Collection

SM sites such as Twitter or Facebook provide an *Application Programming Interface* (API) through which data can be accessed programmatically. Facebook has its own query language, Facebook Query Language (FQL), and Twitter provides a query API for accessing historic data, and a streaming API for real-time data access. The "firehose" API gives access to 100%, the "gardenhose" API to 10% and the "spritzer" API to 1% of real-time data. Gardenhose and spritzer access is free whereas firehose access comes at an expensive cost. Twitter has changed the conditions for external access to its data several times in the past and might

P. Wlodarczak is a research student at the Faculty of Business, Education, Law and Arts, University of South Queensland, Australia, currently doing a PhD in the area of SM and predictive analysis (e-mail: wlodarczak@gmail.com).

J. Soar is a professor at the faculty of Business and Law, University of South Queensland, Australia.

M. Ally is a lecturer in Information Systems in the School of Management and Enterprise at the University of Southern Queensland, Australia (e-mail: allym@usq.edu.au).

change it in the future.

Tweets can be accessed using the Java programming language and the twitter4j library. There are other libraries such as Spring Social and other languages such as Python or Rubi that can be used.

Twitter has a rate limit of 180 requests per time window of 15 minutes in version 1.1 of its API, but that might change over time. To obtain sufficient material, the queries have to run over a certain amount of time. A query for "Apple Inc." using Java and twitter4j is shown in Fig. 1.

```
try {
       Query query = new Query("Apple Inc");
           QueryResult result;
           result = twitter.search(query);
           List<Status> tweets = result.getTweets();
           for (Status tweet : tweets) {
               System.out.println("@" +
tweet.getUser().getScreenName() + " - " +
tweet.getText());
           }
```

Fig. 1 Java code to query Tweets

The collected data can be stored on the local file system or in a database for pre-processing.

B. Data Pre-Processing

SM data is "noisy" and contains *spam* messages. It has to be purified and passed through a *relevance filter*. Irrelevant data such as spam Tweets have to be discarded and content such as smileys or special characters such as "@" have to be removed. Smileys have sometimes been used to determine the sentiment polarity [4].

Tweets which match the regular expressions "http:" and "www." are filtered out as possible spam. Duplicates, retweets and non-English Tweets are discarded. Due to the brevity of Tweets with a maximum of 140 characters automatic language detection tools such as the Guess Language library might fail occasionally.

The Tweets need to be cleaned from *stop words* such as "the" or "and" and punctuations. There is no definitive list of stop words. In some cases words such as "isn't" are removed, however "isn't" is a sentiment polarity shifter and can change the opinion to the opposite.

To get accurate results, usually only Tweets with explicit mood statements are considered, that is statements such as "I love Google". There are dictionaries of words annotated with their semantic orientation, which are the polarity and the strength [9]. They can be used to select the words that will be considered.

In some studies relevance-filtering methods such as *Latent Dirichlet Allocation* (LDA) [4] have been applied. LDA is based on Latent Semantic Indexing. The LDA algorithm is trained with relevant Tweets and generates a latent description. Test Tweets are then passed through the trained LDA filter.

C. Data Classification

The data is *classified* using textual sentiment classifiers. It has to be determined whether a Tweet contains positive or negative sentiments towards a given subject, person or idea. For this purpose the *semantic orientation* has to be determined. Semantic orientation (SO) is a measure of subjectivity and opinion in text [9]. Multiclass sentiment analysis divides Tweets in several mood states such as Happy, Unhappy, Playful, Sceptical whereas binary sentiment classifiers divide the Tweets into two groups, e. g. positive and negative.

Machine learning (ML) techniques are often used for classification. Spam detection is one of the most prevalent applications of ML. Emails are classified in legitimate and spam mails [10]. There are *supervised*, *unsupervised* and *semi-supervised* ML methods. For classification supervised machine learning methods are applied. Supervised methods are used when the class label is known. Unsupervised learning is used for data without class labels, and semi-supervised learning algorithms are used when small amounts of labeled and large amounts of unlabeled data exist.

For supervised learning algorithms, a given data set is typically divided into two parts: training and testing data sets with known class labels [10]. The classifier is fed with the training data. The training goes through several iterations until the *classification accuracy* converges. After every iteration the result is corrected using human judgement.

D. Classifiers

We want to obtain a decision function f, that classifies Tweets t as positive (P), or negative (N). If we denote the set of all Tweets by T, we search for a function f:T $\rightarrow$ {N,P}. We use a set of randomly selected and pre-classified training Tweets $\{(t_1, c_1), (t_2, c_2), \ldots, (t_n, c_n)\}$, where: $t_i \in$ T, $c_i \in$ {P, N}.

Typical supervised learning methods include *naïve Bayes classification*, *decision tree induction*, *k-nearest neighbors*, and *support vector machines* [11]. There are many more ML algorithms. Experience shows that no single machine learning scheme is appropriate to all data mining problems [12]. Usually several algorithms are trained and compared to determine which one gives the most accurate results for a given problem.

The naïve Bayes classifier is a popular algorithm in text categorisation. It is a family of simple probabilistic classifiers based on the Bayes theorem. Decision tree learning, as the name suggests, uses decision trees for data mining. k-Nearest Neighbour or k-NN is a non-parametric method that takes the k closest training examples as input and classifies by a majority vote of its neighbours. Support Vector Machine (SVM) classification is based on statistical learning theory and tries to find a *linear separation boundary* for classifying the training data.

Artificial Neural Networks (ANN) are a large group of algorithms. ANN can be used for both classification and for finding correlations. They consist of *perceptrons*, the neurons, interconnected through weighted *connections*, the axon. The idea of a perceptron is to find a linear function

$$f(x) = w^T x + b \quad (1)$$

such that $f(x) > 0$ for one class and $f(x) < 0$ for the other class. $w = (w_1, w_2, \ldots, w_m)$ is the vector of coefficients (weights) of the function, and b is the bias. During training the weights and bias are adjusted. An ANN can consist of many perceptrons, organised in layers. They are called *multilayer perceptron* and can be visualised as network of layers of neurons interconnected through axons.

Most ML algorithms cannot handle text, only numerical objects, real numbers or vectors. The Tweets have to be converted into *feature vectors*, for instance a vector with the numbers of occurrences of certain words. Defining the feature extractor is a crucial step. If it is chosen so that there might exist a positive and a negative Tweet with the same feature vector, no matter how good the machine learning algorithm is, it will make mistakes. It should be noted that the features in the vector need not all be extracted from the message itself, we may actually add information if beneficial.

From the feature vectors a randomised sample is selected which is used for training. Sometimes a second validation data set is generated for optimisation of the learning algorithm or to predict the error.

There is no general way of defining the sample size. To build an accurate classifier following the rules of thumb applies:

- enough training examples
- good performance on training set
- classifier that is not too "complex" ("Occam's razor")

To measure the purity p of the function, commonly the entropy or the Gini index are used. p is the fraction of positive examples. The entropy is calculated as

$$-p \ln p - (1-p) \ln(1-p) \quad (2)$$

The Gini index as

$$p(1-p) \quad (3)$$

The algorithms compute error rates that are used to select the best performing algorithm. The *training error* is the fraction of training examples misclassified, the *test error* is the fraction of test examples misclassified, and the *generalization error* is the probability of misclassifying new random example. The training error is also used to determine the best tree size in case of a decision tree.

There are various approaches to determine the most appropriate algorithm for the given problem. The simplest approach is counting the proportion of the correctly predicted samples of a test set. This value is the *accuracy*, also called the *1-ErrorRate*.

A more sophisticated method is cross-validation. The test data set is randomly reordered and then split into a number of *n folds* of equal size. In each iteration *n-1* folds are used for training the classifier and one fold is used for testing. The test results are collected and averaged over all folds to determine the *cross-validation estimate* of the accuracy [12].

The output of the training depends on the algorithm. In the case of a decision tree the rules for the decisions will be created, and in the case of a neural network the network, the perceptrons and the weighted connections will be created. Once the training has completed, the test data is applied against the trained algorithm. The algorithm that has the highest *classification accuracy* is selected.

E. Time Series

The classified Tweets will be represented as *time series*, which is the number of positive or negative Tweets per time unit in the case of a binary sentiment classifier. The time window can be hourly, daily etc. depending on the granularity desired. If a correlation between sentiments and financial indexes is analyzed, a time frame of one minute might be appropriate, but for political moods a daily time frame might be applicable. The time series can then be represented graphically.

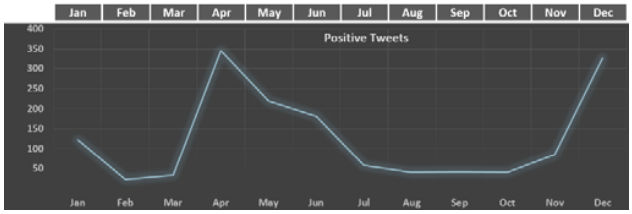

Fig. 2 Time series of positive Tweets

F. Correlations

To determine if a *correlation* between a financial index such as the Dow Jones Industrial Average (DJIA) and a time series exists, a visual analysis by overlaying the DJIA chart over the time series can be performed. Thus it can be determined; if for example a rise of the DJIA is preceded by an increase in positive Tweets. If a time series responds to certain events, this process has to be automated. Different approaches have been used. Bivariate Granger casualty analysis [6], statistics [14] and neural networks [4], [6] were applied. Granger causality analysis rests on the assumption that if a variable X causes Y then changes in X will systematically occur before changes in Y [4]. However the linear Granger causality test does not perform well in detecting nonlinear causal relationships and nonlinear models have been developed to overcome this limitation [19].

As with sentiment analysis, a neural network can be trained to automatically detect correlations. Training goes through the same steps as for classification. The training adjusts the variables that determine if a correlation exists. The latency l, is the time interval between a change in the time series and a change in real world data, d is the direction of the change, and i the increase or j decrease of the change. Depending on the domain the change of direction, d, up or down, might be enough, or the amount by which it increases, i, or decreases, j, might be needed. Twitter has a streaming API to access real-time Tweets. The trained model can be applied against this API to validate the predictions.

Many tools on the market have implementations of machine

learning algorithms and can be used for text mining and predictive analytics including IBMs SPSS predictive analytics software, SAS and Stata. The author is using the WEKA (Waikato Environment for Knowledge Analysis) open source tool for his research. WEKA has data preprocessing capabilities, implementations of most relevant machine learning algorithms, visualization facilities and can compare the performance of different learning methods. For large data sets WEKA can distribute the work load across multiple machines.

III. Challenges

Social media data are vast, noisy, distributed, unstructured and dynamic [11]. Finding the relevant Tweets is a challenge. Some Tweets are spam, give fake or false opinions or origin from users who pretend to be someone else (sockpuppet) [18]. Spam messages are often artfully crafted so they will not be detected by spam filters.

Opinion mining, and more generally text mining remains a challenging task. Automatically detecting sarcasm in a text is very difficult. Some studies could detect sarcasm in only 56% of the cases [1]. Languages are ambiguous and humor and innuendos cannot easily be analyzed using text mining techniques.

Finding the overall statement and the valence of an opinion remains a challenging task since there is no underlying truth, no "ground troth" to validate opinion against.

People tweet on a voluntary basis and not everybody is using SM. So there is a self-selection bias when using SM data.

Challenges in ML are selecting the training data (training data has to be as good as test data), selecting the features (which features are helpful), overfitting of a test set and finding the balance between simplicity and fit to data.

IV. Conclusions and Future Research

Predictive analytics using SM is an emerging area of research. But it remains a challenging task and offers many opportunities for future research.

Spam detection and generally determining the credibility of SM is an area where better filtering mechanisms will lead to better predictive results. More research is determining the trustworthiness of SM is thus highly desirable.

There seems to be much more research on ML than feature extraction. Feature extraction is a crucial step in predictive analysis and remains an area where many optimizations could be found [10].

Much of the traffic on SM sites originates now from mobile devices such as Smartphones or tablets. Mobile devices often give access to the geolocation. Including geospatial data in SM analysis could give useful new insights and improve the predictive capabilities and make an interesting area of research.

Finally, correlation does not mean causation. If X causes Y, this does not explanation as to why. Finding the causative mechanism would be another interesting area of future research.

References

[1] B. Liu, *Sentiment Analysis and Opinion Mining*. CA: Morgan & Claypool Publishers, 2012, ch. 1.
[2] S. Asur, and B. A. Huberman, "Predicting the Future with Social Media," in *Conf. Rec. 2010 IEEE Int. Conf. Web Intelligence*, pp. 492–499.
[3] E. Siegel, *Predictive analytics.* Hoboken, NJ: John Wiley & Sons, 2013, pp. 11.
[4] M. Arias, A. Arraita, and R. Xuriquera, "Forecasting with twitter data," *ACM Trans. Intell. Syst. Technol.*, vol. 5, no. 1, pp. 1-24, Dec. 2013.
[5] A. Tumasjan, I. M. Welpe, P. G. Sandner, A. Tumasjan, and T. O. Sprenger, 'Election Forecasts With Twitter: How 140 Characters Reflect the Political Landscape', *Social science computer review*, vol. 29, no. 4, 2011, pp. 402-18.
[6] J. Bollen, H. Mao, and X. J. Zeng, 'Twitter mood predicts the stock market', *Journal of Computational Science*, 2010, vol. 2, p. 8.
[7] H. Achrekar, A. Gandhe, R. Lazarus, Y. Ssu-Hsin, and L. Benyuan, 'Predicting Flu Trends using Twitter data', in *Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on*, 2011, pp. 702-7.
[8] T. Sakaki, M. Okazaki, and Y. Matsuo, 'Earthquake shakes Twitter users: real-time event detection by social sensors', *Proc. of the 19th international conference on World wide web*, Raleigh, 2010.
[9] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, 'Lexicon-based methods for sentiment analysis', *Comput. Linguist.*, 2011, vol. 37, no. 2, pp. 267-307.
[10] K. Tretyakov, 'Machine Learning Techniques in Spam Filtering', in Data Mining Problem-oriented Seminar, MTAT.03.177, 2004, Estonia.
[11] P. Gundecha, and H. Liu, 'Mining Social Media: A Brief Introduction', *informs*, 2012, vol. 9, pp. 1-17.
[12] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd edn, Morgan Kaufmann Publishers, Burlington, USA, 2011.
[13] I. Maks, and P. Vossen, 'A lexicon model for deep sentiment analysis and opinion mining applications', *Decis. Support Syst.*, 2012, vol. 53, no. 4, pp. 680-8.
[14] C. Diks, and V. Panchenko, 'A new statistic and practical guidelines for nonparametric Granger causality testing', *Journal of Economic Dynamics and Control*, 2006, vol. 30, no. 9–10, pp. 1647-69.
[15] K. Zhang, 'Big social media data mining for marketing intelligence', 3563913 thesis, Northwestern University, via ProQuest Dissertations & Theses A&I, 2013.
[16] D. Power, JP. Daniel and P-W. Gloria, 'Impact of Social Media and Web 2.0 on Decision-Making', *Journal of decision systems*, 2011, vol. 20, no. 3, p. 249.
[17] H. Schoen, D. Gayo-Avello, PT. Metaxas, E. Mustafaraj, M. Strohmaier and P. Gloor, 'The power of prediction with social media', *Internet Research*, 2013, vol. 23, no. 5, pp. 528 - 43.
[18] KL. Short, 'Buy My Vote: Online Reviews for Sale', *Vanderbilt Journal of Entertainment & Technology Law*, 2013, vol. 15, no. 2, pp. 441-71.
[19] Z. Bai, W-K. Wong and B. Zhang, 'Multivariate linear and nonlinear causality tests', *Mathematics and Computers in Simulation*, 2010, vol. 81, no. 1, pp. 5-17.

P. Wlodarczak born in Basel, Switzerland, holds a B.Sc. in computer science from the University of Applied Sciences in Zurich, Switzerland, a Master from the University of Southern Queensland, Toowoomba, Australia, and an EMBA from the University of Applied Sciences in Zurich, Switzerland. He is a research student at the faculty of Business and Law, University of South Queensland, Australia, currently doing a PhD in the area of SM and predictive analysis.