# A Comparison of Linear Discriminant Analysis and Ridge Classifier on Twitter Data

Anagh Singh
Department of Computer Science
and Engineering
National Institute of Technology
Karnataka, Surathkal-575025.
Email: anaghsingh.nitk@gmail.com

Shiva Prakash.B
Department of Computer Science
and Engineering
National Institute of Technology
Karnataka, Surathkal-575025.
Email: shiva96b@gmail.com

K.Chandrasekaran
Department of Computer Science
and Engineering
National Institute of Technology
Karnataka, Surathkal-575025.
Email: kchnitk@ieee.org

***Abstract*—This document is about the accuracy analysis of two of the most prominent classifiers present in today's academic arena. Classifiers are being used extensively in machine learning applications today and need to present a high rate of success to be considered useful. Tikhonov regularization incorporated within the Ridge Classifier is the basis for its classification. It utilises the LevenbergMarquardt algorithm for non-linear least-squares problems to classify objects. Linear Discriminant Analysis, on the other hand, utilises aspects of ANOVA[2,3] and regression analysis. LDA works by getting explicit information from the user. It needs the definition of the variables - both dependent and independent. It doesn't use any implicit assumptions in its modelling. There is no interconnection between the two variables initially. Using these two classifiers we compare their effectiveness at mapping a set of data scraped in real-time from Twitter to its corresponding generalised hashtag, and suggest why the differences, if any, arise.**

## I. Introduction

Machine learning has been slowly been an upcoming field which is maturing recently. It is expanding heavily now and and with data being produced daily at an astounding rate, it is poised to become mainstream within the next five years.

We may not know it but we are using machine learning everyday in our life. It has become an unnoticed necessity in our lives with our interaction with it increasing on a daily basis. Since data is going to increase in the future, it is a given that the study of that data will become omnipresent and will be a major component of technological advancements. The most used data structure in the process is a vector. However there is no dearth of problems which one might face when dealing with them - especially when the units of different comparisons don't align, or when their coordinates are different. This leads us to other data structures which one might use to fulfil the task like sets, bags, lists, queues and more advanced storage like videos and images themselves. Complex structures like these are used frequently and abound in day to day applications.

Query processing which was earlier done on the basis of direct string matching on databases is now slowly shifting to a machine learning approach. Usually a query is independent of its previous one but slowly websites are beginning to make all searches transitive. It is thought that the new search will depend somehow on the previous one. Thus it is, that web pages are also ranking them accordingly.

Another vague area is one of translation of elements - most notably text. This is highlighted by the Google Translate project initiated by Google. It was launched in 2006 with rule based machine translation base and followed it up a year later with statistical machine translation. It aims to automatically translate text. There can be two approaches to translation:

- Understand the linguistic features of a text completely before using a pre-defined set of rules for the language laid forth by a computational linguist.
- Compare a set of documents and get the differences. Using these differences we can learn which words belong to which particular context. This process is definitely not trivial, but once done, it makes translation easy since the task of understanding the entire complexity of the text is considerably reduced. Building on this, we could use the translations itself to learn new translations.

The latter approach is the one followed by many and has proved decidedly successful.

Machine Learning has been applied to a variety of problems the major ones being:

### A. Novelty Detection

This is a comparatively new application of machine learning. Herein we decide the novelty of an item depending on the past measurements which we have recorded. Since unusualness is a matter of one's perspective it is a debatable issue. However one can quantify unusual events as those which occur extremely rarely[27]. In this, an observation may be rated so as to develop a system which could rate events and determine whether it is novel or not.

### B. Binary Classification

Arguably the most common problem in machine learning, this has had a huge impact in the area of algorithmic machine learning leading to a development of large number of algorithms in the past 100 years [6,7,8,9]. It tries to answer a basic question: If we have a variable $x$ drawn from a domain $\chi$ then what will the value of a correlated random binary variable $y \in \{\pm 1\}$ be?

ISBN: 978-1-5090-1666-2/16/$31.00 ©2016 IEEE

### C. Multiclass Classification

This is an augmentation on binary classification. Here the value of the correlated random binary variable $y \epsilon \{1, \ldots, \text{n}\}$ and can be one of many values. Clearly the types of classes increases leading to a multiclass classification. Eg: Google Translate - if we want to translate a piece of text from English to Spanish or Sinhala to Portuguese or Czech to Malay, this is the class it would fall under, with the error factor being dependent on the amount of training done for that particular language.

### D. Regression

This is the type of classification which we have used in our experiment. The main point in these regression problems which make them different is that these problems account for loss too. Eg: Two scenarios - A stock market broker would always like to know the actual value from the regression[28] including any loss values to take an informed decision on his stocks. However consider a sportsperson who wants to check his pulse. He would not be interested in knowing the entire details preferring only to know the average value.

### E. Structured Estimation

This type is an extension of the multiclass estimation. Here we allow $y$ the freedom to have supplementary structures attached to it which are used distinctly in the weighing process. This allows us to estimate values more effectively. For example - $y$ can be one state of a finite state machine, $y$ can be a permutation of objects, $y$ may be used to filter objects or perform rank analysis of documents. $y$ may as well be text annotation in entity recognition based on names. All of them have their own set of associated properties depending how we need to modify our search space for them.

In this paper, we define the problem and speak a bit about the background of the problem. Post that we talk about the classifiers which we have used, and after that move on to the experimental observations. Our work will enable the reader to get an insight into the working of the classifiers described herein. Our research will allow one to understand the behaviour of these two classifiers on datasets. Since the data sets that we have used are general and vary across a wide range of users, featuring different views from different geographical locations, our work is applicable across a wide range of general data sets without being too specific or heavily reliant on one type.

## II. Background

In social media such as Twitter and Facebook, discussions hold significance when they are between people of power. In previous attempts [21,22,23,24] researchers have tried to determine how people use the internet and there have been several studies how people use sentiment analysis. Researchers have tried to establish metrics for communication on social networks [25] by cataloguing standardized metrics for use. Researchers have also tried to establish a correlation in social communication with times of crises and the management of crises [26]. Let us take a quick look at some of the concepts that have underlined the area.

### A. Data Science

The economy of today is growing rapidly and generating more data than it ever has previously. More the growth rate, more the spending, and thus more the the amount of data at hand for the economy. Firms rack up data at a rate faster than they can interpret and study it. And to combat this they want to generate ways of looking at data which take into account not just their own but also give an all encompassing look at other data which is available too, to see whether it is relevant and to use it if so.

This method of deriving observations from the data at hand falls under the purview of data science, and is materializing as a very strong abettor to Big Data - an umbrella term that comprises all of the data available - structured, semi - structured or unstructured that enterprises produce on a large scale.

Data science is in the limelight because the sudden spurt in generation of data - with data being harvested from websites, mobile sites, mobile usage patters, smart devices and social engineering. Data science mostly draws upon machine learning, query optimization, retrieval analysis, signal processing, statistical learning and natural language processing to evaluate data and scrutinize the results.

The compiled data is too large to fit onto a single disk, drive, or computer. It has to be handled in a distributed manner. Normal databases and classical statistical approaches with graphical software do not apply here. This data is not homogeneous - the kind which we are used to dealing with.

Data is obtained from sources which are of aprocryphal that their provenance cant be ascertained while quality is indeterminable. These consist of text which is digitized, for example - from OCR (optical character recognition), visual data, sensor data, and data from other textual sources such as blogs, books etc. With this kind of data at hand, ethics, privacy and security have to be at the forefront of all decisions that are made.

During this time, advancements in the industry are emerging which organise and succeed in interpreting the large amounts of data. With them we organise data and sort it for any regularities which it may exhibit. These help society creating value in the commercial sense and alleviate the human condition too.

Big data can further our understanding of physical and biological phenomena, and help in unravelling a host of others, benefiting humanity socially and economically [20].

### B. Web Scraping

Web scraping refers to the process of systematically scouring and collecting information from the World Wide Web. The solution may be an improvised one requiring manual intervention every time to a fully automated solution which can scour an entire website and collect appropriate information. Most sites have a file $robots.txt$ which tells a crawler which lets a user define what can be accessed and used. This site is useful for administrators for managing access to sites by having a global configuration file defined.

In the dynamic web of today, most pages are structured and are written in HTML, CSS, JavaScript. Some websites my have good content but may not provide data in an easy to read format like CSV or JSON. To resolve this issue, one would use web scraping which would tend to preserve the structure of the data.

- DOM parsing: A program can embed a complete web browser which will successfully trigger client side dynamic scripts too. Then the program can use controls to parse complete web pages into their appropriate DOM trees, essentially allowing one to retrieve a particular part of the page too.
- Web-scraping software: Numerous software tools for this purpose exist on the Internet. They can be customized according to the needs of the individual. Some may include code to allow one to define their own set of rules for extraction while others simply recognise the basic structure and can extract only limited data. After extraction, this data can be transformed into whatever type the user wants it to be transformed to.
- Regex matching: The UNIX command line utility - *grep* can be used in a powerful manner to extract text with regular expressions being supported by it too.
- Socket programming: Static HTTP pages as well as pages with dynamic content can be easily recovered with the help of socket programming.

### C. Natural Language Processing

The systems which we use in our daily life interact with us in artificial high level / middle level / low level languages like Java, C, Python etc. NLP refers to the interaction between systems and users using a natural language such as English for the exchange of information. A digital computer should be able to recognise the language and construct pertinent logical responses to answer back.

A natural assumption which one could make is that one can give the machine a dictionary, a set of words, a set of grammar rules and the type of speech. Thus a computer should be able to match words with their meanings and make contextual sense of sentences, with the process being entirely mechanical ie. all the information is gained only by querying the existing database and by classifying the sentence received.

NLP, if considered broadly, involves the following sub topics:

- Syntactic analysis
- Signal processing
- Pragmatics
- Semantic analysis

An NLP if constructed properly should not only respond in a manner befitting a human but should also be able to generate sentences and participate in conversations ( natural language sentences ) with another NLP, which can be either a computer system or a human.

Much of the discussion that takes place in NLP occurs in the domain of Artificial Intelligence, specifically sequential AI understanding. The end goal is to simulate an internal representation of the context ie. meaning which humans can understand and process the input (natural language) to a representation form which is easily interpreted. Usually the programs used for simulation of NLP tend to be sequentially heavy on a normal digital computer.

We use the bag of words approach[29] in our language processing efforts and try to extend it using latent semantic efforts. This allows us to create multiple sets, and check for any aggregation that might occur. Agglomerations of this sort when found, are grouped to allow for easier processing. We do use the *tf-idf* scheme to weigh and categorise words justly. It uses the document vectors' term frequencies. Our approach to the classifier is a mixture of a black box and a glass box approach. This allows us to run the system on a data set of our choice while concurrently examining its internals.

## III. Classifiers

This section describes the classifiers that we used to carry out the experiment. Text classification problems tend to be quite high dimensional (many features), and high dimensional problems are likely to be linearly separable (as any $d + 1$ points in a $d$-dimensional space with a linear classifier can be separated, regardless of how the points are labelled). So linear classifiers, whether ridge regression or SVM with a linear kernel, are likely to do well. In both cases, the ridge parameter or C for the SVM, control the complexity of the classifier and help to avoid over-fitting by separating the patterns of each class by large margins (i.e. the decision surface passes down the middle of the gap between the two collections of points). However to get good performance the ridge/regularisation parameters need to be properly tuned.

### A. Ridge Regression Classifier

In the scikit-learn implementation of the Ridge Classifier, there are three phases:

*1) Initialization Phase:* The Initialization phase involves entering the various parameters that control how the classifier works. These parameters are initialized during instantiation of the Ridge Classifier. The parameters are as follows:

- alpha : It is the regularization constant used to reduce the variation of estimations and improve classification.
- max iter : specifying the number of iterations used in the solvers.
- solver : The ridge classifier has a lot of built in solvers used int the training of data. The auto option selects a suitable solver for specific cases. Some of the solvers are : cholskey, cholskey kernel, sparse cg, lsqr .

*2) Fit Phase:* In the fit phase, we feed a matrix X and a vector Y to the classifier. Every row x of the matrix X corresponds to the feature vector that maps to the class y that is in the corresponding element in the vector Y. The Classifier then learns a model from this data, it generates a coefficient vector that best fits all this data. The Ridge Classifier works on a linear model i.e. the coefficient vector is used to represent the coefficients of a linear equation where, the elements of the feature vector x are the variables.

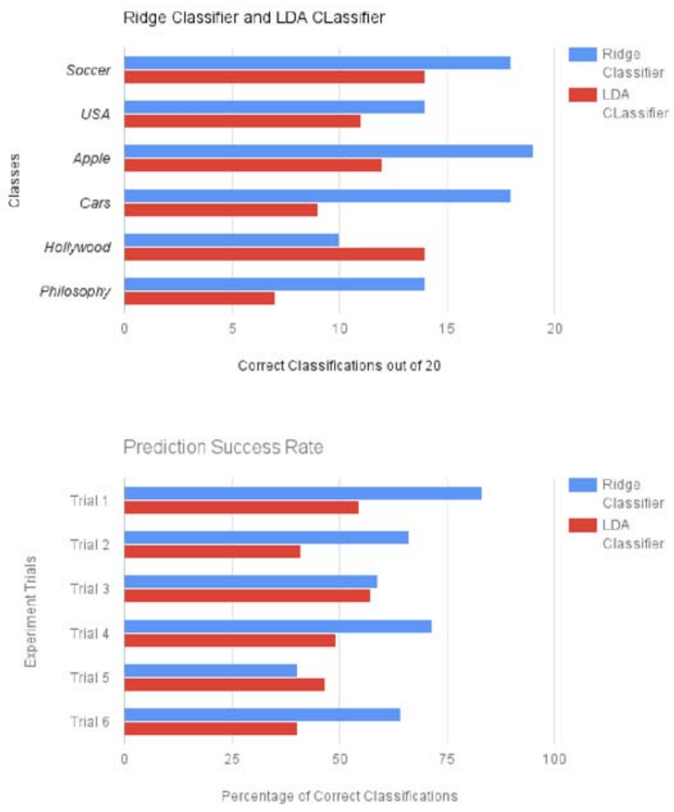

Fig. 1. Variation in Prediction

The scikit-learn implementation uses a one-vs-all approach while classifying. To do this , the classifier generates separate vectors Y for , every class in the original result vector, where it takes one of the classes , in the original result vector , as the base class and maps it to the value ’1’ and maps all other classes to the value ’-1’ . In this way multinomial classification, is reduced to separate binomial classifications.

The above conversion is performed by an instance of the LabelBinarizer class in the scikit-learn library. These feature vectors are generated by minimizing the cost function. Now, this can be done in several ways. In the scikit-learn library the method can be selected by changing the solver parameter during the initialization step of the Classifier.

*3) Predict Phase:* In this phase the matrix is fed to the classifier, and it is the job of the classifier to generate the classes corresponding to every row of the matrix. It does so by using the coefficient matrix that was generated during the Fit phase.

The vectorizer matrix corresponding to every test case is multiplied with a scalar matrix constituting of weights belonging to respective features. The size of both the matrix is the length of the feature vector. The scalar product results in a value for different weight vectors for the classes. The maximum value among these is mapped to the respective class.

### B. Linear Discriminant Analysis Classifier

The primary importance of Linear Discriminant Analysis comes into picture when there is a necessity to reduce the dimensionality which happens a pre-processing procedure. The aim of LDA is to reduce the dimensionality space of data to remove the problem of over fitting. Linear discriminant analysis (LDA) is a generalization of Fisher’s linear discriminant. The result is used as a linear classifier.

LDA prefers to choose a line that separates the vectors the most. The line is a graphical representation of projections. The above criteria is satisfied by following the Fischer’s principle of maximizing the difference between the means. This method can be extended to multinomial distribution as well. An alternate to this method is the principal component analysis.

## IV. Experimental Observation

The aim of the experiment is to compare the accuracy of prediction of the ridge classifier and the LDA classifier. Both these classifiers fall under the category of supervised classifiers. The data used for classification experiment is taken from tweets which vary over different classes of trending data. Raw data obtained from the tweets has to be processed before in can be fed to the classifier. This is because the frequency of occurrence of features of the classifiers have to be increased to have a better accuracy.
The experiment is repeated six times with different data sets. Each data set contains tweets which belong to different categories of tweets. The initial part of cleaning the data is common for the classification of both the classifiers. The complete ad detailed procedure of the experiments is mentioned ahead:

### A. Extracting Data

The data as mentioned is extracted from the twitter and this data can be extracted for research and educational purposes from twitter. Twitter provides a Tweepy library through which users can access live tweets by registering themselves on twitter and by accepting the terms for using the data only for academic and research purposes.
The data is extracted after going past the authentication by twitter servers. Data from different topics have to be considered to conduct the classification. Since the data is supervised we get tweets from 6 different topics. In the first trial we have taken data from Soccer, USA, Apple, Cars, Hollywood and Philosophy. The Tweepy library provides an option to filter tweets based on topic and also set a limit on the number of tweets to extract. However when the tweet is stored in the file it contains a lot of garbage values which have to cleaned. The cleaning algorithm used here is as follows :

- Removal of all non-ASCII characters using regular expressions.
- Going through each word in the text file which contains the tweets and check for spelling errors and removing some of the meaningless words whose frequency of occurrence is high and has a chance of becoming a feature vector. This process was done for the first data set manually.

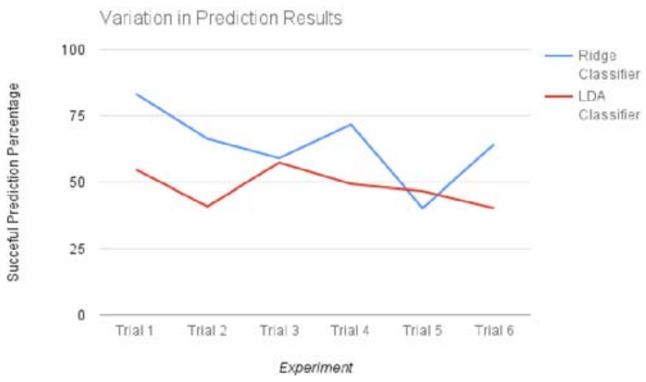

Fig. 2. Prediction comparison between Ridge and LDA Classifier

- Removing the # tags with their respective topics from the tweets so that the classification algorithm is put to a real test.
- Before feeding it to the classifier the stop words which provide no meaning to the classification is removed.
- A Lancaster stemmer is used to stem the tweets so that all the words are brought down to their root word hence reducing the number of features and mapping all the derived words to their root word.

TABLE I. RESULTS OF THE CLASSIFICATION FOR TRIAL 1 FOR CORRECT MAPPING OUT OF 20 TESTS

| Class | Ridge Classifier | LDA Classifier |
|---|---|---|
| Soccer | 18 | 14 |
| USA | 14 | 11 |
| Apple | 19 | 12 |
| Cars | 18 | 9 |
| Hollywood | 10 | 14 |
| Philosophy | 14 | 7 |

### B. Vectorizer

A Vectorizer converts documents of text into a matrix of token counts. The output of a Vectorizer is generally represented in the form of a sparse matrix because of the wide variety of vocabulary. The count Vectorizer uses the bag of words approach in the vectorization process.

A Vectorizer converts documents of text into a matrix of token counts. The output of a Vectorizer is generally represented in the form of a sparse matrix because of the wide variety of vocabulary. The count Vectorizer uses the bag of words approach in the vectorization process. The document is initially passed through the analyzer in the Vectorizer and it undergoes the following operations / functions:

- Preprocessing : It strips the accents (ASCII is faster to process when compare to Uni-code)
- Tokenizing : A tokenizer divides strings to a set sub strings and words depending on the parameters

If a predefined dictionary is not present and the feature selection is not specified, then the number of features in the vectorizer would be the size of the vocabulary of the documents.
The count vectorizer uses only the term frequency to classify documents. Some of the words may have a very high frequency and occur in all documents. Hence the importance is increased in this approach even though their weight age in classification should be negligible. To overcome this problem we use the “Term frequency - inverse document frequency“ approach.

The goals of this approach are:

- to reduce the weight age of words that occur in most of the documents (which doesnt help
- remove the approach that uses just raw frequencies

The two parameters are:

- Term frequency: measures how frequently a term occurs in a document

  $$tf = \frac{(Numberoftimestermtappearsinadocument)}{(Totalnumberoftermsinthedocument)}$$

- Inverse Document frequency: This parameter defines how much weight age must be given to a particular term

  $$IDF = log_e \frac{(Totalnumberofdocuments)}{(Numberofdocumentswithtermtinit)}$$

The product of $tf$ and $idf$ is $(tf - idf)$ can be successfully used for stop-words filtering in various subject fields including text summarizing and classification.

### C. Feeding data to the classifier

Now we have the vectorized data either by using the count vectorizer or $tf - idf$ vectorizer. This data can be fed to the classifier as a result of vectorization. The vectorized data ca be used by both the ridge and LDA classifier. Both the classifiers fall under the category of supervised classifiers. A supervised classifier predicts output from a given set of previously defined output for different mappings. Here the classes to which the tweets have to classified are previously defined.
Any Classification contains two parts :

- Training : The data which is used by the classifier to build the feature vector which is used to classify the data. In this phase the tweets along with the classes or topic it belongs to is fed to the classifier. The mapped details help the classifier learn from the data.
- Testing : In this phase new data is given to the classifier and by using the previous mapped tweets it predicts the classes to which it belongs to.

For each trial in this experiment we take tweets from six different categories belonging to a trending topic. In each trial a 90% - 10% ratio is used for training - testing. For each trial 1200 tweets are taken which are evenly distributed among the six different classes taken in each. Therefore 1180 fed to the classifier for training and the rest 12o are tested.

### *D. Results*

TABLE II. Percentage Accuracy of Prediction

| Trials | Ridge Classifier | LDA Classifier |
|---|---|---|
| 1 | 83.3 | 54.7 |
| 2 | 66.3 | 41.0 |
| 3 | 59.0 | 57.3 |
| 4 | 71.7 | 49.3 |
| 5 | 40.3 | 46.7 |
| 6 | 64.3 | 40.3 |

## V. Conclusion and Future Work

The reason that ridge regression works well is that non-linear methods are too powerful and it is difficult to avoid over-fitting. There may be a non-linear classifier that gives better generalisation performance than the best linear model, but it is too difficult to estimate those parameters using the finite sample of training data that we have. In practice, the simpler the model, the less problem we have in estimating the parameters, so there is less tendency to over-fit, so we get better results.

Ridge regression avoids over-fitting by regularising the weights to keep them small, and model selection is straight forward as we only have to choose the value of a single regression parameter. If we try to avoid over-fitting by picking the optimal set of features, then model selection becomes difficult as there is a degree of freedom for each feature, which makes it possible to over-fit the feature selection criterion and we end up with a set of features that is optimal for the particular sample of data, but which gives poor generalisation performance. So not performing feature selection and using regularisation can often give better predictive performance.

## References

[1] Jackson, Peter, and Isabelle Moulinier. Natural language processing for online applications: Text retrieval, extraction and categorization. Vol. 5. John Benjamins Publishing, 2007.

[2] Hargrove, Levi J., et al. ”Multiple binary classifications via linear discriminant analysis for improved controllability of a powered prosthesis.” Neural Systems and Rehabilitation Engineering, IEEE Transactions on 18.1 (2010): 49-57.

[3] Liu, Yan, and Gavriel Salvendy. ”Effects of measurement errors on psychometric measurements in ergonomics studies: Implications for correlations, ANOVA, linear regression, factor analysis, and linear discriminant analysis.” Ergonomics 52.5 (2009): 499-511.

[4] Smola, Alex, and S. V. N. Vishwanathan. ”Introduction to machine learning.” Cambridge University, UK (2008): 32-34.

[5] Dmoz.org, (2015). DMOZ - the Open Directory Project. [online] Available at: https://www.dmoz.org/ [Accessed 27 Oct. 2015].

[6] Joachims, Thorsten. Text categorization with support vector machines: Learning with many relevant features. Springer Berlin Heidelberg, 1998.

[7] Fan, Rong-En, et al. ”LIBLINEAR: A library for large linear classification.” The Journal of Machine Learning Research 9 (2008): 1871-1874.

[8] Kotsiantis, Sotiris B., I. Zaharakis, and P. Pintelas. ”Supervised machine learning: A review of classification techniques.” (2007): 3-24.

[9] Hsu, Chih-Wei, and Chih-Jen Lin. ”A comparison of methods for multiclass support vector machines.” Neural Networks, IEEE Transactions on 13.2 (2002): 415-425.

[10] He, Jinrong, et al. ”Kernel ridge regression classification.” Neural Networks (IJCNN), 2014 International Joint Conference on. IEEE, 2014.

[11] Tikhonov A.N., Goncharsky A.V., Stepanov V.V., Yagola A.G., 1995, Numerical Methods for the Solution of Ill-Posed Problems, Kluwer Academic Publishers.

[12] Hoerl AE, 1962, Application of ridge analysis to regression problems, Chemical Engineering Progress, 1958, 5459.

[13] Wahba, G. (1990). ”Spline Models for Observational Data”. Society for Industrial and Applied Mathematics.

[14] Golub, Gene H., Michael Heath, and Grace Wahba. ”Generalized cross-validation as a method for choosing a good ridge parameter.” Technometrics 21.2 (1979): 215-223.

[15] Hansen, Per Christian. ”Analysis of discrete ill-posed problems by means of the L-curve.” SIAM review 34.4 (1992): 561-580.

[16] Fisher, Ronald A. ”The use of multiple measurements in taxonomic problems.” Annals of eugenics 7.2 (1936): 179-188.

[17] Sebastianraschka.com, (2015). Linear Discriminant Analysis bit by bit. [online] Available at: http://sebastianraschka.com/Articles/2014_python_lda.html [Accessed 24 Oct. 2015].

[18] Duda, Richard O., Peter E. Hart, and David G. Stork. Pattern classification. John Wiley & Sons, 2012.

[19] Mind.ilstu.edu, (2015). Introduction to Natural Language Processing - The Mind Project. [online] Available at: http://www.mind.ilstu.edu/curriculum/protothinker/natural_language_processing.php [Accessed 1 Nov. 2015].

[20] Datascience.berkeley.edu, (2015). What is Data Science?. [online] Available at: https://datascience.berkeley.edu/about/what-is-data-science/ [Accessed 28 Oct. 2015].

[21] Raghuram, M. A., K. Akshay, and K. Chandrasekaran. ”Efficient User Profiling in Twitter Social Network Using Traditional Classifiers.” Intelligent Systems Technologies and Applications. Springer International Publishing, 2016. 399-411.

[22] Pang, Bo, and Lillian Lee. ”Opinion mining and sentiment analysis.” Foundations and trends in information retrieval 2.1-2 (2008): 1-135.

[23] Pang, Bo, and Lillian Lee. ”A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts.” Proceedings of the 42nd annual meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2004.

[24] Nasukawa, Tetsuya, and Jeonghee Yi. ”Sentiment analysis: Capturing favorability using natural language processing.” Proceedings of the 2nd international conference on Knowledge capture. ACM, 2003. APA

[25] Bruns, Axel, and Stefan Stieglitz. ”Towards more systematic Twitter analysis: Metrics for tweeting activities.” International Journal of Social Research Methodology 16.2 (2013): 91-108.

[26] Terpstra, Teun, et al. ”Towards a realtime Twitter analysis during crises for operational crisis management.” Proceedings of the 9th international ISCRAM conference, Vancouver, Canada. 2012.

[27] Tiitinen, Hannu, et al. ”Attentive novelty detection in humans is governed by pre-attentive sensory memory.” (1994): 90-92.

[28] Yang, Haiqin, Laiwan Chan, and Irwin King. ”Support vector machine regression for volatile stock market prediction.” Intelligent Data Engineering and Automated LearningIDEAL 2002. Springer Berlin Heidelberg, 2002. 391-396.

[29] Wallach, Hanna M. ”Topic modeling: beyond bag-of-words.” Proceedings of the 23rd international conference on Machine learning. ACM, 2006.