Machine Learning Techniques for Sentiment Analysis: A Review

Munir Ahmad[1], Shabib Aftab[2], Syed Shah Muhammad[3] and Sarfraz Ahmad[4]
[1-4]Department of Computer Science, Virtual University of Pakistan
[1]munirahmad@gmail.com, [2]shabib.aftab@gmail.com

Abstract– **Social media platforms and micro blogging websites are the rich sources of user generated data. Through these resources, users from all over the world express and share their opinions about a variety of subjects. The analysis of such a huge amount of user generated data manually is impossible, therefore an effective and intelligent technique is needed which can analyze and provide the polarity of this textual data. Multiple tools and techniques are available today for automatic sentiment classification for this user generated data. Mostly, three approaches are used for this purpose Lexicon based techniques, Machine Learning based techniques and hybrid techniques (which combines lexicon based and machine learning based approach). Machine Learning approach is effective and reliable for opinion mining and sentiment classification. Many variants and extensions of machine learning techniques and tools are available today. The purpose of this study is to explore the different machine learning techniques to identify its importance as well as to raise an interest for this research area.**

Keywords– **Machine learning, Sentiment Analysis, Opinion mining, Social media, Polarity Detection**

I. INTRODUCTION

Due to the rapid growth of social media, bulk of user generated data is available now. Analyzing the sentiments and accurate classification of this gigantic amount of data is a very challenging task. Most of the data available on the internet is in the textual form as it is the most natural and readable form for presenting the thoughts and opinions to the users [1]. In this research Machine Learning algorithms and techniques for sentiment analysis are deeply analyzed. These algorithms are more adaptive to changing inputs.

Unigrams (single word), Bigrams (dual word) and N-grams (multi words) are used by different algorithms for data labeling and data processing. Machine learning techniques are generally used for binary classification and predictions of sentiments as either positive or negative. Machine learning algorithms are further classified in the following categories as explained by [2].

Supervised: In these algorithms training dataset with the pre labeled classes are given and on the basis of this trained dataset the inputs are labeled with the output class/result [3]. These algorithms classify the input data set with the help of trained classifier. Training data is composed of a set of training examples, each of them comprise of input object and desired output results [4]. An inferred function is created by analyzing the training data by supervised learning methods that can be later used for mapping new incoming data which is also called the test data. Mostly machine learning techniques use the Supervised approach. It can be further categorized in two methodologies i.e., Classification and Regression. Most common examples of supervised machine learning algorithms are Linear Regression, Random Forest and Support Vector Machines.

Un-supervised: These type of machine learning algorithms takes the unlabeled input data and then with the help of different algorithms hidden structure/pattern is discovered Unlike the supervised learning this technique does not use the pre labeled data to train the classifier. Un-supervised machine learning can be further divided into clustering and association, the most common example of Un-supervised machine learning algorithms are K-Means and Apriori Algorithm.

Semi-Supervised: These type of algorithms deal with the both labeled and unlabeled data sets. [6] reviewed different lexicon based tools and techniques and mentioned the comparison between the features and accuracy results of different lexicon techniques. Taking it a step ahead, different Machine Learning techniques/algorithms are studied and analyzed in this research. A comprehensive analysis is also formulated between different techniques and accuracies.

II. RELATED WORK

A lot of work has already been done in the field of Sentiment analysis by using machine learning methods, Opinion mining is the process of categorizing the unstructured data and text into positive, negative and neutral categories. In the recent years microblogging platforms like Facebook and Twitter attracted millions of users around the globe to give them open platform to share their thoughts and opinions as described by [7]. Usually sentiments are classified in binary form either positive or negative as described by [8], [9]. There are several machine learning methods for sentiment categorization such as Maximum Entropy (MaxEnt) which is a feature-based model and doesn't takes independent assumptions as explained by [10]–[12]. Stochastic Gradient Descent (SGD) is another machine learning based algorithm that is capable of making the classifier learn even if it is based on non-differentiable loss function as explained by [13]. The [14] introduced Random Forest that targets the enhancing and storing of classification trees. SailAil Sentiment Analyzer (SASA) is another machine learning based sentiment classification algorithm as described by [2]. A robust & non-linear neural network model based on machine learning method was introduced by [15] as Multilayer Perceptron (MLP). Naïve Bayes is another famous, easy to implement and efficient machine learning algorithm and was originally proposed by Thomas Bayes as explained by [15]. Support Vector Machine (SVM) is one of the most famous supervised machine learning based classification algorithm [16]. Different studies and researches featuring different algorithms and tools are available on Machine Learning based sentiment analysis but as compared to several other methods SVM performed better in terms of accuracy and efficiency as the results were comparatively higher according to [17], [18].

III. MACHINE LEARNING TOOLS

A) Maximum Entropy (MaxEnt)

The [10], [11], [12] described that Maximum Entropy (MaxEnt) models are feature-based and do not take independence assumptions. We can add features using bigrams and phrases to MaxEnt with no feature overlapping. The exciting idea behind Maximum Entropy models is that one should favor the most uniform models that fulfil a given constraint as described by [19]. These feature based models can be used to estimate any probability distribution. Finding a distribution over classes using logistic regression is the same as in a two-class scenario. In a four-way text classification where we know that 40% of the documents contain the word "teacher", belong to the teacher class. Intuitively, if a document with the word "teacher" in it has a 40% chance of being member of teacher class then there would be 20% chance for each of remaining three classes. When the document does not contain the word "teacher" then we can guess the uniform class distribution for 25% each. This model is maximum entropy. In this example it's easy to calculate the model but when having many constraints then rigorous methods are required to find the optimal solution.

Naïve Bayes take independent assumption for its features but MaxEnt does not. We can add features using bigrams and phrases to MaxEnt and there will be no feature overlapping.

The following equation shows the model:

$$P_{ME}(c|d,\lambda) = \frac{\exp[\Sigma_i \lambda_i f_i(c,d)]}{\Sigma_{c'} \exp[\Sigma_i \lambda_i f_i(c,d)]}$$

In the formula above, class is represented by c, tweet with d, weight vector is represented by λ, an each f_i (c,d) represents a feature. The higher wait indicates that the feature is a strong indicator for the specified class. The Importance of the feature in classification is decided through weight vectors. Feature is strong pointer if the weight is higher, for classification of MaxEnt the authors has used Stanford classifier and also used conjugate gradient ascent for training the weights. Maxent handles overlapping features better than Naïve Bayes(NB) whereas in practice NB performs well on different problems according to [19].

B) Stochastic Gradient Descet (SGD)

The [13] stated that Stochastic Gradient Descent (SGD) delivers an efficient means to learn some classifiers even if they are based on no-differentiable loss function (hinge loss) used in SVM and it can also adapt changes over time. An implementation of vanilla SGD having a fixed learning rate was experimented by them, hinge loss was optimized with an L_2 penalty which is commonly used to study support vector machines. The formula that was derived for document classification is as under:

$$\frac{\lambda}{2}||\mathbf{w}||^2 + \sum [1 - (y\mathbf{x}\mathbf{w} + b)]_+,$$

Here in the formula the optimized loss function x, w represents the weight vector, b represents the bias, λ represents the regularization parameters while the class labels y are suspected to be (+1,-1).

The performance of this methodology was compared with Pegasos method that was purposed by [20] and do not require specification of an explicit learning rate, but no performance improvements were detected using the latter discussed algorithm. On the other hand, the ability of the algorithm to calculate the explicit learning rate appeared to be an important task with the changing twitter data streams. In the experiments carried out by the researchers, they used λ = 0.0001 and kept the learning rate set for the per-example updates to the classifier's parameters to 0.1.

C) Random Forest

The [14] described Random Forest as ensemble learning method that targets enhancing and storing of classification trees. Tree predictors are arranged in such a format that every single tree is dependent on independently patterned values of random vectors and all the trees are distributed uniformly across the forest. As defined random forest is a classifier comprised of tree structured classifiers {h(x,Ok), k=1,…} having {Ok} as independently identically distributed random vectors and each tree casts a unit vote for the most popular and famous class at input x.

Random forests have been effectively applied to numerous complex difficulties in genetic epidemiology and microbiology within the last few years. Within a very short period of time, random forest among other standard methods became a major data analysis tool. [21] has done research on Classification by Ensembles from Random Partitions.

According to the authors, Classification is always challenging job. On the base of ensembles of classifier, they develop a robust procedure for the classification. This classifier predicts the random partition of the entire set of predictions. The proposed method combines the effect of multiple classifiers to achieve a sensationally improved prediction compared to the original classifier. This approach is designed specifically for high dimensional data sets for which classifier is sought. This classifier named as CERP. This new ensemble based classifier built with the help of classification trees (C-T CERP) and logistic regression (LR-T CERP) trees as base classifier. Result of this classifier showed high accuracy among other classifiers. Result shows that C-T CERP is less dependent on the threshold choice as comparing with the LR-T CERP.

[22] throws light on the fuzzy Random forest. In this study, authors presented fuzzy decision tree as FRF ensemble. A hybridization of the techniques fuzzy trees and random forest for the training was analyzed. The projected ensembles got the benefit of imperfect data management of being robust noise and also having excellent extend of classification with small ensembles. The imperfect datasets and the result gained by the FRF ensembles are very promising. The FRF ensembles has a good performance with the datasets with fuzzy values. The weighted combination method performs better as compared to non-weighted method when using with these datasets. Compared to non-weighted method typically used in the random comparing the FRF ensembles.

D) SailAil Sentiment Analyzer (SASA)

The [2] introduced a learning tool SailAil Sentiment Analyzer (SASA) which is based on SentiStrength 2.0, which can be downloaded from http://sentistrength.wlv.ac.uk/Download. it has been tested by the Amazon Mechanical Turk, 17,000 tweets related to US presidential elections 2012 were labeled by the turkers as either these tweets are positive, negative, have neutral sentiments or any undefined sentiments are present. IBM's InfoSphere Streams platform was used for the development of this real time data processing infrastructure and to achieve better performance, speed and accuracy enabling to write analysis and visualization modules and assembling them into a unified real-time processing architecture.

In [2] twitter was selected as data source as it is the most suitable and real time regarding serval events happening around. Only 1% or less of the entire twitter's traffic is provided by the twitter API, without any specific sampling controls. Subject relevant real time tweets were collected using Gnip Power track by the researchers. After collecting twitter data, preprocessing was done on the target data as the twitter text data is different from the text in the articles, books and even the spoken context having many idiosyncratic uses like emoticons, URLs, RT as in retweets, @ is used for user mentions, # sign is used for hashtags. All these slangs were cleaned and the target text was normalized for further processing.

The algorithm design was based on the assumptions that text in the tweets and the users opinions would be highly subjective and contextualized. For data model training and testing, the crowd sourcing approach was used for the sentiment annotation of the in-domain political data. About 800 turkers were engaged for the annotation from the Amazon Mechanical Turk (AMT) so that most varied the population of annotators can be utilized. An interface having the capability to perform annotation outside the AMT was designed for the Turkers allowing them to annotate anonymously. The turkers were then asked for their personal information including their age, gender and political orientation, afterwards they were presented with a series of random tweets to annotate and classify the sentiment from it as either positive, negative, neutral or unsure. The training data for SASA was comprised of about 17000 tweets having 16%, 56%, 18% and 10% positive, negative, neutral and unsure classifications respectively.

The statistical classifier that was used in the experimental procedures for sentiment analysis was based on naïve Bayes models and the unigram features. Tokenization of the tweets method was used for features calculation by this algorithm while preserving the punctuations that may signify specific type of sentiments like the use of emoticons and exclamation points. The newly developed classifier opted for 59 % of accuracy on the four category classification of positive, negative, neutral and unsure, based on the data that was collected leaving behind the baseline of sorting the data as negative that was previously 56%. As the algorithm receives the tweets rapidly and continuously and multiple rules are used for tracking different types of tweets within a given time period, the algorithm outputs the number of tweets per minute for all categories to analyze the volume and outputs the number of positive, negative, neutral and unsure tweets in a specific sliding five minutes window for analyzing sentiment classification of the provided data.

E) Multilayer Perceptron (MLP)

The Multilayer Perceptron (MLP) is a robust & non-linear neural network model as described by [15], MLP operate as universal function approximator having at least 1 hidden layer and multiple non-linear units making it efficient to learn any relation between input variable sets. Multilayer Perceptron (MLP) has a uni-directional flow of data just like data flowing from the input layer to the output layer. The Neural Network that multilayer perceptron (MLP) starts with the input layer having every node as a predictor variable. Neurons (input nodes) are interconnected with the neurons in the forward flowing and the next layer (labeled as the hidden layer). Similarly the hidden layer neurons are connected with the other hidden layer neuron and so on, the structure of output layer is depicted below:

i). In case the prediction is binary, output layer is composed of a single neuron

ii). In case the prediction is non-binary, the output layer is then composed of N neurons

Having the neurons patterned in this format results in an efficient flow of information from input to output layer. As shown in the image below there exist an input layer and an output layer as in a single layer perceptron but in parallel there exist a hidden layer network in the same algorithm.

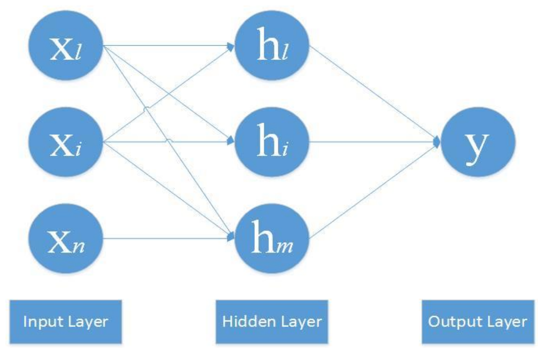

Fig. 1: MLP Architecture

Multi-layer perceptron has two phases in the first phase the activation is propagated from the input layer to the output layer, it is called the forward phase. In the second phase the errors between the real & operational values and the requested nominal values is replicated in the reverse direction. Due to its application as a universal function approximator MLP is a famous algorithm, having at least one hidden layer with multiple non-linear objects that can learn almost all functions or relationships within a given input and output variable set because of its "back propagation"

MLP does not start from any particular assumptions neither it does enforce any constraints on input data, it can also evaluate the data despite of the presence of any noises or distortions in the information.

F) Naïve Bayes (NB)

According to [15] Naïve Bayes Classifier originally given by Thomas Bayes is easy to implement and performs efficient computing compare to other machine leaning algorithms. It is a supervised classifier used to calculate the probability of a data to be positive or negative. [23] reported different challenges faced by the researchers and need more research to over-come these challenges.

The most efficient and effective inductive learning algorithm for machine learning and data mining is Naïve Bayes. It is based on Baye's Theorem with an assumption of independence among predictors. For the real world application its competitive performance in classification is surprisingly rarely true. In simple words, Naïve Bayes classifier assume that predefined properties are unrelated to presence of any other feature exists. Naïve Bayes Model is very useful for large data sets and is based on the Bayes theorem, and it specifies the relation between the probabilities p of two events c and Z and represented as P(c) and P(Z) and the conditional probability of event c conditioned by event Z and vice versa and represented as P(c | Z) and P(Z | c). Thus the Baye's Formula would be:

$$P(c\,|\,Z) = \frac{P(c)P(Z\,|\,c)}{P(Z)}$$

Typically, an example Z is represented by a tuple of attribute values (t1,t2,,,,,,,tn) where ti is value of attribute Ti. Consider C is classification variable and c is the value C. Let we consider two classes positive class denoted by + and negative class denoted by -. [24] evaluated the accuracy by training the Naïve Gauss Algorithm using 5000 sentences and got 0.79939209726444 accuracy and the number of groups (n) was 2. The major advantages of Naïve Bayes classification methods for sentiment analysis is that is easy to interpret and the results are calculated efficiently. While having the assumptions of attributes being independent is a drawback of this algorithm as it might not be valid all the time.

G) Multinomial Naïve Bayes

The Multinomial Naïve Bayes classifier is also based on the Bayes Theorem as referred by [13], it is a popular classifier used for document level sentiment classification and relatively yields good output and performance. This algorithm can be trivially used and applied for the data stream as it plays straightforwardly for updating the counts that are require to estimate the conditional and algorithmic probabilities.

A given document to the algorithm for classification is considered as a collection of words having each class c, P(w|c) as the probability of observing the given words w in the respective class. Then it is estimated from the given training data by simply computing the relative frequency comprehensively for all the given words in the collection of training documents of that particular class. The classifier also requires a straightforward to estimate prior probability referred as P(c) as shown in the given formulae.

$$P(c|d) = \frac{P(c)\prod_{w\in d} P(w|c)^{n_{wd}}}{P(d)},$$

P(d) refers to a normalization factor, and to avoid the zero frequency problem the Laplace correction for every conditional and algorithmic probability involved is used, making all the count values initialized to one instead of zero.

H) Support Vector Machine (SVM)

Support Vector Machine (SVM) as described by [16] is a supervised learning model with efficient result in traditional text categorization leaving behind Naïve Bayes and Max Ent. SVM was originally introduced by [25] it basically locates best possible boundaries to separate between positive and negative training samples and are extensively used because of their exceptional performance over other methods used in most machine learning models as referred by [26] and [27], but there are some complexities as discussed by [28] and [29] that need further research to overcome them.

There are multiple extensions available for SVM making it more efficient and adaptable to real world requirements. ***Soft Margin Classifier*** is one of the SVM extension that classifies most of the data and ignores any outliers and noisy data as the data is sometimes linearly visible for multi-dimensional problems and can be separated linearly. ***Non-Linear Classifier*** is another extension of SVM originally proposed by [7], kernel is used in No-Linear SVM to max the margin hyper planes. Generally SVM and its extensions are used for binary class tasks but for multi-class problems ***Multi-Class SVM*** extension is available having labels designed to objects that are drawn from a finite set of multiple elements as described by [15].

SVMPSO as described by [30] uses PSO which is based on Swarm Intelligence Optimization Technique to refine parameters of general SVM. SVMPSO resulted better than SVM in accuracy and efficiency in this study.

SVM can handle linear separation on high dimensional non-linear input data using an appropriate kernel. Multiple kernel functions such as Polynomial kernel of degree, Gaussian Radial Basic Kernel and Sigmoid kernel are available. Gaussian Radial Basic kernel function (RBF) is exceptionally better having kernel hyper parameter (gamma) and soft margin constant C as described by [31].

LIBSVM is described by [32] and was originally proposed in (2001), it is a famous library for support vector machines (SVM) used for machine learning operations such as Classification, regression and other learning methodologies. Typically LIBSVM obtains a model by using a training dataset and then use this model to predict information of a testing data.

IV. DISCUSION

In this research work different machine learning techniques and algorithms are studied and the performance on different data sets is discussed from past researches in Table I. The table briefly represents and explains the average & respective accuracy of all the examined machine learning based sentiment classification tools and techniques. Different training and datasets were used for the evaluation of these tool by the researchers i.e., movie reviews and product reviews, while some tools were tested on generic datasets and average accuracy of the tools were considered. A list of Machine Learning Based tools with their features and accuracy is provided in the below mentioned Table I.

V. CONCLUSION

In this paper several researches and studies are consulted on Machine Learning based tools and techniques for sentiment analysis and classification. Well known algorithms/techniques are discussed which are usually used for sentiment classification. A comparison in terms of accuracy on different datasets is given that can be used as a ready reference in future research works.

Table I: Tools, Features and their accuracy on different datasets

SR#	Author Name	Publication Year	Tool Name	Features	Accuracy		
					Movie	Product	Average
1	K. Nigam, J. LaÆerty [14]	1999	Maximum Ent	- Feature based Model - Uses, bigrams and phrases	-	-	72.60%
2	L. Breiman [17]	2001	Random Forest	- Ensemble learning method - Uses Classification trees and tree predictors - comprised of tree structured classifiers	-	-	88.39%
3	Wang, H., Can, D., Kazemzadeh, A., Bar, F. and Narayanan, S. [1]	2012	SASA	- Based on sentistrength ver. 2.0 - Used by Amazon AMT - Uses IBM's InfoSphere for speed and performance.	-	-	64.90%
4	P.K Singh and M.Shahid Hussain [20]	2014	MLP	- Robust & non-linear neural network model - Used as universal function approximator - it have one hidden layer and multiple non-linear units	81.05%	79.27%	-
5	P.K Singh and M.Shahid Hussain [20]	2014	Naïve Bayes	- Supervised classifier for binary classification - Based on Baye's Theorem - Useful for large datasets	75.50%	62.50%	-
6	C. Cortes and V. Vapnik [24]	1995	SVM	- based on supervised learning model - can handle linear separation on high dimensional non-linear input data using an appropriate kernel - multiple derivatives and extensions are available	81.15%	79.40%	-

REFERENCES

[1] I. Smeureanu and M. Zurini, "Spam Filtering For Optimization in Internet Promotions Using Bayesian Analysis," *J. Appl. Quant. Methods*, vol. 5, no. 2, 2010.

[2] H. Wang, D. Can, A. Kazemzadeh, F. Bar, and S. Narayanan, "A System for Real-time Twitter Sentiment Analysis of 2012 U.S. Presidential Election Cycle," *Proc. 50th Annu. Meet. Assoc. Comput. Linguist.*, no. July, pp. 115–120, 2012.

[3] P. Goncalves, B. Fabrício, A. Matheus, and C. Meeyoung, "Comparing and Combining Sentiment Analysis Methods Categories and Subject Descriptors," *Proc. first ACM Conf. Online Soc. networks*, pp. 27–38, 2013.

[4] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," *Proc. Conf. Empir. Methods Nat. Lang. Process.*, pp. 79–86, 2002.

[5] S. J. M. Modha, Jalaj S. , Gayatri S. Pandi, "Automatic Sentiment Analysis for Unstructured Data," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 3, no. 12, pp. 91–97, 2013.

[6] M. Ahmad, S. Aftab, S. S. Muhammad, and U. Waheed, "Tools and Techniques for Lexicon Driven Sentiment Analysis : A Review," *Int. J. Multidiscip. Sci. Eng.*, vol. 8, no. 1, pp. 17–23, 2017.

[7] K. Mouthami, K. N. Devi, and V. M. Bhaskaran, "Sentiment analysis and classification based on textual reviews," *2013 Int. Conf. Inf. Commun. Embed. Syst.*, pp. 271–276, 2013.

[8] K. N. Devi and V. M. Bhaskarn, "Online forums hotspot prediction based on sentiment analysis," *J. Comput. Sci.*, vol. 8, no. 8, pp. 1219–1224, 2012.

[9] P. S. Earle, D. C. Bowden, and M. Guy, "Twitter earthquake detection: Earthquake monitoring in a social world," *Ann. Geophys.*, vol. 54, no. 6, pp. 708–715, 2011.

[10] M. Cheong and V. C. S. Lee, "A microblogging-based approach to terrorism informatics: Exploration and chronicling civilian sentiment and response to terrorism events via Twitter," *Inf. Syst. Front.*, vol. 13, no. 1, pp. 45–59, 2011.

[11] A. Go, R. Bhayani, and L. Huang, "Twitter Sentiment Classification using Distant Supervision," *Processing*, vol. 150, no. 12, pp. 1–6, 2009.

[12] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. Trends Inf. Retr.*, vol. 2, no. 1–2, pp. 1–135, 2008.

[13] A. Bifet and E. Frank, "Sentiment knowledge discovery in Twitter streaming data," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2010, vol. 6332 LNAI,

pp. 1–15.
[14] L. Breiman, “Random forests,” *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[15] P. K. Singh and M. Shahid Husain, “Methodological Study Of Opinion Mining And Sentiment Analysis Techniques,” *Int. J. Soft Comput.*, vol. 5, no. 1, pp. 11–21, 2014.
[16] J. Khairnar and M. Kinikar, “Machine Learning Algorithms for Opinion Mining and Sentiment Classification,” *Int. J. Sci. Res. Publ.*, vol. 3, no. 6, pp. 1–6, 2013.
[17] T. M. S. Akshi Kumar, A. Kumar, and T. M. Sebastian, “Sentiment Analysis on Twitter,” *IJCSI Int. J. Comput. Sci. Issues*, vol. 9, no. 4, pp. 372–378, 2012.
[18] A. Pak and P. Paroubek, “Twitter as a Corpus for Sentiment Analysis and Opinion Mining,” *Proc. Seventh Conf. Int. Lang. Resour. Eval.*, pp. 1320–1326, 2010.
[19] K. Nigam, J. Lafferty, and A. McCallum, “Using maximum entropy for text classification,” in *IJCAI-99 workshop on machine learning for information filtering*, 1999, vol. 1, pp. 61–67.
[20] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter, “Pegasos: Primal estimated sub-gradient solver for SVM,” *Math. Program.*, vol. 127, no. 1, pp. 3–30, 2011.
[21] H. Ahn, H. Moon, M. J. Fazzari, N. Lim, J. J. Chen, and R. L. Kodell, “Classification by ensembles from random partitions of high-dimensional data,” *Comput. Stat. Data Anal.*, vol. 51, no. 12, pp. 6166–6179, 2007.
[22] P. Bonissone, J. M. Cadenas, M. Carmen Garrido, and R. Andrés Díaz-Valladares, “A fuzzy random forest,” *Int. J. Approx. Reason.*, vol. 51, no. 7, pp. 729–747, 2010.
[23] A. Rashid, N. Anwer, M. Iqbal, and M. Sher, “A Survey Paper: Areas, Techniques and Challenges of Opinion Mining,” *Int. J. Comput. Sci. Issues*, vol. 10, no. 6, pp. 18–31, 2013.
[24] I. Smeureanu and B. Cristian, “Applying Supervised Opinion Mining Techniques on Online User Reviews,” *Inform. Econ.*, vol. 16, no. 2, pp. 81–92, 2012.
[25] C. Cortes and V. Vapnik, “Support vector machine,” *Mach. Learn.*, pp. 1303–1308, 1995.
[26] A. Abbasi, H. Chen, S. Thoms, and T. Fu, “Affect analysis of web forums and blogs using correlation ensembles,” *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 9, pp. 1168–1180, 2008.
[27] S. Argamon, C. Whitelaw, P. Chase, S. R. Hota, N. Garg, and L. Shlomo Levitan, “Stylistic text classification using functional lexical features,” *J. Am. Soc. Inf. Sci. Technol.*, vol. 58, no. 6, pp. 802–822, 2007.
[28] G. Taşkin Kaya, O. K. Ersoy, and M. E. Kamaşak, “Support vector selection and adaptation for remote sensing classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 6 PART 1, pp. 2071–2079, 2011.
[29] T. Wilson, J. Wiebe, and R. Hwa, “Recognizing strong and weak opinion clauses,” in *Computational Intelligence*, 2006, vol. 22, no. 2, pp. 73–99.
[30] D. K. Nirmala and P. Jayanthi, “SENTIMENT CLASSIFICATION USING Address for Correspondence,” *Int. J. Adv. Eng. Technol.*, pp. 1–3, 2016.
[31] V. B. Vaghela, B. M. Jadav, and M. E. Scholar, “Analysis of Various Sentiment Classification Techniques,” *Int. J. Comput. Appl.*, vol. 140, no. 3, pp. 975–8887, 2016.
[32] C. Chang and C. Lin, “LIBSVM : A Library for Support Vector Machines,” *ACM Trans. Intell. Syst. Technol.*, vol. 2, pp. 1–39, 2013.