



Pragmatic Analysis of Classification Techniques based on Hyper-parameter Tuning for Sentiment Analysis

Rohan Ahuja, Kunal Vats, Chirag Pahuja, Tanuj Ahuja and
Charu Gupta

EasyChair preprints are intended for rapid
dissemination of research results and are
integrated with the rest of EasyChair.

April 18, 2020

Pragmatic Analysis of Classification Techniques based on Hyper-parameter Tuning for Sentiment Analysis

Rohan Ahuja¹, Kunal Vats², Chirag Pahuja³, Tanuj Ahuja⁴, Charu Gupta^{5,*}
^[1,2,3,4,5]Computer Science and Engineering Department

Bhagwan Parshuram Institute of Technology, Delhi, India

ahujarohan180@gmail.com¹

kunalvats2000@gmail.com²

pahuja.chirag98@gmail.com³

tanuj11299@gmail.com⁴

charu.wa1987@gmail.com⁵

Abstract:

The evolution of technology and strong social network has empowered the online user community to share their views on almost every product, event or issue. This has led to a large amount of unstructured online user generated data. Furthermore, every company selling online products analyses its product's demand and also focuses on their corresponding user reviews. This online user data needs to be analysed for effective decision making either for the user or for the manufacturer. For this, Sentiment Analysis plays a vital role and is extremely useful in social media monitoring as it allows insight of the wider public opinion. In the present study, Amazon product review dataset is used to perform sentiment analysis. The proposed model is trained for four different classifiers: Naive Bayes, Support Vector Machine, Logistic Regression, and Random Forest, the model achieved a maximum accuracy of 91% using Logistic Regression. Furthermore, a comparative analysis of various algorithms is also discussed. The study focuses on the importance of hyper parameter tuning while training a classifier which helps in achieving better results than other previous approaches.

Keywords: Amazon Product Reviews, Classification Sentiment Analysis, Social Media, Hyper-parameter tuning.

I. Introduction

Sentiment analysis or opinion mining is a field of natural language processing which analyses the positive, negative or neutral sentiments (emotions) associated in text, speech or both. It extracts subjective information from the text corpus to provide valuable insights which provide the required decision-making rules to business. Sentiment mining is a significant research area as there is a significant increase in the user online data on E-trade sites

where understanding an individual's opinions is an important criterion. Around 90% of the users' information has been given during the most recent two years. Hence there is a dire need to carefully analyse this plethora of information. Although sentiment analysis is one of the most widely used techniques to find sentiment in the text, it has numerous challenges [9]. *Firstly*, online text on the internet consists of slang, abbreviations, typos, poor punctuation, poor grammar, which makes it difficult for the classifier to predict accurate results. *Secondly*, sarcasm in text data is a major problem in identifying the polarity of the statement. *Thirdly*, anaphora resolution which is the process of resolving the reference of a pronoun or a noun phrase in a sentence [3]. For example, "We went to play cricket and watched the movie, it was awful." What does "It" refer to here? This is a significant hurdle in the process of sentiment analysis. Furthermore, the ability to identify the correct interpretation of the context in which certain words used remains a challenge.

In this paper, an online user review analysis system (based on text only) is designed to create an easy to use environment which can be used by the companies/manufacturers to analyse the impact (good or bad) of the company's product in the market. The proposed methodology is experimented with four different classifiers, namely Naive Bayes, Support Vector Machine (SVM), K- Nearest Neighbors, and Random Forest (RF) [6] on the amazon earphone review data-set. The motivation behind the proposed methodology is to critically

examine various classifiers with hyper-parameter tuning for predicting the best result of finding the polarity of the text. The results are further compared with existing works in the literature.

The rest of the paper is organized in the following sections. Section II discusses the related work and section III explains the methodology of the proposed work. Section IV, V and VI illustrates the implementation, experimental results and comparative analysis respectively. Section VII discusses the conclusion of the proposed methodology with critical examination of the results and future work.

II. Literature Survey

Nowadays, every company wants to analyse how good their products are in the online market. May it be an online store or only an organization that wants to test its employee's satisfaction. Opinion mining and sentiment analysis have long been proposed as a technique used to solve this problem and became a field of interest for many researchers. Sentiment Analysis has been tackled at various levels of detail, including document-level classification in [1], sentence-level in [2], and phrase-level in [3]. In [4] the methodology used integrates existing sentiment analysis approaches, and increases the accuracy of the system. In [5], it is shown that support vector machines (SVM) perform better than Naive Bayes, which is in agreement with the proposed results. In [6], technique for opinion mining using R on plain text data from Twitter using a lexicon approach is proposed. *However, none of the above approaches have shown the importance of hyper-parameter tuning while training a classifier.* To understand the effect of hyper-parameter tuning, the proposed framework does a comparative analysis of the following classifiers: Naive Bayes, SVM, Logistic Regression and RF Algorithm. The motivation of the proposed study is to help and guide the decision-maker to choose the most appropriate classifier for a given dataset.

III. Proposed Methodology

A flowchart is a sort of framework that addresses a work procedure or method. A flowchart can similarly be described as a diagrammatic depiction of any process or method [4]. The flow graph in Figure 1 depicts the proposed methodology of the process of this paper.

A. Data Pre-processing & Dataset

Data pre-processing is mainly carried out to remove inconsistent, noisy, and incomplete data from the training set. It consists of different steps: tokenization, stop words removal, Stemming, Lemmatization [10].

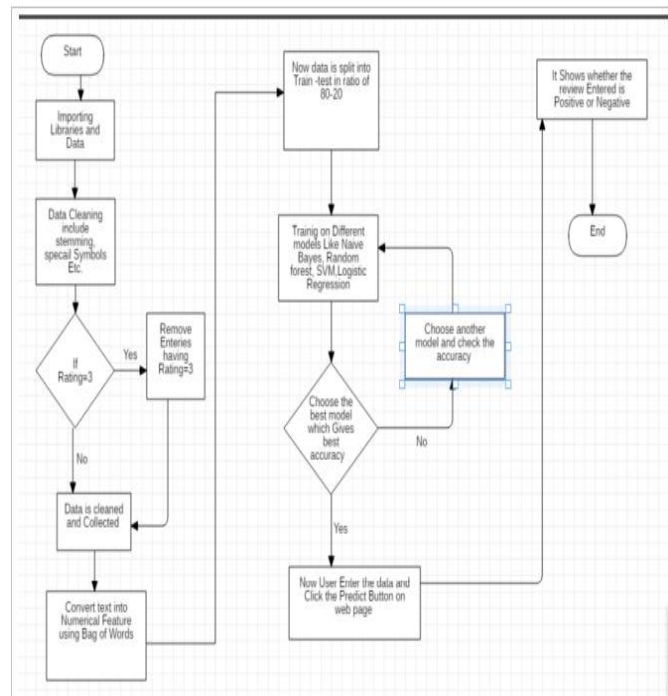


Figure 1: Flowchart of the Proposed System

Tokenization: It is the process of recognizing basic units inside a sentence which need not be disintegrated in subsequent processing. The resultant individual units after the process of tokenization are

known tokens. These tokens are input to the next step(s) in the pre-processing stage.

Stop words removal: The majority of words in a sentence or a paragraph are connecting words which do not contribute much towards the polarity. In this process, these unnecessary words from the text are removed. According to the proposed framework, this step is not optional. In absence of stop-word removal, the feature space might get too large, which can significantly affect the performance of the algorithm(s).

Stemming: In this process, the characters in a word are removed which reduces the word to its root. In the proposed work, Porter stemming is used to perform this task. It works by removing the everyday person's morphological and inflexional endings from words in English.

Lemmatization: The objective of lemmatization is equivalent to Stemming. It reduces inflectional structures and derivationally related types of a word for a typical base structure [3]. It takes into consideration the meaning of the word rather than stemming, which aims to reduce the characters in the word.

Hyper-parameter Tuning

Hyper-parameters are the values which are used in machine learning algorithms and whose values are set before the learning process begins. Tuning of the hyper-parameters means finding out the best suited values for each algorithm which would work best. Every algorithm has its different hyper-parameters to be tuned. The respective hyper-parameters for each algorithm are shown in the Table 1.

Table 1: Hyper-parameter tuned with respect to various classification algorithms

S.N	Algorithm	Formula	Parameters
1	Logistic	$V_i = \beta_0 +$	$C=0.7(For$

	Regression	$\beta_1 Y_1 + \epsilon_1$	Hyperparameter tuning), Other as default.
2	Naive Bayes	$P(c t) = P(c) \cdot P(t c)$	var_smoothing = 10^{-12} Other as default.
3	k-NN	$S(d, c_i) = \sum \text{sim}(d, d_j) \delta(d_j, c_i)$	n_neighbors = 10 Other as default.
4	Random Forest	$G(t) = 1 - \sum p^2(k t)$	All as default.

Data Set Features:

ReviewTitle : Title of the Review
 ReviewBody : Body of the Review
 ReviewStar : Stars given by Customer ProductProduct: Name

IV. Design and Implementation

The ideology of the proposed work is to understand the usage of online product reviews taken from a well-known dataset repository (Kaggle). The steps in the proposed methodology are as follows:

- Step 1 : Data collection of reviews for products
- Step 2 . Data cleaning like stop-words removal.
 - Step 3.1: Tokenize each review.
 - Step 3.2: Lemmatize each word.
- Step 3 : Converting text to numerical features using Bag-of-Words.
- Step 4 : Splitting data into train and test data.
- Step 5 : Analysing different algorithms.
 - Step 5.1 : Apply different machine learning algorithms on the cleaned text and analyse the accuracy of the respective model.
 - Step 5.2 : Hyper-parameter tuning for the algorithm with the best accuracy on the given dataset.

As it can be seen from the step 1 to step 5, the proposed model fetches the data, performs cleaning

or remove stop words, classifies reviews, and gets the polarity of the reviews.

Further, almost all machine learning methods can be used to the task of classifying texts. Most often used and well-proven SVM, Bayes Method, Nearest Neighbor Method, Neural networks, Decision trees, Rocher classifier. However, the proposed work develops an appropriate method for the Classification of online user review text using four classic algorithms: SVM, Logistic Regression, Naïve Bayes, and RF. These algorithms are easy to understand and used widely in the literature.

V. Experimental Results

In order to implement the aforementioned steps, Python is used for sentiment analysis. The packages, thus, utilized includes CGI, counter, accuracy_score, model_selection, nltk, stopwords, WordNetLemmatizer, train_test_split, RandomizedSearchCV, Logistic Regression.

The experimental results show that SVM and Logistic Regression have better average performance than RF and Naïve Bayes. Initially, Logistic Regression reached 89% using combinations of representative design with prior processing tokenization, filtering, normalization, and root stemmer. TF-IDF is used as a representation of characteristics with/without a selection of any particular feature. SVM reached 81.00% using a combination of tokenization, filtering in as pre-processing, and TF-IDF as a representation indicator with information gain as a choice of the indicator. Further, it is observed that easy stemming is the best cutting technique. This is because easy stemming is better than stemming from linguistics. From the semantic point of view it takes the least time for pre-processing and has the excellent average classification accuracy. Also it is observed that the development of indicators (hyper-parameters) is very important for improving the accuracy of the classification.

Table 2: Confusion matrix of Logistic prediction effect

		Actual Value		
		1 (positive class)	0 (negative class)	Merge
Predictive value	1 (positive class)	29 (TP)	5 (FP)	34
	0 (negative class)	4 (FN)	62 (TN)	66
	Merge	33	67	100

The confusion matrix of Logistic Regression is shown in Table 2 and that of SVM prediction is shown in Table 4. Also, the Precision - recall of Logistic Regression is shown in Table 3 and that of SVM is depicted in Table 5.

Table 3: Precision - recall with F1-score matrix of Logistic model used

	Precision	Recall	F1-score	Support
0	0.60	0.62	0.60	34
1	0.925	0.939	0.932	66
Accuracy			0.89285	100
Macro avg	0.77	0.78	0.77	100
Weighted avg	0.89	0.89	0.89	100

Table 4: Confusion matrix of SVM prediction effect

		Actual Value		
		1 (positive class)	0 (negative class)	Merge
Predictive value	1 (positive class)	261 (TP)	122 (FP)	383
	0 (negative class)	146 (FN)	871 (TN)	1017
	Merge	407	993	1400

Table 5: Precision - recall with F1-score matrix of SVM model

	Precision	Recall	F1-score	support
0	0.64	0.68	0.66	383
1	0.88	0.86	0.87	1017
Accuracy			0.81	1400
Macro avg	0.76	0.77	0.76	1400
Weighted avg	0.81	0.81	0.81	1400

Table 6: Comparative Analysis of Accuracy in Various Learning Models (Product Reviews)

Model	Accuracy
Naive Bayes	70.5%

Random Forest	78.5%
SVM	83%
Logistic Regression	91%

VI. Comparative Analysis

In this section, the proposed methodology is compared with the existing work in the literature. A comparative analysis of this study is shown in Table 6 which examines the proposed methodology with other similar works in literature [12,13,14].

Table 7: Comparison with other similar works

Paper Title	Dataset	Accuracy	
Amazon Reviews, business analytics with sentiment analysis [12]	Amazon Product Reviews (Mobile Reviews)	MNB	72.95%
		SVM	80.11
Feature Selection Methods in Sentiment Analysis and Sentiment Classification of Amazon Product Reviews [13]	Amazon data for Books	Phrase Level	70.00%
		Single Word	70.00%
		Multi Word	80.00%
	Amazon data for Music	Phrase Level	62.00%
		Single Word	80.00%
		Multi Word	68.00%
	Amazon data for Camera	Phrase Level	62.00%
		Single Word	80.00%
		Multi Word	68.00%
Sentiment Analysis in Amazon Reviews Using Probabilistic Machine Learning [14]	Data of reviews of books	84.44%	
	Data of reviews of Kindle	87.33%	
Proposed Model	Amazon	Naive Bayes	70.5%

Earphone Review Dataset	Random Forest	78.5%
	SVM	83%
	Logistic Regression	91%

From the experimental results obtained, it is observed that Logistic regression outperforms other algorithms in classifying the reviews with an accuracy of 91%. This study can be further utilised to understand the effect of parameters and hyper parameters used in various classification algorithms. The proposed methodology can be studied with soft computing techniques as well.

From Table 7, the approaches dealt do not explicitly concentrate on the values of hyper-parameters during the process of training however in the proposed work, Hyper-parameter tuning on the Logistic Regression model gave the best accuracy when sample models were trained.

The empirical analysis suggests that these parameters play a vital role in improving the resulting accuracy. This is because tuning hyper-parameters helps in getting rid of under-fitting and over-fitting of the model. Hyper-parameter tuning helps in reducing loss factor through a great margin as the parameters are fine tuned in correspondence to the training data. *For example* - In Logistic regression, to get the right classifying plane it is really important to get the appropriate weights associated with each of the features. This can be easily tested by tuning the hyper-parameters which is true for other algorithms also.

VII. Conclusion and Future Work

With the increased interest of people in online shopping, tweeting, writing opinions, there is a need to analyse these opinions that contain a large amount of decision making information. This information is useful for both customers as well as for the manufacturer. With the proposed methodology, these opinions are analysed using various classification algorithms. Also, the importance of product reviews is analysed. The classification of the reviews is discussed with an emphasis on the importance of hyper-parameter tuning. Through empirical testing it is observed that hyper-parameter tuning is of great significance and can improve the accuracy of any classification algorithm drastically.

References

- [1] Pang, B. and Lee, L., 2004, July. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd annual meeting on Association for Computational Linguistics* (p. 271). Association for Computational Linguistics.
- [2] Liu, B., 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1), pp.1-167.
- [3] Nguyen, H., Veluchamy, A., Diop, M. and Iqbal, R., 2018. Comparative Study of Sentiment Analysis with Product Reviews Using Machine Learning and Lexicon-Based Approaches. *SMU Data Science Review*, 1(4), p.7.
- [4] Bhatt, A., Patel, A., Chheda, H. and Gawande, K., 2015. Amazon review classification and sentiment analysis. *International Journal of Computer Science and Information Technologies*, 6(6), pp.5107-5110.
- [5] Shivaprasad, T.K. and Shetty, J., 2017, March. Sentiment analysis of product reviews: a review. In *2017 International Conference on Inventive Communication and Computational Technologies (ICICCT)* (pp. 298-301). IEEE.
- [6] Ray, P. and Chakrabarti, A., 2017, February. Twitter sentiment analysis for product review using lexicon method. In *2017 International Conference on Data Management, Analytics and Innovation (ICDMAI)* (pp. 211-216). IEEE.
- [7] onnx.ai , <https://onnx.ai/>[Last Accessed on:26-12-2019]
- [8] es.scribd.com, <https://www.scribd.com/>[Last Accessed on :23/12/2019]
- [9] Gautam, A., Bhateja, V., Tiwari, A. and Satapathy, S.C., 2018. An improved mammogram classification approach using

back propagation neural network. In *Data Engineering and Intelligent Computing* (pp. 369-376). Springer, Singapore.

[10] Nandal, N., Tanwar, R. and Pruthi, J., 2020. Machine learning based aspect level sentiment analysis for Amazon products. *Spatial Information Research*, pp.1-7.

[12] Elli, M.S. and Wang, Y.F., 2016. Amazon Reviews, business analytics with sentiment analysis.

[13] Shaikh, T. and Deshpande, D., 2016. Feature selection methods in sentiment analysis and sentiment classification of amazon product reviews. *Int J Comput Trends Technol*, 36(4), pp.225-230.

[14] Rain, C., 2013. Sentiment analysis in amazon reviews using probabilistic machine learning. *Swarthmore College*.