

Target Sentiment Analysis Ensemble for Product Review Classification

Rhoda Viviane Achieng Ogutu, Jomo Kenyatta University of Agriculture and Technology, Kenya*

Richard M. Rimiru, Jomo Kenyatta University of Agriculture and Technology, Kenya

Calvins Otieno, Maseno University, Kenya

ABSTRACT

Machine learning can be used to provide systems with the ability to automatically learn and improve from experiences without being explicitly programmed. It is fundamentally a multidisciplinary field that draws on results from artificial intelligence, probability and statistics, information theory and analysis, among other fields that impact the field of machine learning. Ensemble methods are techniques that can be used to improve the predictive ability of a machine learning model. An ensemble comprises of individually trained classifiers whose predictions are combined when classifying instances. Some of the currently popular ensemble methods include boosting, bagging, and stacking. In this paper, the authors review these methods and demonstrate why ensembles can often perform better than single models. Additionally, some new experiments are presented to demonstrate the computational ability of the stacking approach.

KEYWORDS

Artificial Intelligence, Data Science, Ensemble Learning, Machine Learning, Naïve Bayes, Sentiment Analysis, Sentiment Classification, Supervised Learning, SVM, Unsupervised Learning

INTRODUCTION

There are various methods of constructing ensembles with different classifiers for data science and analysis. However, just as Wolpert and Macready (1997) puts it, different learning methods can be developed to solve different classification problems. Generally speaking, according to Seijo-Pardo, Porto-Díaz, Bolón-Canedo, and Alonso-Betanzos (2017) there are two kinds of ensembles, homogeneous ensembles and heterogeneous ensembles. Homogeneous ensembles are ensembles where all classifiers are of same type or family, while heterogeneous ensembles are ensembles where the classifiers are of different type or family, that is, diverse (Abuassba, Zhang, Luo, Shaheryar, & Ali, 2017).

Rokach (2005) explains that there are several factors that can be used to differentiate ensemble methods in different dimensions. These include factors to do with Inter-classifier relationship, classifier combining methods, classifier diversity generator and Ensemble size. However, classification can be based on two main dimensions, Inter-classifier relationship and Combining Methods (Araque, Corcuera-Platas, Sánchez-Rada, & Iglesias, 2017). Inter-classifier relationship

DOI: 10.4018/JITR.299382

*Corresponding Author

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

refer to the methods of achieving learning process, which include Sequential and Concurrent methods (Araque et al., 2017). In Sequential approaches, there is an interaction between the learning runs and it is possible to take advantage of knowledge generated in previous iterations to guide the learning in the next iterations. This approach includes techniques such as Model-guided Instance Selection where the classifiers that were constructed in previous iterations are used for manipulating the training set for the next iteration. Such model-guided algorithms include Boosting, Uncertainty Sampling among others. Sequential approaches also includes Incremental Batch Learning, where the classifier produced in one iteration is given as “prior knowledge” to the learning algorithm in the following iteration (along with the subsample of that iteration). The learning algorithm uses the current subsample to evaluate the former classifier, and uses the former one for building the next classifier. The classifier constructed at the last iteration is chosen as the final classifier (Rokach, 2005). On the other hand, in the Concurrent ensemble approaches the original dataset is partitioned into several subsets from which multiple classifiers are induced concurrently. The subsets may be disjoint (mutually exclusive) or overlapping after which a combining procedure is applied in order to produce a single classification for a given instance. Since the method for combining the results of induced classifiers is usually independent of the induction algorithms, it can be used with different inducers at each subset. Concurrent methods aim either at improving the predictive power of classifiers or decreasing the total execution time. Algorithms that can be implemented in this approach include Bagging, Cross-validated Committees among others (Rokach, 2005).

The second dimension Combining Methods include simple multiple classifier combination and Meta combining methods. The simple combining techniques are best suited for classification problems where the individual classifiers perform the same task and have comparable success. However, these combiners are more vulnerable to outliers and to unevenly performing classifiers. The simple combining techniques include techniques such as Uniform Voting where each classifier has the same weight, while a classification of an unlabelled instance is performed according to the class that obtains the highest number of votes (Rokach, 2005). In addition, in majority voting every classifier makes a prediction (votes) for each test instance and the final output prediction is the one that receives more than half of the votes (majority). If none of the predictions get more than half of the votes, it can be assumed that the ensemble method could not make a stable prediction for this instance (Demir, 2015). On the other hand, the Meta learning technique means that learning is from the classifiers produced by the base learners (inducers) and from the classifications of these classifiers on training data. This includes techniques such as Stacking and Arbiter Trees (Rokach, 2005).

Rocca (2019) identifies three common ways of creating ensembles from these classifiers or learners. In bagging approach, the original dataset is partitioned into several subsets from which multiple classifiers are concurrently induced. The subsets may be disjoint (mutually exclusive) or overlapping. A combining procedure is then applied in some kind of deterministic averaging process. Bagging approach can be used to decrease the total execution time for classifiers (Rokach, 2005). The second one is boosting, which is also a method for improving the performance of any learning algorithm (Rokach, 2005). The approach learns the weak learners sequentially in a very adaptive way and combines them following a deterministic strategy (Rocca, 2019). According to Rokach (2005) sequential learning can be achieved by repeatedly running a weak learner, on various distributed training data. The classifiers produced by the weak learners are then combined into a single composite strong classifier in order to achieve a higher accuracy. Finally, the third approach is stacking, where meta-learning is done by learning from the classifiers produced by the inducers, and from the classifications of these classifiers on training data. This method tries to induce which classifiers are reliable and which are not. Stacking is usually employed to combine models built by different inducers (Rokach, 2010).

Supervised vs. Unsupervised Machine Learning Approaches

According to Mitchell (1998) Machine Learning is a field concerned with the question of how to construct computer programs that automatically improve with experience, that is, Machine Learning explores the study and construction of algorithms that can learn from and make predictions on data.

Zhou (2012) state that the purpose of Machine Learning is to learn from training data so as to make predictions on new or unseen data, this is what is referred to as 'Generalization'. There exist different learning settings, among which the most common ones are supervised learning and unsupervised learning (Zhou, 2012).

Supervised Learning is a Machine Learning approach where all data is labelled and the algorithms learn to predict the output from the input data or the training dataset. Specific output values are supplied, and the algorithm iteratively makes predictions on the training data, while learning is improved with the experience learned. Learning stops when the algorithm achieves an acceptable level of performance (Hastie, Tibshirani, Friedman, 2009).

According to Korjus, Hebart and Vicente (2016) and Alpaydin (2014) Supervised Machine learning require splitting of the dataset into a training set and a test set. A training set is the actual dataset or a sample of the dataset used to train (fit) a model. The model learns from the training set. A test set on the other hand, is the sample of the dataset used to provide an unbiased evaluation of a final model trained on the training dataset. It provides the ultimate standard used to evaluate the model. Finally, the goal of Supervised Machine Learning for a classification problem is to find a model that accurately classifies and generalizes, such as applicable in Text classification. To test the generality of a learned model, that model is typically applied to the test dataset and the prediction outcome then informs a researcher about the performance of the model. This therefore informs the understanding that the learning process in a simple machine learning model is divided into two steps; training and testing (Nasteski, 2017).

Unsupervised Learning on the hand is a Machine Learning approach where algorithms are trained to use data that is not labelled. This means that no training data is provided and the algorithm is made to learn by itself while classifying the data without any prior experience (Sethi, 2020). The goal for unsupervised learning is to model the underlying structure in order to learn more about the data, provide insights that were previously unknown and to identify hidden patterns (Brownlee, 2019). As such, there aren't necessarily defined outcomes from unsupervised learning algorithms. Rather, this type of learning determines what is different or interesting from the given dataset. Unsupervised Learning algorithms can therefore be practically applicable in fraud detection and identification of human errors during data entry among other applicable areas (Sethi, 2020).

The goal of the researchers in this study was to optimize performance criteria using experience. For this reason, the Supervised Learning algorithms applied allowed the researchers to collect data, label the data and produce a data output from the previous experiences.

According to Jurafsky and Martin (2017), one of the oldest tasks in Text classification is assigning a library subject category or topic label to a text. The researchers study reveal that subject category classification is the task for which Naïve Bayes algorithm was invented. The researchers state that the goal of classification is to take a single observation, extract some useful features, and thereby classify the observation into one of a set of discrete classes. There are other various methods of classifying texts such as the application of Rule-based classifiers, which, according to the researchers prove to be fragile, as situations or data change overtime while for some tasks humans aren't necessarily good at coming up with the rules. For this reason, most cases of classification in language processing are done by Supervised Machine Learning. However, Decision Tree classifiers training in Supervised Learning on the other hand, are relatively expensive as complexity and time taken to train the model is more (Dhiraj, 2019).

This study's researchers therefore implemented Naïve Bayes algorithm for sentiment detection, while experiments such as 'Bag-of-words' and n-gram were carried out for subjective analysis of the content. Naïve Bayes has also been shown by Huang, Lu and Ling (2003) to be superior in terms

of CPU and memory utilization. In addition, the algorithm is a probabilistic classifier and is used to indicate the probability of observation being in the class, by calculating the conditional probability of each attribute occurring in the predicted classes. The algorithm also assumes attribute independence, meaning, each individual feature is assumed to be an indication of assigned class, independent of each.

Support Vector Machine (SVM) on the other hand, is a Supervised Machine Learning algorithm which is also widely used for classification problems, even though it can be employed for both classification and regression purposes. It is a non-probabilistic binary linear classifier that has the ability to linearly separate classes by a large margin (Osisanwo, Akinsola, Awodele, Hinmikaiye, Olakanmi & Akinjobi, 2017). According to Balahur and Perea-Ortega (2015), SVM has been proven to be highly effective in traditional text classification and has been applied successfully in many opinion mining tasks, performing better than other machine learning techniques. The rationale behind SVMs for the purpose of this study, can be best explained by considering a set of data points that belong to one of two classes, '*positive*' and '*negative*', as illustrated in Figure 1. The two classifying algorithms, Naïve Bayes and Support Vector Machine (SVM), were then used to build the Ensemble model.

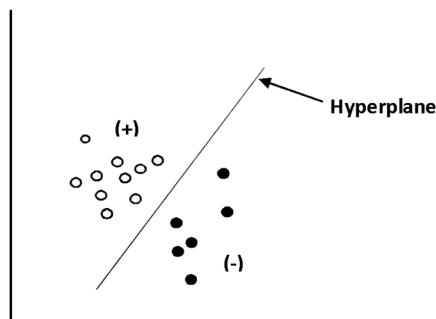
RELATED WORK

Application of Ensemble Learning

Ensemble methods can be used for the creation and building of a more conventional model. For instance, Ensemble methods can also be used to evaluate the relationship and responses in conventional statistical models. As such, Akhtar, Gupta, Ekbali, and Bhattacharyya (2017) presented a cascaded framework of feature selection and classifier ensemble using Particle Swarm Optimization (PSO) for aspect based Sentiment Analysis, where aspect term extraction and sentiment classification was carried out. The researchers used features that were identified based on the properties of different classifiers and domains. Three classifiers were used as base learning algorithms. The eligible classifiers identified were combined using either majority or weighted voting. In this study, the ensemble learner was used to find out the most eligible models, that when combined together, maximized some classification quality measure such as F-measure or accuracy.

Additionally, Ensemble learning has also been applicable in areas such as deep learning techniques for Sentiment Analysis to provide automatic feature extraction, rich in both representation capabilities and better performance as compared to traditional feature based techniques in social applications (Araque et al., 2017). The focus of deep learning techniques is to learn complex features extracted from data with minimum external contribution (Bengio, 2009) using deep neural networks (Alpaydin, 2014). Araque et al. (2017) confirmed that the performance of their proposed ensemble models surpassed that of their original baseline models on F1-Score.

Figure 1. Linearly Separable Dataset



In addition, Ensemble learners have also been used to minimize error rate of models in Sentiment Analysis. For instance, Alnashwan, O’Riordan, Sorensen and Hoare (2016) employed a four base learners to investigate the effectiveness of using a combination of existing lexicon resources as meta-level features in ensemble learning for sentiment classification. This offered advantages over using either a single lexicon resource or a single classifier. Moreover, their experiment showed that, based on a combination of existing lexicon resources, the ensemble learners minimized the error rate by avoiding poor selection from stand-alone classifiers.

Fersini, Messina, and Pozzi (2014), studied Ensemble Learning to reduce the noise sensitivity related to language ambiguity and therefore to provide a more accurate prediction of polarity. The researchers asserted that most of the existing approaches in Sentiment Analysis select the best classification model leading to over-confident decisions that do not take into account the inherent uncertainty of the natural language. They addressed the classifier selection problem by proposing a greedy approach that evaluated the contribution of each model with respect to the ensemble.

Boosting and Bagging Ensemble Methods

Boosting is a sequential ensemble approach, that works by repeatedly running weak learners on the training data (Viola & Jones, 2004). Schapire (1990) in his paper described weak learning as a situation where the concept class is weakly learnable if the learner can produce a hypothesis that performs only slightly better than random guessing. The classifiers that were constructed in previous iterations in boosting are then used for manipulating the training set for the next iteration. This makes it possible to take advantage of knowledge generated in previous iterations to guide the learning in the next iterations. Eventually, the classifiers produced by weak learners are combined into a single composite better performing classifier (Rokach, 2005) and (Viola & Jones, 2004).

A weaker model of learnability drops the requirement that the learner be able to achieve arbitrarily high accuracy. Algorithms such as boosting approach can therefore be used to improve the performance of a weak learner, that is, boost the low accuracy of a weak learning algorithm (Schapire, 1990). Due to the reason that it generates a final classifier whose error on the training set is small by combining many hypotheses whose error may be large (Rokach, 2005). Furthermore, in boosting, since base classifiers are influenced by the performance of those built previously, the new classifier pays more attention to error that were made in the previous ones and to their performances (Viola & Jones, 2004).

Bagging approach main goal is to improve accuracy by creating an improved composite classifier, by a means of integrating the various outputs of learned classifiers into a single prediction (Rokach, 2005). In comparison to bagging ensemble, which require that learning systems be unstable (Quinlan, 1996) and (Saugata, 2018) boosting however, does not require the use of unstable learning systems, provided that their error rate is kept below 50% (Rokach, 2005; Quinlan, 1996). Unstable classifiers are characterized by high Variance (Breiman, 1998).

Both bagging and boosting reduce bias to some extent, however, their major contribution to accuracy is in the large reduction of variance. As such, boosting performs better than bagging because it does better at variance reduction (Breiman, 1998).

A major drawback in boosting is over fitting which may lead to deterioration in performance. This can be brought about by having many iterations (Quinlan, 1996). A possible way to avoid this issue of over fitting is by keeping the number of iterations as small as possible (Rokach, 2005).

Fundamental Contributions of Ensemble Learning

An ensemble learning is a Machine Learning process to get better prediction performance by strategically combining the predictions from multiple learning algorithms (Abuassba et al., 2017; Tuwe, 2015). Further, Dietterich (2000) states that an ensemble of classifiers is a set of classifiers whose individual decisions are combined in some way to classify new examples. The study’s previous discussions have demonstrated that ensembles are usually significantly more accurate and successful

than single learners, as the base classifiers are trained to solve the same original problem but combined to get better results (Araque et al., 2017).

In addition, Dietterich (2000) elaborates that there are three fundamental reasons why ensemble methods outperform single classifiers. The first reason is statistical reason, where a learning algorithm or classifier can be viewed as searching a space H of hypotheses to identify the best hypothesis in the space. The statistical problem arises when the amount of training data available is too small compared to the size of the hypothesis space. Without sufficient data, the learning algorithm can find many different hypotheses in H that all give the same accuracy on the training data. By constructing an ensemble out of all of these accurate classifiers the algorithm can ‘average’ their votes and reduce the risk of choosing the wrong classifier. The second reason is a computational reason where the researcher identified that many learning algorithms or classifiers work by performing some form of local search that may get stuck in local optima. In instances where the statistical problem is absent, due to availability of enough training data, it may still be difficult computationally for the learning classifier to find the best hypothesis. An ensemble constructed by running the local search from many different starting points may provide a better approximation to the true unknown function than any of the individual classifiers. Finally, the third reason is representational, which refers to the functions which the model can learn. In most applications of machine learning, the true function f cannot be represented by any of the hypotheses in H . By forming weighted sums of hypotheses drawn from H , it may be possible to expand the space of representable functions. This is a somewhat subtle issue because there are many learning algorithms for which H is, in principle, the space of all possible classifiers (Dietterich, 2000).

Often times, there are various important factors to consider when creating ensembles such as biasness or variance of base learners (Rocca, 2019). However, Hansen and Salamon (1990) and Dietterich (2000) points out that accuracy and diversity are also some of the key metrics used when developing ensembles. With this, the researchers note that a necessary and sufficient condition for an ensemble of classifiers to perform better than its individual members, is if the individual classifiers are *accurate* and make independent errors (*diverse*) that is, the errors made by the classifiers are uncorrelated (Dietterich, 2000) and (Rokach, 2010). Diversity is important while creating ensemble models because it allows stronger learners from different regions to combine their ability to reduce risk of misclassification. While selecting and creating the predictive model in the study, classification accuracy - the proportion of correctly classified instances, was used as one of the metric. An accurate classifier is one that has an error rate of better than random guessing on new x values. On the other hand, the diversity of the members of an ensemble is known to be an important factor in determining its generalization error (Melville, 2003; Rokach, 2010; Moudřík & Neruda, 2015).

ENSEMBLE OF CLASSIFIERS

Sentiment Analysis Models

This research therefore presented the Sentiment analysis models proposed in the previous work (Ogutut et al., 2019). The models were validated with data sets of tweet reviews of “OnePlus 7 Pro” mobile phone product, in the first round of experiments where the data was categorized as *positive* and *negative*. As mentioned earlier, building a diverse set of initial models was essential for this study. Initial experiments in Ogutut et al. (2019) demonstrated the accuracy and diversity of Naïve Bayes and SVM classifiers, as used in the classification of twitter product review datasets. This can be seen in Tables 1 and 2.

As shown in Table 1 the average performances for Naïve Bayes and SVM in the initial experiments carried out by Ogutut et al. (2019) formed the basis for the intension to build an ensemble model. In addition, results in Table 2 also demonstrated that the errors made by the classifiers were uncorrelated. These outcomes indicated that the classifiers were strong, yet very diverse, a condition that supported

Table 1. Comparative Results for Classifier Accuracy

Sr. No.	No. of Reviews	Accuracy	
		Naïve Bayes	SVM
1	500	89.1%	90.8%
2	700	94%	95.7%
3	1000	95.8%	93.5%
4	1500	97.9%	95.8%
5	2000	98.4%	97.5%
6	3000	93.8%	97.6%
	Average Performance	94.8%	95.2%

Table 2. Comparative Results for Classifier Error Rates

Sr. No.	No. of Reviews	Error Rate (<i>err</i>)	
		Naïve Bayes	SVM
1	500	10.9%	9.2%
2	700	6%	4.3%
3	1000	4.2%	6.5%
4	1500	2.1%	4.2%
5	2000	1.6%	2.5%
6	3000	6.2%	2.5%

the building of the predictive ensemble model. A class of concepts is strongly learnable if there exist a polynomial-time algorithm that achieves low error with high confidence for all concepts in the class (Schapire, 1990).

With this in mind, the main assumption was that if Naïve Bayes and SVM classifiers were adequately and correctly combined, an even more accurate and robust model could be obtained from the previous study in Ogutu et al. (2019) with an intension of improving predictions.

Meta Classifier Model

With a Meta classifier technique, the outputs of the base classifiers are treated as inputs for the meta-learning model. In this approach, the model can learn and adapt to different situations (Araque et al., 2017).

Stacking Ensemble

Stacking is an ensemble approach where predictions by each different model is given as input for a Meta level classifier whose output is the final class (Wolpert, 1992). Stacking is usually employed to combine models built by different inducers, that is, heterogeneous models (Rokach, 2010) and is equally regarded as a technique for achieving the highest generalization accuracy for strong learners (Wolpert, 1992). Using a meta-learner, the method tries to induce which classifiers are reliable vis-a-vis the ones that are not. Most importantly, Stacking actively seeks to improve the performance of an ensemble as the Meta classifier is trained to learn and correct the errors of the base classifiers (Breiman, 1996; Wolpert, 1992).

This study focused on how the two heterogeneous classifiers, Naïve Bayes and SVM could be improved by combining them to get an even better performance. Stacking has been used by various researchers to achieve greater predictive accuracy for strong learners (Džeroski & Ženko, 2004; Moudřík & Neruda, 2015). As depicted by experimental results in Table 1 and 2, the study uses the two strong yet diverse base learners which are then aggregated by a Stacking ensemble.

Mathematically, if an ensemble has H base classifiers with an error rate of $e < \frac{1}{2}$, while at the same time, the base classifiers' errors are uncorrelated, then the probability that the ensemble will make an error is the probability that more than $\frac{H}{2}$ base classifiers misclassify the example. Simply, if an input-output pair (x, y) is left out of the training set of h_i , after training is complete for h_i , the output y can still be used to evaluate the classifier's error. Since (x, y) was not in the training set of h_i , $h_i(x)$ may differ from the desired output y . A new classifier is then trained to estimate this discrepancy, given by $y - h_i(x)$. Essentially, a second classifier is trained to learn the error the first classifier made. Adding the estimated errors to the outputs of the first classifier can improve generalization (Oza & Tumer, 2008).

Binary classification model was then fit with caretEnsemble package in R-Studio. caretEnsemble has three primary functions: caretList, caretEnsemble and caretStack. caretList is used to build lists of caret models on the same training data, with the same re-sampling parameters. caretEnsemble and caretStack are used to create ensemble models from such lists of caret models. caretEnsemble uses a Generalized Linear Models (GLMs) to create a simple linear blend of binary classification models, while caretStack uses a caret model to combine the predicted outputs from several component caret models (Mayer, 2019). Using the caretList function, the base caret models Naïve Bayes "nb" and SVM "svmLinear" was fit to the same datasets. The function returned a list of objects which were then passed to caretEnsemble and caretStack.

EVALUATION

The evaluation results from this second round of experiments were based on two commonly used performance metrics, accuracy and error rate (Kalaivani & Shunmuganathan, 2013). This was in accordance with the goal of the study which laid emphasis on accuracy measures, precision and robustness. Robustness is the ability of a model to cope with errors (Rodriguez, 2019).

The Confusion Matrix

The study was a binary classification problem in which sentiments were classified as either *positive* or *negative*. According to Hossin and Sulaiman (2015) the discrimination evaluation of the optimal solution during a classification training can be defined based on confusion matrix as in the Table 3. The confusion matrix is used for determining the correctness and accuracy of the model. The rows of the confusion matrix table represents the predicted class, while the column represents the actual class.

RESULTS AND DISCUSSION

For the purposes of carrying out the second round of experiments, collection and mining of tweet reviews of "OnePlus 7 Pro" mobile phone dataset was again done. The goal this time was to experiment on the accuracies and the error rates of the base models, so as to establish the impact and effectiveness of building the ensemble model.

Table 3. The Confusion Matrix

	Actual Negative Class	Actual Positive Class
Predicted Negative Class	True Negative (TN)	False Negative (FN)
Predicted Positive Class	False Positive (FP)	True Positive (TP)

True Negatives (TN): The cases which were predicted Negative and the actual output was also negative.

True Positive (TP): The cases which were predicted Positive and the actual class of the data point was also positive.

False Negative (FN): False Negatives were the cases in which the actual class of the data point was positive and the predicted was Negative. False was because the model had predicted incorrectly and negative because the class predicted was a negative.

False Positive (FP): False Positives were the cases in which the actual class of the data point was Negative and the predicted was Positive. False was because the model had predicted incorrectly and positive because the class predicted was positive.

Naïve Bayes, SVM and the Stacked Ensemble Classifiers Comparative Results

See Table 4.

Confusion Matrix Statistics and Model Inspection

In the analysis of the two classification approaches, the results indicated that in terms of error rate, Naïve Bayes seemed to be generating a significant amount of errors during computation as compared to SVM. As indicated in Table 4, SVM performances were better than Naïve Bayes in terms of accuracies and error rate measure. However, the stacked ensemble model's performances were similar to that of SVM performances, except for experiments carried out for 700 Reviews where the ensemble had a significant amount of errors at 3.9%. A 10-fold cross validation was also carried out during classifier training with the intension of tuning and developing a 'best' model; a model with minimum amount of error rate.

CONCLUSION

The goal of the final phase of experiments was to evaluate the performance for Sentiment classification in terms of accuracy, and robustness. We therefore compared the two supervised Machine Learning algorithms, for sentiment classification of Twitter product reviews. From the experiments, as much as there existed significant amount of errors, Naïve Bayes could not be regarded as an unstable learning system, since the error rate had been kept below 50%. Arguably, the experiments indicated that SVM and Naïve Bayes were good classifiers for Sentiment analysis and Text Classification due to the

Table 4. Comparative Results for Sentiment Classification and Classifier Performance – with Caret Package

Sr. No.	No. of Reviews	Accuracy			Sr. No.	No. of Reviews	Error Rate (err)		
		Naïve Bayes	SVM	Stacked Ensemble			Naïve Bayes	SVM	Stacked Ensemble
1.	500	91.6%	97.5%	97.5%	1.	500	8.3%	2.5%	2.5%
2.	700	81.1%	96.9%	96.1%	2.	700	18.9%	3.1%	3.9%
3.	1000	80.9%	96.6%	96.6%	3.	1000	19.0%	3.4%	3.4%
4.	1500	87.4%	98.1%	98.1%	4.	1500	12.6%	1.9%	1.9%
5.	2000	69.4%	97.4%	97.4%	5.	2000	30.6%	2.6%	2.6%
6.	3000	75.9%	96.5%	96.5%	6.	3000	24.1%	3.5%	3.5%

relatively good performances achieved, with a minimum accuracy record of 69.4% and a maximum error rate of 30.6% for Naïve Bayes. The Stacked ensemble model additionally demonstrated a good ability to cope with errors as indicated in Table 4.

FUNDING AGENCY

The publisher has waived the Open Access Processing fee for this article.

REFERENCES

- Abuassba, A. O., Zhang, D., Luo, X., Shaheryar, A., & Ali, H. (2017). Improving Classification Performance through an Advanced Ensemble Based Heterogeneous Extreme Learning Machines. *Computational Intelligence and Neuroscience*, 2017, 1–11. Advance online publication. doi:10.1155/2017/3405463 PMID:28546808
- Akhtar, M. S., Gupta, D., Ekbal, A., & Bhattacharyya, P. (2017, June). Feature selection and ensemble construction: A two-step method for aspect based sentiment analysis. *Knowledge-Based Systems*, 125(1), 116–135. doi:10.1016/j.knosys.2017.03.020
- Alnashwan, R., O’Riordan, A.P., Sorensen, H., Hoare, C. (2016). Improving Sentiment Analysis through Ensemble Learning of Meta-level Features. In *KDWEB 2016: 2nd International Workshop on Knowledge Discovery on the Web*. Sun SITE Central Europe (CEUR)/RWTH Aachen University.
- Alpaydin, E. (2014). *Introduction to machine learning*. MIT Press.
- Araque, O., Corcuera-Platas, I., Sánchez-Rada, J. F., & Iglesias, C. A. (2017, July). Enhancing deep learning sentiment analysis with ensemble techniques in social applications. *Expert Systems with Applications*, 77(1), 236–246. doi:10.1016/j.eswa.2017.02.002
- Balahur, A., & Perea-Ortega, J. M. (2015). Sentiment Analysis system Adaptation for Multilingual Processing: The case of Tweets. *Information Processing & Management*, 51(4), 547–556. doi:10.1016/j.ipm.2014.10.004
- Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, 2(1), 1–127. doi:10.1561/22000000006
- Breiman, L. (1996). Stacked Regressions. *Machine Learning*, 24(1), 49–64. doi:10.1007/BF00117832
- Breiman, L. (1998). Arcing classifiers. *Annals of Statistics*, 26(3), 801–824.
- Brownlee, J. (2019, August 12). Supervised and Unsupervised Machine Learning Algorithms. *Machine Learning Mastery - Making Developers Awesome at Learning*. <https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/>
- Demir, N. (2015). Ensemble Methods: Elegant Techniques to Produce Improved Machine Learning Results. *Data Science and Databases*. <https://www.toptal.com/machine-learning/ensemble-methods-machine-learning>
- Dhiraj, K. (2019, May 20). Top 5 advantages and disadvantages of Decision Tree Algorithm. *Medium*. <https://medium.com/@dhiraj8899/top-5-advantages-and-disadvantages-of-decision-tree-algorithm-428ebd199d9a>
- Dietterich, T. G. (2000). *Ensemble Methods in Machine Learning*. doi:10.1007/3-540-45014-9_1
- Džeroski, S., & Ženko, B. (2004). Is Combining Classifiers with Stacking Better than Selecting the Best One? *Machine Learning*, 54(3), 255–273. doi:10.1023/B:MACH.0000015881.36452.6e
- Fersini, E., Messina, E., & Pozzi, F. (2014). Sentiment analysis: Bayesian ensemble learning. *Decision Support Systems*, 68, 26–38. doi:10.1016/j.dss.2014.10.004
- Girard, J. M. (2015, January 1). *Research Gate*. Retrieved from Research Gate: https://www.researchgate.net/post/in_simple_words_what_do_you_mean_by_feature_vector_in_image_processing
- Hansen, L., & Salamon, P. (1990). Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10), 993–1001. doi:10.1109/34.58871
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). Springer Series in Statistics. doi:10.1007/978-0-387-84858-7
- Hossin, M., & Sulaiman, M. N. (2015). A Review on Evaluation Metrics for Data Classification Evaluations. *International Journal of Data Mining & Knowledge Management Process*, 5(2).
- Huang, J., Lu, J., & Ling, C. X. (2003). Comparing naive Bayes, decision trees, and SVM with AUC and accuracy. *Third IEEE International Conference on Data Mining*, Melbourne, FL. doi:10.1109/ICDM.2003.1250975
- Jurafsky, D., & Martin, J. H. (2017). *Speech and Language Processing* (3rd ed.). Academic Press.

Kalaivani, P., & Shunmuganathan, K. L. (2013). Sentiment Classification of Movie Reviews by Supervised Machine Learning Approaches. *Indian Journal of Computer Science and Engineering*, 4(4), 285-292.

Korjus, K., Hebart, M. N., & Vicente, R. (2016). An Efficient Data Partitioning to Improve Classification Performance While Keeping Parameters Interpretable. *PLoS One*, 11(8), e0161788. doi:10.1371/journal.pone.0161788 PMID:27564393

Liu, A. Y., & Martin, C. E. (2011). Smoothing Multinomial Naïve Bayes in the Presence of Imbalance. Machine learning and data mining in pattern recognition. In *7th international conference, MLDM 2011, New York, NY, USA, August 30 – September 3, 2011 Proceedings* (pp. 46-59). doi:10.1007/978-3-642-23199-5_4

Lo, S. L., Cambria, E., Chiong, R., & Cornforth, D. (2016). Multilingual Sentiment Analysis: From Formal to Informal and Scarce Resource Languages. School of Design, Communication and Information Technology, The University of Newcastle.

Mayer, Z. (2019, December 12). *A Brief Introduction to caretEnsemble - caretEnsemble-intro*. Retrieved from Contributed Packages - The R Project for Statistical Computing: <https://cran.r-project.org/web/packages/caretEnsemble/vignettes/caretEnsemble-intro.html>

Melville, P. (2003). *Creating Diverse Ensemble Classifiers*. The University of Texas at Austin.

Mitchell, T. (1997). *Machine Learning*. McGraw Hill.

Mitchell, T. M. (1998). *Machine Learning*. Academic Press.

Moudřík, J., & Neruda, R. (2015). Evolving Non-linear Stacking Ensembles for Prediction of Go Player Attributes. *2015 IEEE Symposium Series on Computational Intelligence*, 1673–1680. doi:10.1109/SSCI.2015.235

Nasteski, V. (2017). *An overview of the supervised machine learning methods*. 10.20544/HORIZONS.B.04.1.17.P05

Ogut, R. V., Rimiru, R., & Otieno, C. (2019, July). Target Sentiment Analysis Model with Naïve Bayes and Support Vector Machine for Product Review Classification. *International Journal of Computer Science and Information Security*, 17(7), 1–17.

Osisanwo, F. Y., Akinsola, J. E. T., Awodele, O., Hinmikaiye, J. O., Olakanmi, O., & Akinjobi, J. (2017, June). Supervised Machine Learning Algorithms: Classification and Comparison. *International Journal of Computer Trends and Technology*, 48(3).

Oza, N. C., & Tumer, K. (2008). Classifier ensembles: Select real-world applications. *Information Fusion*, 9(1), 4–20. doi:10.1016/j.inffus.2007.07.002

Quinlan, J. (1996). Bagging, Boosting, and C4.5. *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, 725-730.

Rocca, J. (2019, April 23). Ensemble methods: bagging, boosting and stacking. *Understanding the Key Concepts of Ensemble Learning*.

Rodriguez, J. (2019, March 29). *The Three Pillars of Robust Machine Learning: Specification Testing, Robust Training and Formal Verification*. Retrieved from Towards Data Science: <https://towardsdatascience.com/the-three-pillars-of-robust-machine-learning-specification-testing-robust-training-and-formal-51c1c6192f8>

Rokach, L. (2005). Ensemble Methods for Classifiers. In O. Maimon & L. Rokach (Eds.), *Data Mining and Knowledge Discovery Handbook*. Springer. doi:10.1007/0-387-25465-X_45

Rokach, L. (2010). Ensemble-based classifiers. *Artificial Intelligence Review*, 33(1), 1–39. doi:10.1007/s10462-009-9124-7

Saugata, P. (2018, November 30). Ensemble Learning. *Bagging, Boosting, Stacking and Cascading Classifiers in Machine Learning using SKLEARN and MLEXTEND libraries*. <https://medium.com/@saugata.paul1010/ensemble-learning-bagging-boosting-stacking-and-cascading-classifiers-in-machine-learning-9c66cb271674>

Schapire, R. E. (1990). The Strength of Weak Learnability. *Machine Learning*, 5(2), 197–22. doi:10.1007/BF00116037

- Seijo-Pardo, B., Porto-Díaz, I., Bolón-Canedo, V., & Alonso-Betanzos, A. (2017). Ensemble feature selection: Homogeneous and heterogeneous approaches. *Knowledge-Based Systems*, 118, 124-139.
- Sethi, A. (2020, April 6). Supervised Learning vs. Unsupervised Learning – A Quick Guide for Beginners. *Analytis Vidhya*. <https://www.analyticsvidhya.com/blog/2020/04/supervised-learning-unsupervised-learning/>
- Tang, J., Alelyani, S., & Liu, H. (2015). *Data Classification: Algorithms and Applications*. CRC Press.
- Tuwe, L. (2015). *On Effectively Creating Ensembles of Classifiers: Studies on Creation Strategies, Diversity and Predicting with Confidence* (PhD thesis). Stockholm University.
- Viola, P., & Jones, M. (2004). Robust Real-Time Face Detection. *International Journal of Computer Vision*, 57(2), 137–154. doi:10.1023/B:VISI.0000013087.49260.fb
- Wolpert, D., & Macready, W. (1997). *No Free Lunch Theorems for Optimization*. *Evolutionary Computation. IEEE Transactions on Evolutionary Computation*, 1(1), 67–82. doi:10.1109/4235.585893
- Wolpert, D. H. (1992). Stacked generalization. *Neural Networks*, 5(2), 241–259. doi:10.1016/S0893-6080(05)80023-1 PMID:18276425
- Wolpert, D. H. (1996). *The Supervised Learning No-Free-Lunch Theorems: Soft Computing and Industry*. Springer.
- Zhou, Z.-H. (2012). *Ensemble Methods - Foundations and Algorithms*. Chapman & Hall/CRC.

Rhoda Viviane Ogutu received her undergraduate degree in Business Computing from Kampala International University, Uganda in 2009. She is currently pursuing her Master's Degree in Information Technology, MscIT while also employed as a part-time lecturer at Jomo Kenyatta University of Agriculture and Technology – JKUAT, Kenya; teaching diploma students on Information Technology issues. She has also attained a Diploma in Tax Administration from Kenya School of Revenue Administration (KESRA) and is also employed at Kenya Revenue Authority – KRA as a Tax Officer, thereby possessing expertise on Kenyan taxation law. She writes and widely presents on issues about Artificial Intelligence, Data science and is the correspondent author of Target Sentiment Analysis Model with Naïve Bayes and Support Vector Machine for Product Review Classification (2019, International Journal of Computer Science and Information Security - IJCSIS).

Richard Rimiru is a Computer scientist, researcher and educator who attained BSc. Degree in Statistics and Computer Science from Jomo Kenyatta University of Agriculture and Technology – JKUAT, Kenya, in 1999; Master's Degree, MSc. in Computer Science from the National University of Science & Technology, Zimbabwe in 2002 and PhD in Computer Science and Technology from Central South University, Changsha, China in 2013. He is currently working as a Senior lecturer at the School of Computing & Information Technology at JKUAT, Kenya. His areas of interest include the wider Artificial Intelligence and Data Science.

Calvins Otieno received his Master's Degree - MSc Software Engineering in 2013 and PhD IT in 2016 from Jomo Kenyatta University of Agriculture and Technology – JKUAT, Kenya. He is currently a lecturer in Department of Information Technology at JKUAT. He teaches computer science to both undergraduate and post graduate students. He is also involved in post graduate supervision and research. His major areas of interest in research are in the fields of Artificial Intelligence and Software Engineering. He has supervised a number of masters' students and co-authored papers with students in Fuzzy logic, Software quality and Systems security. His major publications are; Hybrid Machine Learning Approach for Credit Scoring using PCA and logistic regression, Multimodal Biometric System:-Fusion Of Face And Fingerprint Biometrics At Match Score Fusion Level and the third on Fuzzy Logic Sensor Fusion for Obstacle Avoidance Mobile Robot in static unknown environment.