International Journal of Engineering &Technology, *8 (4) (2019) 627-635*

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Content-based prediction: big data sampling perspective

Waleed Albattah [1] *, Saleh Albahli [1]

[1] Department of Information Technology College of Computer, Qassim University Saudi Arabia
**Corresponding author E-mail: w.albattah, salbahli @qu.edu.sa*

Abstract

Today, large volumes of data are actively generated on the order of terabytes or even petabytes. Hence, processing data on such a large scale in an efficient and effective manner is extremely challenging. However, existing research studies apply machine learning algorithms by loading the entire training dataset into the computer's main memory (RAM). This causes a problem as the data grows too big over time and can't be supported by most of the conventional models or hardware within a single machine's memory. Inspired by current research studies, this paper discusses the benefits of implementing two sampling techniques that could be used for machine learning models: (1) sampling with replacement and (2) reservoir sampling. In this study, 40 experiments were performed by reducing the number of data instances by 50% of the original data using random sampling of a video dataset that was more than 40 GB in size. Remark that accuracies of SVM and random forest are very competitive classifiers and give the importance score of all repeated ten rounds of the process for each of the four combinations of sampling techniques and machine learning classifiers.

Keywords: *Sampling Techniques; Sampling with Replacement; Reservoir; Big Data; Machine Learning; Classifier; SVM; Random Forest.*

1. Introduction

Today in any industry or sector, huge amounts of data are continuously generated by specific systems daily. The term "big data" was created to refer to huge volumes of data that required processing. Big data involves the collection and processing of structured and unstructured data from a wide variety of sources. This data, being so big and huge, requires special tools and techniques for processing and analyzing. More specifically, big data is defined as "data that is too big, too fast, or too hard for existing tools to process" [1]. An additional meta definition that is often used to describe big data is that it is "data whose size forces us to look beyond the tried-and-true methods that are prevalent at that time." [2].

Analyzing huge data requires high computational power and significant time and effort, which is very inefficient in emergency cases when time is of the essence or in cases where limited computing resources are available. In order to deal with this problem, machine learning algorithms are developed, and through programming, this big data can be sorted, with only a small amount of data sampled. The resulting small sample, however, should be an accurate representation of the whole dataset. Additionally, this technique helps by intelligently sampling data to predict future data and the results after analysis. Through the use of efficient algorithms, there will be a preprocessing requirement of big data in the future.

Sampling is a method to select a small representative data subset from the original set of data for analysis. The main goal is to use machine learning techniques in order to reduce the data for efficient processing. This technique should, in most cases, yield the same results found by processing the original dataset. Rather than employing the usual sampling techniques like density sampling, machine learning uses simple random sampling in which data is retrieved through uniform probability [3]. However, in systematic sampling, data is drawn at specified intervals. Another technique called stratified sampling collects data from specified categories. Resampling, on the other hand, is a technique used to draw data again to improve the algorithm for better processing..

There are a number of analytic techniques for big data sampling. One that is widely implemented is machine learning, which uses computer algorithms to predict outcomes and provide data analysis. Data fusion and integration is another technique by which data is drawn from multiple sources for analysis. There is also data mining, which uses statistical and machine learning methods together to analyze data.

Volume in big data is generally referred to as huge data generation in a period of time. To understand this, consider the huge number of emails and Twitter and Facebook posts that are produced every second all over the world. The companies hosting these services analyze and process this massive volume of data to gain useful information from it. At times, the quantity of data can be huge and reach sizes of up to zettabytes or brontobytes.

The use of huge amounts of data is not necessarily guaranteed to provide lots of useful information. Data is used to predict usage patterns and results after analysis and processing. In order to run our set of experiments smoothly without requiring many computational resources or expending significant time, the data has been reduced in size, and this work here is the challenging part of this entire project. This process is the main object and goal of this research, which targets the sampling of data in a big data paradigm. High sampling accuracy can be achieved by implementing advanced methods without using entire datasets. Thus, our experiments achieved good performance results using smaller 50% random sampling datasets when compared with other classical techniques that use the entire original datasets. We ultimately measured the performance and the accuracy of the produced models to draw our final conclusions.

The rest of the paper is organized as follows. Section 2 reviews up-to-date literature on big data sampling. In section 3, we have reported the two machine learning algorithms used in our experiments and their analytics. Section 4 presents our proposed model for sampling techniques with its experimental analysis and evaluation results. Experimental discussions are reported in Section 5. Finally, conclusions and perspectives are given in Section 6.

2. Background

With the increased capabilities of new technology, there has been a subsequent rise in sophisticated technologies and systems, which in turn, has led to the development of cutting-edge techniques and processes for data collection, processing, and classification. The vast volumes of data associated with these systems is now termed “big data.” Therefore, as the name implies, big data is defined as a data type that is huge in size. As such, fetching, processing, and implementing big data takes considerable processing time and automation, along with a formatting style that fits the organization of data and the potential that it requires time as well [4]. Generally speaking, this big data is used in applications to predict future outcomes in order to make informed decisions based on these outcomes. It is, therefore, quite elaborative that big data is termed and processed by large servers and is accessed as per requirement [5].

This processing means that the dataset or the big data used for making implementations is totally based on the decision-making of the organization, and in the process of this decision making, some essential tasks of the business are carried out [6]. Though there is a choice for the processing of data as the size and application are directly proportional, that is to say, that with the increase in performance, the application will be of greater size and require more processing time [7]. There are numerous examples of this positioning system as it is used for the automation of vehicles and recognition of facial features.

With more data, this software increases its efficiency, and with increased datasets, more and diverse forms of data training are required. Moving forward towards a more practical approach, the storage of this big data is not the only problem, as it also requires additional processing time. These two requirements make it difficult for the whole dataset to be properly analyzed. In order to achieve optimization in this regard, there must be a suitable mechanism of extracting subsets of data to identify key attributes between similar knowledge present in the subsets and the original dataset [8].

Big data has been identified as a risk factor on the basis of the needed computational processing resources as well as the extended period of time for the data to be analyzed that often doesn’t support timely decision-making. This is the need that must be adequately addressed. In order to address this need, the greatest perceived threat is a data leak. The users providing the data have a lot of data that they pass on without consideration.

This risk must be considered while running big data queries. To effectively deal with the problem, some viable parameters should be developed, and in the process of development, big data quality and the quality of information should be preserved [9]. The parameters should include but not be limited to the association of identity, syntactical validity, the valid and appropriate association of attributes, precision, accuracy, theoretical relevance, temporal applicability, audibility, controls, currency, and completeness.

In addition, there are other issues created by the management of servers, access control of data, and privileges paired with sortation and security. These issues are also vital for a complete and thorough analysis [10]. It is a record that by the year 2002, the total data in said 92% of the digital devices were more than five exabytes [11]. Moreover, the amount of data has been increasing at a rapid pace since then, and with it, the problems of adequate computational resources and processing time. Now the industry has become a $46.4 billion industry [11], indicating that the problems of data handling and the interests of users have been growing exponentially over the last decade. There are now hundreds of groups that are purely focused on data mining, collecting and classifying, with just minor differences in their approaches, which has increased the workload and contributed to the compilation time.

Numerous applications have been created to deal with the concept of big data and are addressed globally through data mining, computational intelligence, semantic web, and information fusion [12]. Because of this concept, the data processing issues, use of data mining patterns, stage and retrieval of data, along with data visualization and tracking of data has captured a lot of attention [13].

This is the solution that has been searched far and wide, for the issues pertaining to assembly issues have been exponential in its increase, and this has created havoc in the industry, as they use various libraries and classes for the processing of data. As the libraries progress, this, in turn, increases the size and speed of the application. These libraries are increasing, and so is the potential for mining with greater granularity. This drives continued research and development to identify the most optimal big data processing solution in terms of memory consumption, time consumption, and speed. Research to date has led to the application of various approaches such as incremental learning, grid searches, divide and conquer, and distributed computing. There are other methods and processes that make the procedure of this processing complex, and it does incorporate other methods of data division [11].

As far as computing and processing are concerned, the most significant issue that big data presents is its complexity. This is, unfortunately, an outside factor that can’t be controlled and serves as a determinant of the computational budget and the inefficiencies of fulfilling the completed tasks up to its full potential [14]. Currently, there are a number of sampling efforts in place, and typically this leads to an increased number of datasets that are created as a result of this sampling. However, it is a general assumption that if the sampling effort induces some biases, with the help of sampling effort that makes richness in a sample and its method of determination [15]. There is a selection bias that makes computation and determination of the sampling technique. A conventional data sampling technique typically takes more time, so in order to battle this effect, an inverse sampling method is to be applied, and this is to be done with the integration of big data and the probability or p-value of each sample [16]. In this study, the size of the sample is of critical importance as it plays a vital role in accuracy determination by the system under analysis [17]. In previous approaches, there have been numerous instances where countless issues occurred in the accuracy of big data and its sampling. One such approach is the Zig Zag Method, along with non-probability sampling [18], inverse sampling and cluster-based sampling [19].

It can be seen that machine learning is an integral part of data analytics and comes with a steep learning curve. This not only makes the available data that is depicting a trend and then implements the trend to learn to predict the future [19]. After the preliminary analysis is done, there is a computation conducted for further analysis to make the data available for either a supervised machine learning model or an unsupervised model. These are the core difference of techniques that are used in data science for predictive analytics.

The core of machine learning lies in the art of making the machine understand the evident and underlying trends of the analysis, and is used to determine a specific meaning from a data mining learning exercise to improve predictability over previous learning exercises. This processing is only possible if there is some data available to the problem, and the machine is given all the tools and techniques needed to derive the specific problems and to make useful meanings out of them. This derivation of meaning is based on the analysis of different analogies along with the connections that exist between them. It includes the implementable strategies from derived results and ultimately makes the adjustments of different parameters that guide different machine learning models and algorithms. There are some serious constraints attached to each machine, and the results are totally dependent upon the amount of information that is provided for processing and the amount of information a machine can handle [7].

This makes the machine learning algorithms that make the enhancement that makes the accuracy better with an increase in the number of input observations provided. There is a downside to machine learning as the algorithms for learning are the primary sources for information gathering and computation. In general, the data sets that are used and the machine learning algorithms are relatively straightforward. Moving ahead with the simple datasets, this can create challenges for complex and big datasets. With further observation, it can be seen that there is a division for training and testing for almost as different periods for both of these. These issues arise, like if the data is fast-moving as in the stock market, or consumes energy, then there are unlabeled datasets and an unbalanced distribution of samples [20].

The conventional neural networks, or CNNs, are usually coupled with accurate data modeling according to different classifications that make the processing power. This is the case if the data is either an image or some text data. It is observed that CNNs typically require more processing power. This processing power enhances the future of the dataset by making it more compatible and more rigorous.

Table 1 summarizes the above literature.

Table 1: Comparison Table of Previous Methodologies for Big Data Sampling

Reference	Technology Discussed/used	Machine Learning/ Big Data	Methodology	Findings
[4]	R programming	Machine learning	The methodology opted in this paper is to power the sampling method in big data analysis	The findings identify a key sampling technique, and it also illustrates that the potential of the sampling technique is of vital importance.
[5]	IBM Modeler	Big Data	This paper takes the literature as its primary approach and illustrates the role of big data in critical decisions and challenges.	This results in the findings that on one hand big data leads to vexing issues while on the other hand, has helped various fields, including but not limited to health, security and disaster management.
[6]	C C++	Big Data	This paper opts to the compilation of relevant literature along with computing methods that make the potential increase in relevant fields.	The results or conclusion of this paper illustrates that increase in design and schematics of the circuits can lead to an exponential increase in their processing power.
[7]	R Matlab Weka	Both	The methodology used here is based on both machine and deep learning, and it uses the big data concepts to illustrates the decision charts showing the series of different concepts	The findings of this paper includes the systematic review of challenges discussed. It differentiates between different ML approaches and how each approach effects the speed and accuracy of the problem.
[8]	Hadoop	Machine learning	Since this paper is based on bot detection, it can be said that the author checked various open source tools to check the bot detection of the company, along with using machine learning to counter these attacks.	As a result, it was seen that the power of Mahout, with a random forest-based trees, would make the detection of peer to peer type bots in real time. As a result of some initial testing, the implementation of this setup, with added performance metrics to highlight the effects on the potential scenario at hand.

3. Classification

3.1. Support vector machine

Support Vector Machine (SVM) is a supervised learning classifier that is introduced in 1990s by Boser, Guyon, and Vapnik [21]. It is widely used because of its accuracy, ability to deal with high-dimensional data, and its flexibility in modeling different sources of data. The SVM has two advantages: first, it has the ability to produce non-linear decision boundaries by using methods of linear classifiers; secondly, the classifier can be applied to data with no fixed- dimensional vector space representation [22]. Moreover, SVM has a robust theoretical foundation, which is statistical learning theory; and successful empirical applications as well. It has been applied to different fields such as hand written digits recognition, text classification, and objects recognition [22]. The SVM is in this article is used due to its over-all good detection performance in similar areas.

3.2. Random forest

Random Forests is a very popular ensemble learning method which builds a number of classifiers on the training data and combines all their outputs to make the best predictions on the test data. Random forest algorithm has seen success of late and thus can be one of our models for predicting target values.

The Random Forests algorithm is a variance minimizing algorithm that uses randomness when making split decision to help avoid overfitting on the training data. The training algorithm for Random Forest applies the general technique of bootstrap aggregating, or bagging, to tree learners. Given a training set $X = x_1, x_2, \ldots, x_n$with responses $Y = y_1, y_2 \ldots, y_n$, bagging repeatedly selects a random sample with replacement of the training set and fits trees to these samples. Random Forests further use a modified tree learning algorithm that selects, at each candidate split in the learning process, a random subset of features for training the tree. Thus if $D(x, y)$ denotes the whole training set , and $f(x|\theta_1), f(x|\theta_2), \ldots\ldots, f(x|\theta_k)$denotes each of the tree classification models, then each model $f(x|\theta_j)$ is built using a different subset $D_{\theta_j}(x, y) \subset D(x, y)$.

The final output y is built by aggregating the results in the following manner : $y = \mathrm{argmax}_{p\in\{f(x_1),\ldots.f(x_k)\}}\{\sum_{j=1}^{k} I(f(x|\theta_j) = p)\}$, where I is an indicator function, such that $I(\mathrm{True}) = 1$ and $I(\mathrm{False}) = 0$.

Two popular methods of classification trees have grabbed researchers' attention: bagging and boosting. These two methods can generate many classifiers and aggregate their results [23]. One of the important advantages of Random forest is that it can be used for regression or classification problems. In an enhancement addition to bagging, Breiman [24] proposed random forests as an additional layer of randomness. Either in regression or classification problems, Random forest can help in ranking the importance of variables. Random forest has only two parameters: the number of trees in the forest and the number of variables in the node. These two parameters constitute to the straightforwardness of Random forest. Moreover, it constructs every tree with a different bootstrap sample of data, which changes how trees are constructed in regression and classification. Each node is split by the best predictor chosen at the node randomly among a subset of predictors [24]. Many trees are grown and every tree vote for a particular class. The class with high number of trees is the final class assigned to particular data instance.

4. Experimental setup and results

4.1. Dataset

To conduct our experiments, the article uses datasets from the NDPI videos. Further details of the NDPI dataset is available in [25]. NDPI is a huge dataset that is comprised of more than 40 Gigabytes of video data. For experimental analysis, the data is divided into three classes of data; Unacceptable, Acceptable, and Flagged. Figure 1 shows some samples. The experiment setup uses the data from image-based filtering and a large amount of data. It has three main reasons. Firstly, the data is well organized into three classes, which is a good representation of the problem for machine learning algorithms? Secondly, although the data is an image in the feature form, the data is converted into numerical values. Thus, the data equates to other datasets and similar machine learning problems. Thirdly, the data set is considered huge because it is more than 40 Gigabytes. Therefore, it is assumed that the data that is processed in this article is big data, and so the results can be extended to other datasets of similar nature.

For feature extraction, the article uses the autocorrelogram. We use the F-measure as an evaluation parameter as it is mostly used in state-of-the-art applications for similar problems and is favorable for this evaluation as well. F-measure takes into account Precision and Recall.

Fig. 1: Representative Sample Images from NDPI [25].

4.2. Sampling

In data analysis domains, an instance represents the individual object of which the problem is composed of. This means that if the problem is based on the color, such as in computer vision, the instance is the set of pixels for the problem concerned. The instance may also represent a complete image if the features are globally extracted from the images. In most cases, the instance is directly related to the number of objects available for training and testing. If instances are reduced, the amount of training data, and ultimately the testing data, is reduced. If instances are increased, the amount of training and testing data is also increased. If a ten-fold cross-validation technique is used, the increase in the number of instances results in an increase of 90% in the training samples and 10% in the testing samples. This can have one of three impacts on the results. The result could remain neutral, or it could increase in some cases and decrease in others.

The neutral case can occur in two ways. The first happens if the instance added is of similar nature to the previous data. This means that the instance is already represented in the model of the machine learning classifier, and its addition has contributed no extra information. Thus, there is an increase in the size of the dataset without any subsequent benefit to the machine learning model. The second neutral case occurs when the contribution of the additional instance is negligible due to the large number of existing data samples. In other words, the data is already covering most of the model generation cases, and no additional data is required.

The increase in classification results because of the increase in the number of instances can be attributed to the fact that the new instances contribute strong classification information to the model. In other words, it means that the new addition closely represents the classes in the dataset and also exhibits a strong correlation with the attributes of that instance. This type of scenario is always ideal and the objective in the machine learning paradigm. However, every machine learning algorithm has certain limits, and adding more strong instances may not contribute any additional useful information for classification purposes. One of the interesting phenomena that can occur by adding strong instances is what is called “overfitting.” In this case, the model can become very diverted to special cases and does not generalize well.

The decrease in classification performance can be due to either the addition of new instances that are not related to the classes in specific problems or because the added instances represent (contribute) strong noise to the model. This phenomenon is quite common, so collecting the correct dataset is the challenge for most machine learning related problems. Therefore, the data cleaning procedure is essential in many classification tasks to produce reliable model generation. The decrease in classification performance can also be due to a small number of data instances. Many machine learning algorithms require a considerable amount of data (not big data) for reliable model generation and generalization of unseen test data instances. However, increasing the amount of data to a specific limit does not always mean that performance will also increase. Every classifier has limits for specific problems, and thus, a thorough analysis is required for the final model generation and to determine the amount of data needed for the particular problem.

The experiments of this study aim to analyze the influence of sampling techniques on the outcome of models. Two sampling techniques are used in the experiments; sampling with replacement and reservoir sampling. Our approach in the experiments is to minimize the data using random sampling by reducing the number of instances by 50% and measuring the performance and accuracy of the produced model. This process is repeated in ten rounds. In each round, a sample consisting of 50% of the data is randomly selected and analyzed by using 90% of the data for training and 10% for testing the generated model. This means that a specific sample set of data is half the size of the original data, and 90% of this data is used for generating the model by training the classifier, while 10% of the data is used for testing the performance of the generated model. This experiment is repeated ten times for each of the four combinations of sampling techniques and machine learning classifiers. At the conclusion of testing, 40 experiments will have been performed for the study. Table 2 presents the four combinations of sampling techniques and machine learning classifiers. Figure 1 shows the rounds of the experiments for the first combination.

Table 2: Four Combinations of Sampling Techniques and Machine Learning Classifiers

		Sampling Techniques	
		Sampling with replacement	Reservoir sampling
Classifiers	SVM	Combination 1	Combination 3
	Random Forest	Combination 2	Combination 4

In the first combination, sampling with replacement and SVM classifier is used. 50% of the original data are selected with replacement technique in each round, which means that every time an instance is randomly selected, it is returned back to the original data, and a new instance is randomly selected and so on. There is a possibility that the same instance is selected more than one time for a given sample. To measure the performance and the accuracy of the generated model, F-measure is calculated in each round. The first experiment, which uses the original collection of data (100% of data), is called base round. The F-measure for the base round (original collection of data) is 0.784. Normally, it is acceptable to think that the F-measure for 100% of the data would yield the best value among all the F-measure values for combination 1 experiments since all instances of the data are used to generate the model. However, this is not true, at least in the combination 1 experiment, which adds more value to the approach of sampling big data. In the first round of the combination 1 experiment where only 50% of the data is randomly selected, F-measure is 0.819, which is better than the F-measure of the base round (100% of the original data). Interestingly, the performance in this round (round 1) beats the performance when the entire original dataset is used. F-measure for the second round is even better at 0.872. However, F-measure is decreased in round 3 to 0.775. Although F-measure is decreased in this round, it is still very close to the F-measure value of the original collection of data (100%). Moreover, when comparing the amount of data for both sets of the considered collections, the F-measure value for round 3 can be considered an improvement based on the cost of resources used for processing. F-measure continues to increase in round 4 and registers 0.842. In the fifth round, F-measure registers the highest value (0.873) among the ten rounds as well as the base round. Likewise, F-measure keeps registering values better than the F-measure of the base round (original collection of data). F-measure values for round 6, 7, 8, and 9 are 0.825, 0.824, 0.811, and 0.845, respectively. In the last round, F-measure is 0.855. The average F-measure of all the ten rounds (labeled as average round) is 0.834, which is even better than the F-measure of the base round (where 100% of the data is used). Consequently, the difference between F-measure values of the base and the average of all rounds is 0.050 (i.e., 5% from the original dataset). Therefore, the model of sampled data is 5% better and more accurate than the model that uses 100% of the data.

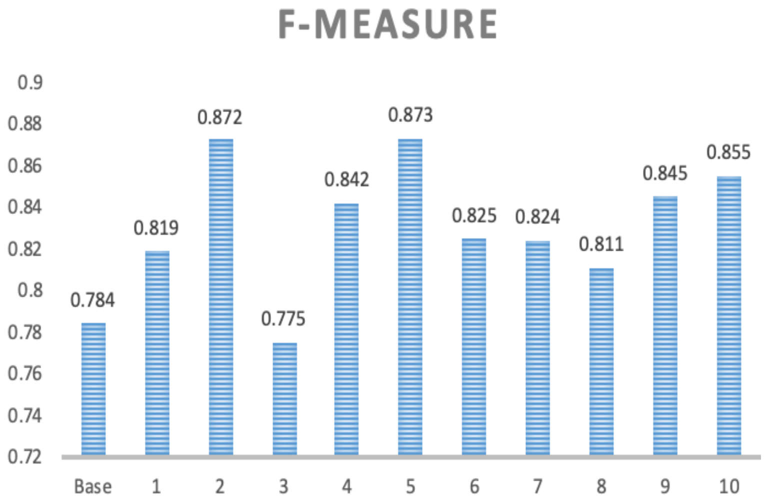

Fig. 2: The Base and Ten Rounds of the Experiments for Combination 2 with Replacement Technique.

In the second combination of the experiments, sampling with replacement is again used, but this time with a random forest classifier. Figure 2 shows the base and the ten rounds of experiments for combination 2. Similar to combination 1 experiments, 50% of the original data are selected using a replacement technique in each round, which means that every time an instance is randomly selected, it is returned back to the original data, and a new instance is randomly selected and so on. There is a possibility that the same instance is selected more than one time in a given sample. F-measure is calculated in each round to measure the performance and accuracy of the generated model. The F-measure for the base round (where 100% of data is used) is 0.841. A general look at figure 2 shows that F-measure for all ten rounds is better than the F-measure for the base round. Round 1 and 2 have F-measures of 0.858 and 0.888, respectively. F-measure in round 3 is reduced to 0.875 but is still better than the base round value. F-measure increases in round 4 to 0.880. In rounds 5, 6, and 7, F-measure has a slight decrease in each round and registers 0.886, 0.882, and 0.861, respectively. However, even with this decrease, F-measure still returns better values in these rounds than the base round. F-measure then begins to increase slightly to 0.866 and then has a slight drop off to 0.864 in round 9. In the last round, F-measure increases and registers 0.893, which is the best value among all the experiments in combination 2. The average F-measure for all of the ten rounds (labeled as average round) is 0.875, while F-measure for the base round (where 100% of data is used) is 0.841. Therefore, the difference between F-measure values of the base and average rounds is 0.034 (i.e., 3.4% from the original dataset). Therefore, the model of sampled data is 3.4% better and more accurate than the model that uses 100% of the data.

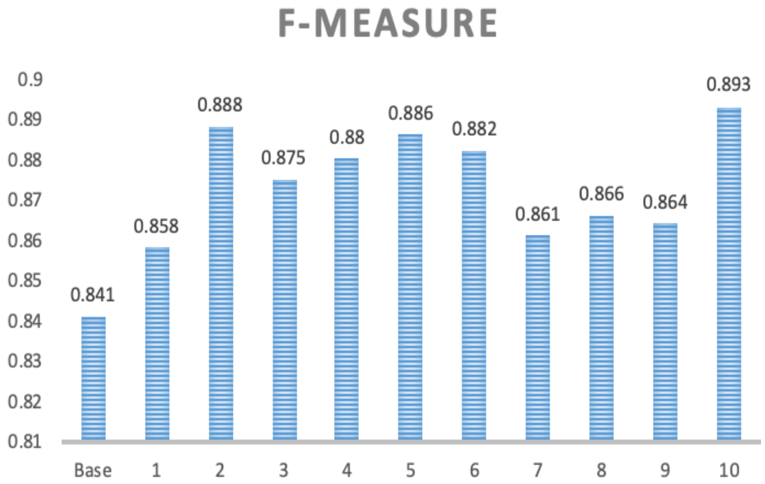

Fig. 3: Comparisons of Ten Rounds F-Measure with the Base Round F-Measure.

In combination 3, a reservoir sampling technique and SVM classifier are used. Figure 3 shows the F-measure for the base round as well as the ten sample rounds. An F-measure of 0.784 is registered for the original data (100%). Clearly, a careful review of the F-measure of the ten rounds, reveals that most of the values are less than the base round, where 100% of the data is used. F-measure for round 1 is 0.745. In round 2, F-measure increases to 0.771. The F-measure for round 3 then decreases to 0.737. Round 4 has an F-measure of 0.705. Rounds 5, 6, and 7 have F-measures of 0.749, 0.735, and 0.750, respectively. With a very slight increase in round 8, F-measure registers 0.787, which is followed by a decrease in round 9 to 0.720. The last round yields the highest value of F-measure at 0.807 among all of the ten experiments. The average F-measure for all ten rounds (labeled as average round) is 0.751, while F-measure for the base round (where 100% of data is used) is 0.784. Therefore, the difference between F-measure values of the base and average rounds is 0.033 (i.e., 3.3% from the original dataset). Therefore, the model that uses 100% of the data is 3.3% better and more accurate than the model of sampled data.

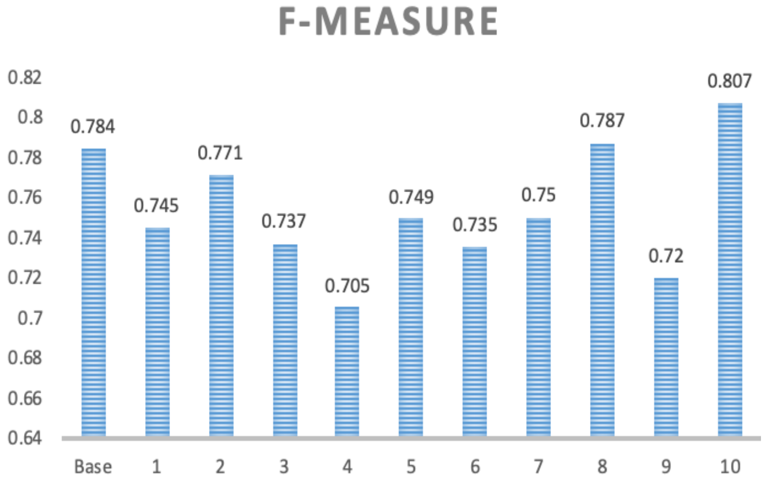

Fig. 4: F-Measure for Base Round and Also the Ten Sampling Rounds.

In the last combination, a reservoir sampling technique and a random forest classifier are used. Figure 4 shows F-measure for the base round as well as the ten sampling rounds. An F-measure of 0.841 is recorded for the original data (100%). Obviously, a careful review of Figure 4 shows that the F-measure of the ten rounds fluctuates up and down. However, all of the values are less than that of the base round, where 100% of the data is used. The F-measure for the first round is 0.807. In round 2, F-measure increases to 0.815. F-measure for round 3 dips to 0.800. Round 4 has an F-measure of 0.818. Rounds 5, 6, and 7 have F-measures of 0.791, 0.814, and 0.826, respectively. In round 8, F-measure decreases to 0.780, followed by an increase in round 9 to 0.840. F-measure for the last round is 0.799. The average F-measure (labeled as average round) for all of the ten rounds is 0.809, while F-measure for the base round (where 100% of the data is used) is 0.841. Therefore, the difference between the F-measure values of the base and average rounds is 0.032 (i.e., 3.2% from the original dataset). Consequently, the model of 100% of data is 3.2% better and more accurate than the model of sampled data. Therefore, the model that uses 100% of the data is 3.2% better and more accurate than the model of sampled data.

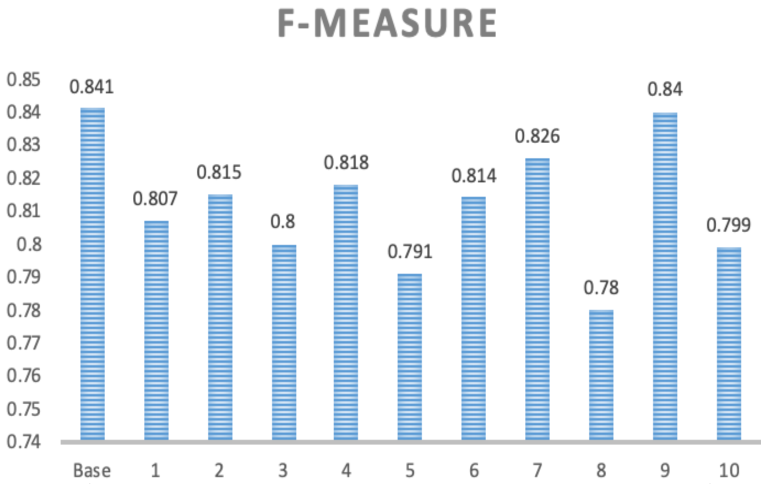

Fig. 5: F-Measure for the Base Round (100% Data Used) Achieve Better Than the Model of Sampled Data.

5. Discussion of results

Four different types of experiments are performed in this study. Table 1 presents four combinations where the sampling techniques and machine learning classifiers are used. This section discusses the findings derived from the results presented in the previous section for the four combinations of experiments.

Looking carefully at Figures 1, 2, 3, and 4, as well as Table 3, one can notice that the results are interesting and could have implications in future applications. First of all, sampling the data can provide good improvements over the original data. Figures 1 and 2, where sampling with replacement is used, show that a 50% reduction in data can improve the generated model in most experimental rounds. On average, both the SVM and Random Forest classifiers improved performance by 5% and 3.4%, respectively, when only 50% of the data is used. In general, it is reasonable to believe that F-measure would be the best across all the experiments when 100% of the data is used to generate the model. However, this is not always true in our case, as noted in the combination 1 and combination 2 experiments, which add more value to the approach of sampling big data. It is clear, though, that the performance in most rounds beats the performance when the whole dataset is used.

Therefore, the reduced use of processing resources and the improved performance of the model is a very good goal that has been achieved in these experiments. In this case, not only was the size of the data reduced, but the performance and accuracy of the model were improved, which has a two-fold benefit. The first benefit is that reducing the amount of processed data can save RAM and CPU processing resources, which enables the processing of such data to be performed by less capable. The second benefit is that although only 50% of data was only used, the performance increased in most test rounds, which means that it is not necessary to include all the instances of data when creating the model. This could be because of the redundancy of some of the data instances or because of data instances that added no value during model creation.

On the contrary, Figures 3 and 4, where reservoir sampling is used, show that a 50% reduction in data affects the performance in almost all rounds. On average, both the SVM and random forest classifiers had decreased performance by 3.3% and 3.2%, respectively, when 50% of the data was used. Basically, it is expected that F-measure for 100% of the data would be better than F-measure for a 50% sample of data since all instances of data are used to create the model. In the combination 3 and combination 4 experiments, the performance in most rounds fell below the performance when the whole dataset was used.

However, the use of less processing resources with only slightly decreased performance is still a worthwhile trade-off. In this scenario, although the performance was slightly decreased, there were significant savings in required processing resources such as memory, as well as the time needed for processing. Consequently, the sampled data can be processed by less capable computers. Undoubtedly, reducing the amount of processed data while also increasing the performance and accuracy of the model is the optimal goal, which was achieved in combination 1 and 2 experiments, does not mean having them all or losing them all. In combinations 3 and 4, one important improvement was achieved, which was saving processing resources.

One additional observation worth pointing out is that sampling with a replacement technique outperformed the reservoir sampling technique in this study. So, it can be said that the sampling technique used for a particular application does have an effect on the results with dealing with big data. Figure 5 shows that the best results in this study were achieved in combination 1, where sampling with replacement and SVM was used, followed by combination 2, where sampling with replacement and random forest were used. These results cannot be deemed conclusive but are indicative of the potential random sampling has when dealing with big data. Additional studies in this regard are needed to further expand on the findings of our research.

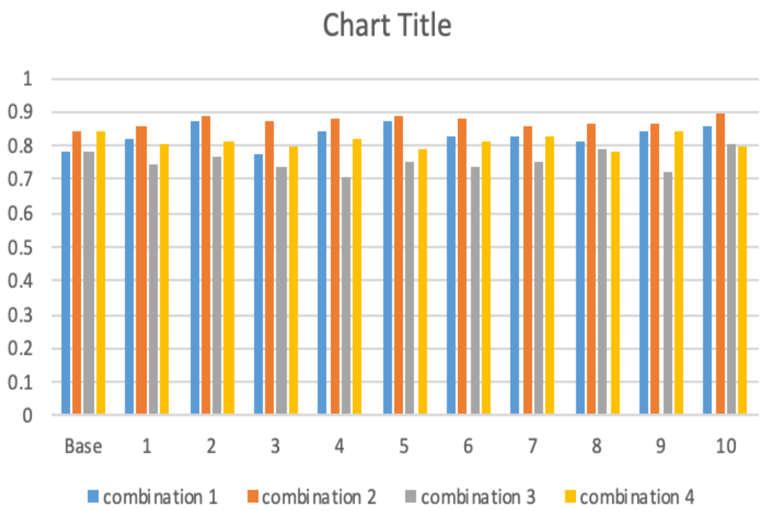

Fig. 6: Results of Different Combinations of Big Data Sampling Using SVM and Random Forest.

Table 3: Results of Four Combinations of Sampling Techniques and Machine Learning Classifiers

Sampling	With Replacement		Reservoir	
Classifier Round	SVM	Random Forest	SVM	Random Forest
Base	0.784	0.841	0.784	0.841
1	0.819	0.858	0.745	0.807
2	0.872	0.888	0.771	0.815
3	0.775	0.875	0.737	0.8
4	0.842	0.88	0.705	0.818
5	0.873	0.886	0.749	0.791
6	0.825	0.882	0.735	0.814
7	0.824	0.861	0.75	0.826
8	0.811	0.866	0.787	0.78
9	0.845	0.864	0.72	0.84
10	0.855	0.893	0.807	0.799
Average	0.834	0.875	0.751	0.809

6. Conclusion and future work

This paper continued our previous works related to the sampling and processing of big data. In this paper, two different random sampling techniques were used, as well as two different machine learning classifiers. Our research used 40 GB of image data in experiments consisting of four different combinations of sampling techniques and machine learning classifiers to find the most suitable process for analyzing big data. The results, at least in this study, showed that a random sampling combination of replacement and SVM classifier was better than any other sampling combinations in terms of model accuracy and performance. Although the results in this study are indicative rather than conclusive, it showed how the sampling technique used could affect the process of model creation. The results also rely heavily on the type of machine learning classifier used. Therefore, sampling and processing big data depends on different factors that can have an impact on the resulting accuracy and performance of the produced model.

In the future, further investigation in this direction will need to be performed. Different types of data will be used to explore whether textual data, for example, has an impact on model performance. Additionally, the experiments can be extended to cover other sampling techniques and machine learning classifiers in order to explore other processing options to find the best combination for any given application.

References

[1] Madden, S. (2012). From databases to big data. IEEE Internet Computing, 16(3), 4-6. https://doi.org/10.1109/MIC.2012.50.

[2] Akhgar, B., Saathoff, G. B., Arabnia, H. R., Hill, R., Staniforth, A., & Bayerl, P. S. (2015). Application of big data for national security: a practitioner’s guide to emerging technologies. Butterworth-Heinemann.

[3] Albattah, W., & Khan, R. U. (2018). Processing Sampled Big Data. *INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS*, *9*(8), 350-356. https://doi.org/10.14569/IJACSA.2018.090846.

[4] W. Albattah, ―The Role of Sampling in Big Data Analysis, in Proceedings of the International Conference on Big Data and Advanced Wireless Technologies - BDAW ‘16, 2016, pp. 1–5. https://doi.org/10.1145/3010089.3010113.

[5] M. Hilbert, ―Big Data for Development: A Review of Promises and Challenges, ‖ Dev. Policy Rev., vol. 34, no. 1, pp. 135–174, Jan. 2016. https://doi.org/10.1111/dpr.12142.

[6] D. A. Reed and J. Dongarra, ― “Exascale computing and big data”, Commun. ACM, vol. 58, no. 7, pp. 56–68, 2015. https://doi.org/10.1145/2699414.

[7] A. L‘Heureux, K. Grolinger, H. F. Elyamany, and M. A. M. Capretz, ―Machine Learning With Big Data: Challenges and Approaches, IEEE Access, vol. 5, no. 1, pp. 7776–7797, 2017. https://doi.org/10.1109/ACCESS.2017.2696365.

[8] K. Singh, S. C. Guntuku, A. Thakur, and C. Hota, ―Big Data Analytics framework for Peer-to-Peer Botnet detection using Random Forests, Inf. Sci. (Ny)., vol. 278, pp. 488–497, 2014. https://doi.org/10.1016/j.ins.2014.03.066.

[9] R. Clarke, ―Big data, big risks, Inf. Syst. J., vol. 26, no. 1, pp. 77–90, Jan. 2016. https://doi.org/10.1111/isj.12088.

[10] D. Sullivan, ―Introduction to big data security analytics in the enterprise. [Online]. Available: https://searchsecurity.techtarget.com/feature/Introduction-to-big-datasecurity-analytics-in-the-enterprise. [Accessed: 31-Jul-2018].

[11] C.-W. Tsai, C.-F. Lai, H.-C. Chao, and A. V. Vasilakos, ―Big data analytics: a survey, J. Big Data, vol. 2, no. 1, p. 21, Dec. 2015. https://doi.org/10.1186/s40537-015-0030-3.

[12] G. Bello-Orgaz, J. J. Jung, and D. Camacho, ―Social big data: Recent achievements and new challenges, ‖ Inf. Fusion, vol. 28, pp. 45–59, Mar. 2016. https://doi.org/10.1016/j.inffus.2015.08.005.

[13] J. Zakir, T. Seymour, and K. Berg, ―Big Data Analytics, Issues Inf. Syst., vol. 16, no. 2, pp. 81–90, 2015.

[14] U. Sivarajah, M. M. Kamal, Z. Irani, and V. Weerakkody, ―Critical analysis of Big Data challenges and analytical methods, J. Bus. Res., vol. 70, pp. 263–286, Jan. 2017. https://doi.org/10.1016/j.jbusres.2016.08.001.

[15] K. Engemann et al., ―Limited sampling hampers ‗big data estimation of species richness in a tropical biodiversity hotspot., Ecol. Evol., vol. 5, no. 3, pp. 807–820, 2015. https://doi.org/10.1002/ece3.1405.

[16] J. K. Kim and Z. Wang, ―Sampling techniques for big data analysis in finite population inference, Jan. 2018. https://doi.org/10.1111/insr.12290.

[17] S. Liu, R. She, and P. Fan, ―How Many Samples Required in Big Data Collection: A Differential Message Importance Measure, Jan. 2018.

[18] J. Bierkens, P. Fearnhead, and G. Roberts, ―The Zig-Zag Process and Super-Efficient Sampling for Bayesian Analysis of Big Data, Jul. 2016.

[19] J. Zhao, J. Sun, Y. Zhai, Y. Ding, C. Wu, and M. Hu, ―A Novel Clustering-Based Sampling Approach for Minimum Sample Set in Big Data Environment, Int. J. Pattern Recognit. Artif. Intell., vol. 32, no. 2, pp. 1–10, Feb. 2018. https://doi.org/10.1142/S0218001418500039.

[20] L. Zhou, S. Pan, J. Wang, and A. V. Vasilakos, ―Machine learning on big data: Opportunities and challenges, Neurocomputing, vol. 237, no. 1, pp. 350–361, 2017. https://doi.org/10.1016/j.neucom.2017.01.026.

[21] Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992, July). A training algorithm for optimal margin classifiers. In Proceedings of the fifth annual workshop on Computational learning theory (pp. 144-152). ACM. https://doi.org/10.1145/130385.130401.

[22] Tong, S., & Koller, D. (2001). Support vector machine active learning with applications to text classification. Journal of machine learning research, 2(Nov), 45-66. https://doi.org/10.1145/500156.500159.

[23] Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. R news, 2(3), 18-22.

[24] Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32. https://doi.org/10.1023/A:1010933404324.

[25] NPDI Pornography Database, (2013), https://sites.google.com/site/pornographydatabase/.