CHALLENGES IN HANDLING IMBALANCED BIG DATA: A SURVEY

B.S.Mounika Yadav[1], Sesha Bhargavi Velagaleti[2]
[1]Asst. Professor, IT Dept., Vasavi College of Engineering
[2]Asst. Professor, IT Dept., G.Narayanamma Institute of Technology and Science

Abstract

Big Data describes enormous sets that have more divergent and intricate structure like weblogs, social media, email, sensors, and photographs. These unstructured data and peculiar characteristics from traditional databases typically associated with extra complications in storing, analyzing and applying further procedures or extracting results. Big Data analytics is the process of auditing gigantic amounts of complex data to find out unseen patterns or recognizing hidden correlations. Big Data applications are rising during the last years, and researchers from many disciplines are aware of the advantages related to the knowledge extraction from this type of problem. However traditional learning approaches cannot be enforced due to the scalability issues. Being still a recent discipline, handful research has been conducted on imbalanced data classification for Big Data. The apprehension behind this is mainly the difficulties in adapting standard techniques to the Map-Reduce programming style. Additionally, inner problems of imbalanced data, namely lack of data for training, the overlap between classes, the presence of noise and small disjuncts, are emphasized during the data partitioning to fit the Map-Reduce programming style. A literature survey on classification problem in Big Data has been done and existing methodologies were discussed with their pros and cons in this paper. This study suggests that there is a great need for finding a new method of classification when it comes to Big Data which addresses several issues like multi-class problems, class imbalance.

Key words: Classification, Class-imbalance , Big Data

I. INTRODUCTION

The development and sophistication of the information technologies have enabled an exponential growth on the data that is produced, processed, stored, shared, analyzed and visualized. According to IBM in 2012, every day 1.5 quintillion bytes of data is created. Internet users generate 2.5 quintillion bytes of data each day, on average, according to recent research cites by Domo. Big data comprehend a collection of datasets whose size and complexity challenges the standard database management systems and defies the application of knowledge extraction techniques. This data comes from a wide range of sources such as sensors, digital pictures, videos, purchase transactions, social media like Facebook and Twitter, etc. This generation and collection of massive datasets has further inspired the analysis and knowledge extraction process with the belief that with more data available, the information that could be derived from it will be more precise. However, the standard algorithms that are used in data mining are not usually able to deal with these massive datasets. In this manner, classification algorithms must be altered and adapted considering the solutions that are being used in big data so that they can be used under these circumstances maintaining its predictive capacity. One of the complexities that make difficult the extraction of useful information from datasets is the problem of classification with imbalanced data. This problem occurs when the number of instances of one class (positive or minority class) is considerably smaller than the number of instances that belong

to the other classes (negative or majority classes). In this situation, the interest of the learning is focused towards the minority class as it is the class that needs to be accurately identified in these problems. Big data is also sensitized by this unseen class distribution.

II. DIFFICULTIES IN CLASSIFYING BIG DATA

With the development of information technologies, organizations have had to face new challenges to analyze vast amounts of information. Thus "Big Data" came into existence, which is applied to all the information that cannot be processed or analyzed using traditional techniques or tools. Big data is commonly characterized using some V's, they are Volume, Velocity, Variety, Veracity, Valence, and value. Volume is the huge amount of data that is created every second, minutes, hour, and day in our digitized world. Variety refers to the ever-increasing different forms that data can come in such as text, images, voice, and geospatial data. Velocity is the speed at which data is generated and the pace at which data navigates from one point to the next. Volume, variety, and velocity are the three main dimensions that characterize big data. And describe its challenges. We have vast amounts of data in varying formats and quality, which must be processed instantly. More V's have been introduced to the big data community as it lead to the discovery of new challenges and ways to illustrate big data. Veracity and valence are two of these additional V's which gains more attention. Veracity refers to the noise and abnormality in data. It is often the unmeasurable uncertainties and trustworthiness of data. Valence refers to the connectedness of big data in the form of graphs, just like atoms.

These data volumes that we call big data are coming from different sources. It can be broadly categorized into three: Machine-generated Data, Human-Generated Data, and Organizationgenerated Data. People generated data is highly unstructured, and thus it is the major challenge in classifying this type of data.

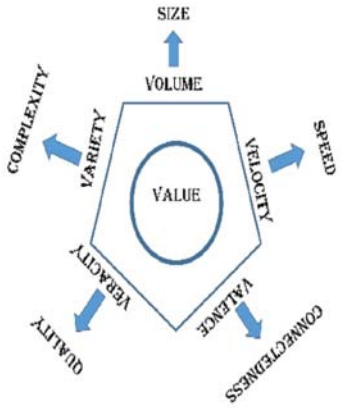

Fig.1: Characteristics of Big-Data

III. PERFORMANCE EVALUATION METRICS OF IMBALANCED BIG DATA

Evaluation measures play a crucial role in both assessing the classification performance and guiding the classifier modeling. Traditionally, accuracy is the most commonly used measure for these purposes. However, for classification with the class imbalance problem, accuracy is no longer a proper measure since the rare class has a very little impact on accuracy as compared to the prevalent class. In the bi-class scenario, one class with very few training samples but high identification importance is referred to as the positive class; the other as the negative class. Samples can be categorized into four groups after a classification process as denoted in the confusion matrix presented in Table 1.

	Predicted as positive	**Predicted as negative**
Actually positive	True positives (TP)	False negatives (FN)
Actually negative	False positive (FP)	True negatives (TN)

Table 1: Confusion Matrix

- **True Positive Rate:** ***TPrate = TP / (TP + FN)***
- **True Negative Rate:** ***TNrate = TN / (TN + FP)*** **False Positive Rate:** ***FPrate = FP / (TN + FP)***
- **False Negative Rate:** ***FNrate = FN / (TP + TN)***
- **Positive Predictive Value:** ***PPvalue=TP /(TP + FP)***
- **Negative Predictive Value:** ***NPvalue = TN / (TN + FN)***

a) F-measure :

When only the performance of the positive class is concerned, two measures are important: True Positive Rate (*TPrate*) and Positive Predictive Value (*PPvalue*). In information retrieval, True Positive Rate is defined as *recall* denoting the percentage of retrieved objects that are relevant:

Recall = TPrate = TP / (TP + FN)

Positive Predictive Value is defined as *precision* denoting the percentage of relevant objects that are identified for retrieval:

Precision = PPvalue = TP / (TP + FP)

F-measure = 2RP / (R + P)

In principle, *F*-measure represents a harmonic mean between recall and precision

F-measure = 2 / (1/R + 1/P)

The harmonic mean of two numbers tends to be closer to the smaller of the two. Hence, a high *F*measure value ensures that both recall and precision are reasonably high.

b) G-mean :

When the performance of both classes is to be considered, both True Positive Rate (*TPrate*) and True Negative Rate (*TNrate*) are expected to be high simultaneously.

G-mean = $\sqrt{TPRATE.\ TNRATE}$

G-mean measures the balanced performance of a learning algorithm between these two classes.

***c) ROC (Receiver Operating Characteristic) Analysis*:**

Each threshold value generates a pair of measurements of (*FPrate*, *TPrate*). By linking these measurements with the False Positive Rate (*FPrate*) on the *X*-axis and the True Positive Rate (*TPrate*) on the *Y* -axis, a ROC graph is plotted. The ideal model is one that obtains 1 True Positive Rate and 0 False Positive Rate (i.e., *TPrate* = 1 and *FPrate* = 0). The area under a ROC curve (AUC) provides a single measure of a classifier's performance for evaluating which model is better on average.

IV. EXISTING METHODOLOGIES

1. Data pre-processing techniques

a) Traditional data-based solutions for Big Data:

Several pre-processing techniques were enforced in a MapReduce workflow [1]. Especially the Random Over Sampling Over Sampling Technique (ROSBigData), Random Under Sampling Technique(RUSBigData) and the SMOTE (SMOTE-BigData) MapReduce Versions. For every technique, each Map process does the job of adjusting the class distribution for their data partition, either by the random duplication of minority class instances (ROS-BigData), the random expulsion of majority class instances (RUS-BigData) or the synthetic data generation carried out by SMOTE (SMOTE-BigData). Then, a Reduce collects the outputs generated by each mapper and randomized them to form the balanced dataset considering the majority voting. The Random Forest implementation from Mahout2 [2,3] was chosen as baseline classifier for the experiments. *Constraints:*

- Pre-processing and classification methods worked locally within each Map, thus limiting the potential of these algorithms.
- Loss of information that comes with removing samples from training data.
- Replication of instances tends to increase computational cost.
- Lack of flexibility.
- SMOTE leads to over generalization.

To avoid these barriers, some approaches are defined such as Borderline SMOTE and Adaptive Synthetic Sampling for generalization.

Evolutionary algorithms and sampling methods are used to deal with the class

imbalance problem. The ensemble methods like AdaBoost, RUSBoost, and SMOTEBoost are coupled with SMOTE to solve imbalanced data problems.

2. Algorithm Based Solutions

a) *Random oversampling with evolutionary feature weighting and random forest (ROSEFW-RF):*

The algorithm, named as ROSEFW-RF [4], was based on several Map-Reduce techniques to (1) balance the classes distribution through random oversampling, (2) detect the most relevant features via an evolutionary feature weighting process and a threshold to choose them, (3) build an appropriate Random Forest model from the pre-processed data and finally (4) classify the test data.

The combination of the instance and feature pre-processing approaches accomplish high-quality results.

Constraint:

- Applying a high-ratio of oversampling requires high training time.

b) Evolutionary Under Sampling

Regarding under sampling approaches, in [5] authors developed a parallel model to enable evolutionary under sampling methods under the Map-Reduce scheme. Precisely, model consisted of two Map-Reduce procedures. The first Map-

Reduce task builds a decision tree in each map after performing evolutionary under sampling preprocessing. Then, a second Map-Reduce job is inducted to classify the test set. The evolutionary under sampling step is further boosted by adding a windowing scheme adapted to the imbalanced scenario. The experiment was carried with decision tree on KDDcup'99 dataset. The results were better regarding accuracy and efficiency.

Constraint:

- Loss of some important information while under sampling.

c) NRSBoundary-SMOTE

Here in [6], authors proposed a method where it consists of two Map-Reduce procedures. The first Map-Reduce job divided the training set according to neighborhood relation and, it generated three subsets as output, called Positive, Minority and Boundary. The Positive subset contained the majority class samples where its neighbors have the sample class label, the Minority subset contained the minority samples, and the Boundary subset contained the minority samples that have any majority class sample in its neighbors. In the second Map-Reduce job, every map gets a data block of the Boundary set, and it computed for each sample in its partition the k nearest neighbors. Then, the reduce process selected for each sample one of its neighbors randomly to interpolate with it. If the new synthetic sample belonged to the neighbor of samples that in Positive, another neighbor was selected from the list. Otherwise, the synthetic example was generated. In both Map-Reduce processes, the Positive and Minority sets were added to the Hadoop Distributed Cache.

Constraint:

- Focused on only Two-class imbalance.

d) Extreme Learning machine resampling:

Map-Reduce approach based on ensemble learning and data resampling were developed. This algorithm [7], consists of four stages: (1) alternately over-sample p times between positive class instances and negative class instances; (2) construct l balanced data subsets based on the generated positive class instances; (3) train l component classifiers with extreme learning machine algorithm on the constructed l balanced data subsets; (4) integrate the l ELM classifiers with simple voting approach.

Constraint:

- Computationally expensive because of the iterative oversampling process applied in the first stage.

3. Cost-Sensitive Learning Studies

a) Instance weighting SVM:

In [8], a method is proposed which combines an instance weighted variant of the SVM with a Parallel Meta-learning algorithm using MapReduce. Specifically, a symmetric weight-boosting method was developed to optimize the instanceweighted SVM. In the Map-Reduce design, each Map process applies a sequential Instance Boosting SVM algorithm in the examples of its partition and generates a base learner. Then, the models generated by all Maps form an ensemble of classifiers. Therefore, no Reduce step is used as no fusion of the models was required.

Constraints:

- This Map-Reduce scheme is the iterative process that is performed in each Map task which leads to overhead.
- Also, datasets used in the experiments were not more than half a million instances, so it is difficult to decide whether this approach can be scalable for real Big Data problems.

b) Cost-sensitive random forest:

Random forest is the popular ensemble learning method that is used in classification. To deal with imbalance big data original RF should be modified so that it can effectively deal with the scalability issues of big data. In [9], authors divided the entire RF into two processes. The first process was the creation of the model where each map task was responsible to build a subset of the forest with the data block of its partition and generated a file containing the built trees. Then, the second MapReduce process was initiated to estimate the class associated with a data test set. In this process, each map estimated the class for the examples available in its partition using the previously learned model, and then the predictions generated by each map were concatenated to form the final predictions file.

Constraint:

- Random Forest depends on the type of problem and the influence of the lack of density over the specific approach.

c) Cost-sensitive fuzzy rule-based classification system (FRBCS)

In [10] authors proposed a technique Chi-FRBCSBig Data, a Map-Reduce implementation of an FRBCS which was developed earlier[11], to address imbalanced Big Data. The Chi-FRBCS BigDataCS algorithm consisted of two MapReduce processes: the first Map-Reduce process, each Map process builds a rule base using only the data present in its partition, then, the Reduce process collects and combines the rule bases produced by each map task to form the final rule base.

When the first Map-Reduce process devoted to the building of the model had finished, the second Map-Reduce process was initiated. In this process, each map task estimated the class for the examples included in its data partition using the previously learned model, then, the predictions generated by each map were aggregated to confirm the final predictions file. The classification job did not include a reduce step. The experimental study showed that the proposal could handle imbalanced Big Data obtaining best results regarding computation time and classification performance.

Constraint:

- The synergy between both strategies alleviates some intrinsic data problems, like the small sample size problem, which are induced because of the way the learning is done.

V. CONCLUSION

Despite the various advantages of Big Data regarding storing processing and retrieval, still there are many issues left unaddressed due to the complexity of all the V's of Big Data. Even though many existing methodologies focused on issues like providing cost sensitive solutions, over and under sampling mechanisms, fuzzy-logic based classification etc., still classification and clustering of Big Data is a major research challenge. Our paper mainly focusses on studying various existing algorithms for classification of Big Data and hence to analyze their constraints which are to be addressed if a new method is to be introduced.

REFERENCES

1. Río S, López V, Benítez J, Herrera F (2014) On the use of MapReduce for imbalanced Big Data using random forest. Inf Sci 285:112–137
2. Owen S, Anil R, Dunning T, Friedman E (2011) Mahout in action, 1st edn. Manning Publications Co., Greenwich
3. Lyubimov D, Palumbo A (2016) ApacheMahout: beyond MapReduce, 1st edn. CreateSpace Independent, North Charleston
4. Triguero I, Río S, López V, Bacardit J, Benítez JM, Herrera F (2015) ROSEFWRF: the winner algorithm for the CBDL'14 Big Data competition: an extremely imbalanced Big Data bioinformatics problem. Knowl Based Syst 87:69–79
5. T.riguero I, GalarM, Vluymans S, Cornelis C, Bustince H, Herrera F, Saeys Y (2015)

Evolutionary undersampling for imbalanced Big Data classification. In: IEEE congress on evolutionary computation (CEC), pp 715–722.

6. Hu F, Li H, Lou H, Dai J (2014) A parallel oversampling algorithm based on NRSBoundary-SMOTE. J Inf Comput Sci 11(13):4655–
7. Zhai J, Zhang S, Wang C (2015) The classification of imbalanced large data sets based on MapReduce and ensemble of elm classifiers. Int J Mach Learn Cybern. doi:10.1007/s13042-015-0478-
8. Wang X, Liu X, Matwin S (2014) A distributed instance-weighted SVM algorithm on large-scale imbalanced datasets. In: Proceedings of the 2014 IEEE international conference on Big Data, 2014, pp 45–51.
9. Río S, López V, Benítez J, Herrera F (2014) On the use of MapReduce for imbalanced Big Data using random forest. Inf Sci 285:112–137
10. López V, Río S, Benítez JM, Herrera F (2015) Cost-sensitive linguistic fuzzy rule based classification systems under the MapReduce framework for imbalanced Big Data. Fuzzy Sets Syst 258:5-38.
11. Río S, López V, Benítez JM, Herrera F (2015) A MapReduce approach to address Big Data classification problems based on the fusion of linguistic fuzzy rules. Int J Comput Intell Syst 8(3):422–437