ISBN 978-981-14-4787-7
Proceedings of 2020 the 10th International Workshop on Computer Science and Engineering (WCSE 2020)
Yangon (Rangoon), Myanmar (Burma), February 26- February 28, 2020, pp. 22-26
doi: 10.18178/wcse.2020.02.005

Real-time Big Data Analytics for Feature Selection on Apache Spark

Lwin May Thant[+], Sabai Phyu

University of Computer Studies, Yangon, Myanmar

Abstract. Real-time data analysis is a key research in many domains. It can be applied to pre-existing or prescriptive models. The effective result is that monitor the account and review on a real-time action. Apache spark machine learning library Mllib can be distinct display place for real-time assessment foe extracting, transforming and selecting features and classification, clustering and frequent pattern mining. Feature selection is the detection in a group of feature what are the most relevant and removing the redundant data. Specifically, we made using the Apache spark tool and analyze the streaming time-series data using Mllib to extract the high qualitative feature in efficiently to get qualitative and high performance model.

Keywords: feature selection, apache spark, filter method, real-time data

1. Introduction

Nowadays, text data processing on large-scale is quickly important for many research area and business domains for real-time analytics. Data discovery and analytics using the model are basically in machine learning with real-time data. In an analysis on historical data is big-time to build the machine learning mode. In analytics phase, predict the model on live events. Apache spark platform are implemented several models for parallel and distributed processing on multiple machines. Moreover, spark Mllib is the implementation of machine learning framework to the distributed memory-based Spark architecture. It is platform independent and open-source libraries for big data implementation for distributed architecture and automatic data parallelization. Mllib can work a variety of machine learning functionalists such as extracting, transforming and selecting the features and classification, clustering and frequent pattern mining.

In those function, we implemented to extracting, transforming and selecting features on real-time data. This is the reason to reduce the computational cost of modeling and to improve the performance of the model. In this paper, three filters methods are used on high-dimensional classification with real-time data sets. We search the high accuracy methods with low run time. The remainder of this paper is organized as follows: Section 2, explain the related works. In section 3, we discuss the background knowledge for this paper and the three filter methods. Section 4, describe the real time streaming framework. Section 5 explain the experiments to compare the filter methods and analyze the results. In section 6, conclusion of our work.

2. Related Work

Many faster filter methods based on information theory, especially mutual information and SVM feature weights to mathematically evaluate the relevance and redundancy of data, optimizing their implementation through efficient parallelization is also crucial for challenge ultrahigh dimensional issued in big data [1, 2]. Author in [3], Evolutionary computation work with a filter feature selection algorithm. To obtain subsets of features from big data use the MapReduce archetype. To break down the original datasets into blocks of instances and learn from them a final vector of feature weights. The algorithm is implemented on the Spark framework and experiment show that increasing classification accuracy and runtime with big data. In [4],

[+] Corresponding author. Tel.: +09783788109
E-mail address: lwinmaythant@ucsy.edu.mm

network traffic feature selection on Spark with FSMS method. It is based on Fisher score and employ for subsets with a sequential featured search. On the Spark framework, this method decrease the classification and modelling time.

The work in [5], proposes the use case of X2 feature selection which is very popular in supervised learning pipeline. It is implemented the algorithm of the Scikit-learn Python machine learning library. The experiments run over the Data bricks platform and show that partitioning scheme of the data. Most of the feature selection algorithms depend on the number of features or instances, ReliefF depends linearly on both of them [6]. In [7] introduce the new general definition of L1-norm SVM (GL1-SVM) for feature selection and prove that solving the new proposed optimization problem reduces error penalty and enlarges the margin between two support vector hyper-planes. When facing with high dimensional instances, it can perform exactly well. In this paper, we focus the three methods for feature selection with real-time data on Spark framework. We evaluate the performance of those methods with respect to the classification accuracy and run time.

3. Background Theory

3.1. Feature Selection

Some of feature selection methods describe in this section. In filtering of feature selection methods, we will consider two kinds of methods upon a variable types: numerical and categorical and also two main groups of variables: input and output. In feature selection, input variables are typically provided as input to a model and indicates a classification predictive modelling problem. The statistical for filter-based feature selection are performed one input variable at a time with the target variable. These stasticial measures that any interaction between input variables is not considered in the filter process. In figure1, we demonstrate the feature selection methods based on input variable.

In feature selection methods can be classified into three categories according to their relationship with the classification algorithms: Filter, Wrapper and Embedded [8, 9]. In the wrapper method, the “usefulness” of a subset of a features is evaluated on the basis of the classifier performance [8, 9]. Embedded method exploit intrinsic characteristics of a given model to guide the feature selection process and choose feature which best contribute to the accuracy performance of the model [8, 9].

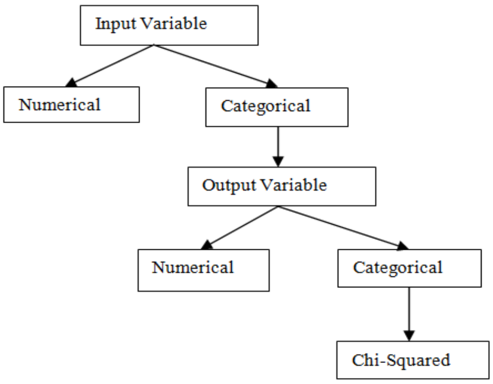

Fig. 1: Categorical flow of feature selection methods based on input variable

3.2. Feature Selection on Big Data

For a real time data sets, data collection, maintain, transmitting and processing within time interval are very difficult. So, very optimal technologies are required for processing the data without any fault and small amount of time interval. In big data research area, extracting require information from collections of data is one of the important fact. Moreover, efficient algorithms are need to apply to extract information in vast amount of data. In those day, classical big data analytics are special important problem to explore the data. Gartner [10] referred to big data in terms of volume, velocity, and variety, that is, the 3Vs, to which a further 2Vs were subsequently added, namely, veracity and value. Data analytics are interested in big data area, the large amount of instances while focusing to the feature aspect. To get more effective learning and prediction model, feature selection methods are needed for big data processing.

Removing of irrelevant, redundant and noisy features from complex features is the key research area in big data. For large-scale data, new feature selection techniques are very essential to obtain optimal information. We will discuss the feature selection framework for big data, this is the main challenges made to develop the classical approach to the new big data research trend. We also analyze the complexity arising from parallelization of the operations. We will consider the following phase while maintaining certain features.

Phase 1: Columns Transformation - The access pattern selection is variety, which operate on rows or columns. Although this may be considered, it can be decrease performance when compute relevance and redundancy in feature selection method. For distributed framework like Spark, this issue is conspicuously important when the data partition is a big effect on performance.

Phase 2: Broadcasting - Be minimal to avoid CPU usage, all features values have been grouped and partitioned into different partitions. By replicating the output feature and the final selected feature, data shift is reduced in each iteration.

Phase 3: Precomputed Data Caching - Subsequent marginal and join proportions are also performed. So, redundancy computation by features are reduced.

Phase 4: Greedy Approach - This phase reduce the common of complexity in feature selection by selecting the number of features. Greedy search is selected only one feature in each iteration.

3.3. Apache Spark on Big Data

Apache Spark framework for big data analytics is in-memory programming model and libraries for scalable machine learning, graph analysis, data streaming and structured and unstructured data processing. It is a cluster computing framework and open source project, with an increasing development in both academia and industry especially who are beginners in this area.

Extraction of the right features is one of the most challenging tasks in big data processing. Although solution of this task is Spark Mllib provide different methods for feature selection. While processing of feature extraction is to extract features from raw data feature transformers can be used for scaling, normalization, converting features, modifying features and so on. The machine learning library of Apache Spark contains methods for selecting subsets of features from larger sets of features In figure 2 show the framework for feature selection with Spark in big data.

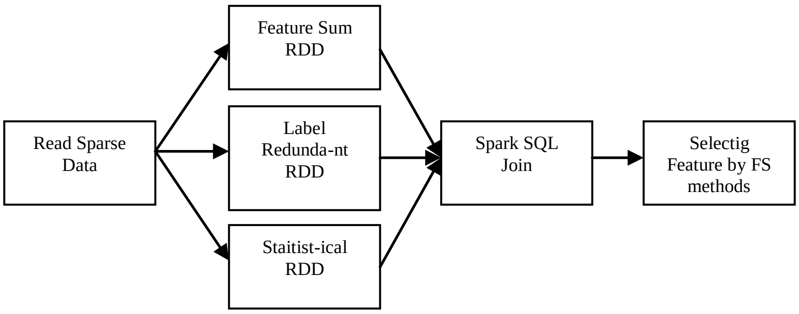

Fig. 2: Feature Selection Framework with Spark

3.4. Featuring Selection Methods

3.4.1. Kruskal-test

Filter Kruskal-test is a non-parametric method which checks whether the samples originate from the same distribution or not. In the first step, the features are sorted in ascending order and then rank the sorted data point and if any tied values present then give an average rank. Finally, calculate the filter score for feature X_k.

3.4.2. Chi-squared Test

Chi-squared test X^2 is the prominent statistical test and perform independent between the observed and expected distribution of a feature. The chi-squared use the value of X^2 as score. The value of X^2 statistical is directly proportional to the dependency between the class variable and feature.

3.4.3. Relief

Relief is the analysis of the quality of attribute in their values distinguish between instances. The Relief search the two nearest neighbors: one for within one class and another for different class. This method select randomly and regarded scored as x is the sum of weighted differences within same class or different class. In above mention methods are used in this paper and we will extract with optimal features with best method.

4. Real Time Streaming Big Data with Spark

The streaming data contains a wide variety of thousands of data sources. The real time streaming framework are working with these flow. In the data source layer, data are collecting from file system, cloud bucket databases and real time stream from IoT devices. Apache Nifi is a data collection tool that allows to send, receive, route, transform and sort in an automatic way in ingestion layer. In this layer, data are transformed by predefined processors. For ingestion of real time data, Apache Kafka is used and this is messaging system. Apache Kafka allow the partition to parallelize by splitting the data. Each partition are work on a separate machine in parallel. Apache Spark framework is the structured streaming model to provide fast, scalable, fault tolerant low latency. It is perform batch and streaming processing. It is provided in transformation layer.

When reading streaming data from Kafka through Apache Spark, Kafka send a data frame. Kafka cluster specify the location of data frame which are reading the data frame. In spark structure wait for 1 second and batches all the events during the time between micro batches. When finishing the micro batch processing, new batch is received and schedule again. However, latency does not decrease in streaming execution. Processing flow are shown in Figure3.

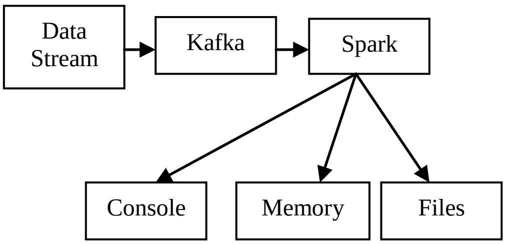

Fig. 3: Spark Flow for real time streaming

Table 1: Summary Dataset

Dataset	Size	Rows	Features
1	1.3 GB	4,203,855	11,556,704
2	2.67 GB	8,407,733	20,215,650

Table 2: Comparison of classification performance on Dataset 1

Comparison of feature selection methods	Support Vector Machine	K Nearest neighbor	Na ïve Bayes	Random Forest
Kruskal-test	0.80	0.83	0.80	0.82
Chi-squared test	0.78	0.81	0.82	0.85
Relief	0.73	0.69	0.71	0.79

Table 3: Comparison of classification performance on Dataset 2

Comparison of feature selection methods	Support Vector Machine	K Nearest neighbor	Na ïve Bayes	Random Forest
Kruskal-test	0.91	0.92	0.93	0.95
Chi-squared test	0.87	0.90	0.91	0.92
Relief	0.87	0.77	0.86	0.85

5. Experiment and Result

Evaluation of feature selection methods can be compared in many ways. We choose the way is to classify the performance of features by selected methods. When featuring the significant instance correctly with featuring methods, we will classify the features with perfect performance with small amount of features. If we will know the exactly features, we will calculate the feature selection methods effectively. But, real time data are not static and thus we calculate the data with unknown various features. Since the exactly features are reported the real time data, it can be applied the different feature selection methods. Over a various separate set, we used the feature selection for training set and classification for validation set. So, featuring selection methods are used to obtain the valid feature set in training feature. Classification performance are calculated on the validation set for feature selection methods. To measure the classification performance, AUC values are used and increase the AUC values we will better classified.

In our implementation, we used the Apache Spark Mllib for each classification algorithms and summary of data set shown in Table 1. Table 2 shows the data set 1 of maximum value for the combination of feature selection methods and classification techniques. For dataset2, we compared the four classification performance for features. 10- Folds cross validation method are used for calculation of AUC values and 9 folds are feature selection and training. For the combination of selection methods and classification algorithms repeating the process 10 times over the ten distinct cases. Table 2 show the combination of selection methods and classification algorithms of maximum AUC values. We can be said what selection method is better performance in feature selection and which is low run time.

6. Conclusion

In this work, a filter-based feature selection method has been applied to real time streaming data. We calculated performance on two data sets: one from the news twitter and the other from Amazon. In this paper, on both data sets were compared with filter-based feature selection methods and classification performance in selecting features for different classification algorithms. We can trained a model with various feature selection methods and classifiers what methods are better performance on selecting features. Finally, in evaluating the AUC values we analyzed their performance. Thus we can conclude that Kruskal-test is a suitable technique for feature selection on streaming data.

7. References

[1] "A Large Scale Filter Method for Feature Selection Based on Spark", 2017 4th IEEE International Conference on Soft Computing and Machine Intelligence.

[2] Jazeera K.U. and Julie M. David. Issues, challenges, and solutions: big data mining. Sixth International Conference on Networks & Communications. DOI: 10.5121/csit.2014.41311.

[3] D Peralta, S Del R ό, S Ram ŕez-Gallego, I Triguero, Jose M. B F Herrera. Evolutionary Feature Selection for Big Data Classification: A MapReduce Approach. Mathematical Problems in Engineering Volume 2015 (2015)

[4] Yong Wang, Wenlong Ke, Xiaoling Tao.A Feature Selection Method for Large-Scale Network Traffic Classification Based on Spark. Information (2078-2489). 2016, Vol. 7 Issue 1, p1-11. 11p.

[5] M Nassar, H Sofa, A Al Mutawa, A Helal, Iskander GABA "Chi-squared Feature Selection over Apache Spark M Mandal, A Mukhopadhyay.

[6] An Improved Minimum Redundancy Maximum Relevance Approach for Feature Selection in Gene Expression Data. Jul 2016.

[7] Hai Thanh Nguyen, Katrin Franke, and Slobodan Petrovi'c, "On General Definition of L1-norm Support Vector Machines for Feature Selection," International Journal of Machine Learning and Computing vol. 1, no. 3, pp. 279 283, 2011.

[8] C Liu, W Wang, Q Zhao and M Konan. A new feature selection method based on a validity index of feature subset. Pattern Recognition Letters, Volume 92, 1 June 2017, Pages 1- 8.

[9] S Ramırez-Gallego, H Mourino-Talın, Francisco Herrera. An Information Theory Based Feature Selection Framework for Big Data under Apache Spark. Journal of latex class files, vol. 13, no. 9, September 2014.

[10] Y-W Chang, C-J Lin. Feature ranking using linear SVM. Proceedings of the Workshop on the Causation and Prediction Challenge at WCCI 2008, PMLR 3:53-64, 2008.