Finding Electric Energy Consumption Patterns in Big Time Series Data

R. Perez-Chacon, R. L. Talavera-Llames, F. Martinez-Alvarez and A. Troncoso

Abstract In recent years the available volume of information has grown considerably due to the development of new technologies such as the sensor networks or smart meters, and therefore, new algorithms able to deal with big data are necessary. In this work the distributed version of the k-means algorithm in the Apache Spark framework is proposed in order to find patterns from a big time series. Results corresponding to the electricity consumptions for years 2011, 2012 and 2013 for two buildings from a public university are presented and discussed. Finally, the performance of the proposed methodology in relation to the computational time is compared with that of Weka as benchmarking.

Keywords Big data · Time series · Patterns · Clustering

1 Introduction

Rapid and huge data storage is in frequent use nowadays. This new scenario causes extreme difficulties to both efficiently process and store such big amount of data [6]. In this context, much effort is being devoted to enhance existing data mining techniques in order to process, manage and discover knowledge from this big data [13].

The limitations of the MapReduce paradigm [3] for iterative algorithms development have led to new paradigms, such as Apache Spark [8], which is an open source software project. Among its most important capacities, multi-pass computations, high-level operators, diverse languages usage, in addition to its own language called Scala, are most notable. Moreover, a machine learning library, MLlib [7], is also integrated within the framework.

R. Perez-Chacon · R.L. Talavera-Llames · F. Martinez-Alvarez · A. Troncoso(✉)
Division of Computer Science, Universidad Pablo de Olavide, 41013 Seville, Spain
e-mail: {rpercha,rltallla,fmaralv,ali}@upo.es

S. Omatu et al. (eds.), *DCAI, 13th International Conference*,
Advances in Intelligent Systems and Computing 474,
DOI: 10.1007/978-3-319-40162-1_25

The objective of this work is the discovery of patterns in big time series of electricity consumption. Given its size, the collected data cannot be processed with classical machine learning approaches. Therefore, implementations for distributed computing must be used and, in particular, a distributed methodology based on the, still relatively unknown, parallelized version of k-means++ is proposed. This methodology has been developed by using MLlib in the framework Apache Spark, under the Scala programming language. Real–world big data sets collected from a sensor network located in several buildings of Pablo de Olavide University have been analyzed. The successful analysis of these patterns is expected to be used for efficient management of the university electricity resources, as well as for characterizing the electricity consumption over time.

Increased attention has been paid to big data clustering in recent years. A survey on this topic can be found in [5]. Specifically, several approaches have been recently proposed to apply clustering to big time series data. Namely, in [4] the authors propose a new clustering algorithm based on a previous clustering of a sample of the input data. The dynamic time warping was tested to measure the similarity between big time series in [14]. In [16] a data processing based on mapreduce was used to obtain clusters. A distributed method for the initialization of the k-means is proposed in [2]. However, there is still much research to be conducted in this field, especially considering that very few works have been published.

The study of electricity profiles by means of clustering techniques for small and medium datasets has been studied in the literature. In [15] a methodology based on the visualization to obtain the clusters is provided. In [12] the authors examined Spanish electricity prices, discovering some associated patterns to different days and to different seasons. The study was performed by applying crisp clustering, in contrast to the study carried out in [11], where fuzzy clustering was also shown to be useful in this context.

Clustering consumption data was also the goal in [10] but, this time, the authors went one step further and used this information as input for consumption forecasting.

Later in 2012, classification and clustering of electricity demand patterns in industrial parks was addressed [9]. In this work, a data processing system to analyze energy consumption patterns in Spanish parks, based on the cascade application of a Self-Organizing Maps and the k-means algorithm, was introduced.

As for the particular case of clustering big time series consumption data, there is no study carried out so far, to the best of the authors' knowledge. And this is precisely the reason why this study is presented.

The remainder of the paper is structured as follows. In Section 2 the proposed method to discover patterns in time series is described. Section 3 presents the experimental results corresponding to the clustering of the energy consumption coming from a sensor network of building facilities. Finally, Section 4 summarizes the main conclusions drawn from this study.

2 Methodology

This section describes the methodology proposed in order to find consumption patterns in electricity-related big time series data. In particular, the k-means algorithm, included in MLlib, is used in a Spark context to obtain clusters that define consumption patterns.

Figure 1 shows the key steps of the proposed methodology to obtain patterns as a result of the clustering. The first phase consists of data cleansing and the transformations carried out over a RDD variable of Spark, in order to use it in a distributed way. The dataset is the electricity consumption time series from two buildings from a public university for every fifteen minutes of the years 2011, 2012 and 2013. Each row of the dataset contains the following information: building name, date (split into five fields) and the electricity consumption data. Nevertheless, some of these rows contains accumulated consumption power data due to existing missing values, which were successfully preprocessed to learn correctly the models in the next phase. The preprocess stage is properly detailed in the Section 3.

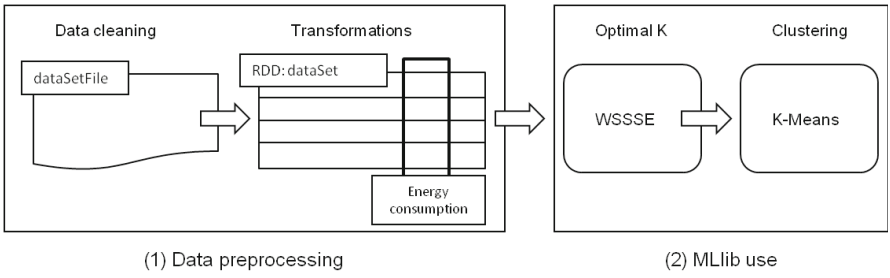

Fig. 1 Illustration of the proposed methodology.

Once the RDD dataset is created, it is necessary to group it in rows of 96 values (4 values per hour per 24 hours of a day) and reduce the dimension of the original RDD dataset before creating a model. This reduction consists in removing the building name field, and transforming the date into a numerical index. Thus, each row finally contains the 96 values corresponding to electricity consumption for a given day.

The second phase consists in the use of MLlib. Firstly, it is necessary to obtain an optimal number of clusters k, which will be used as an input parameter in the k-means algorithm. For that, the Within Set Sum of Squared Errors (WSSSE) index, defined by the sum of the squared Euclidean distance between the elements of a cluster and its centroid for all clusters and instances of the big data, is computed when applying the k-means for a certain number of clusters. In fact, the optimal k is usually the one which is a local minimum in the WSSSE graph.

MLlib includes a parallelized version of k-means++ [1], called k-means||, that it is used in this work to obtain the resulting models. The k-means|| algorithm runs the k-means algorithm a number of times in a concurrent way, returning the best clustering result.

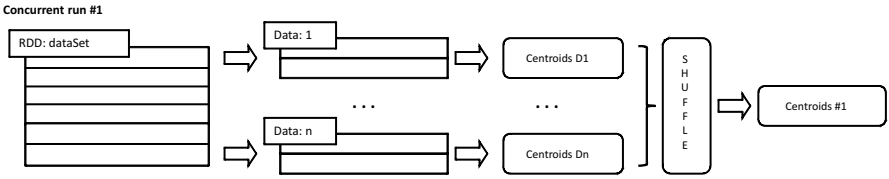

Fig. 2 One concurrent execution of the parallelized k-means algorithm.

In Figure 2 one execution of the the parallelized k-means algorithm version is described. Firstly, the RDD dataset is parallelized in n nodes for each concurrent run of the k-means. Therefore, n provisional centroids are obtained. Secondly, Spark shuffles the n centroids to provide a resulting centroid for each concurrent run. Finally, once all concurrent runs have been executed, the k-means|| algorithm computes the WSSSE for each centroid (note that there are as many centroids as number of concurrent runs). Thus, the algorithm returns the centroid that minimizes the WSSSE as the best centroid.

3 Results

The datasets used are related to the electrical energy consumption in two buildings located at a public university for years 2011, 2012 and 2013. The consumption is measured every fifteen minutes during this period. This makes a total of 35040 instances for years 2011 and 2013, and 35136 for the year 2012.

Note that there were several missing values ($< 3\%$). However, subsequent time stamps store the accumulated consumption for such instances. Therefore, the cleansing process consisted in searching for such values and assuming that consumption had been constant during these periods of time. That is, the stored value after missing values is divided by the number of consecutive registered missing values and assigned to each one.

Figure 3 shows the error obtained when applying the k-means for a number of clusters varying from 2 to 15 for the consumption of electricity of the years 2011, 2012 and 2013. The error used was the sum of squares of the distance between the points of each cluster. The error decreases smoothly for values greater than 6, do to this a number of clusters equal to 6 can be selected to provide satisfactory results.

Figure 4 presents the classification into 6 clusters obtained for K-means for the year 2013 (similar figures were obtained for the years 2011 and 2012). With just a quick look, the weekends, the working days and the typical periods of vacations in the university such as the Easter week (values from 83 to 90), the summer holidays (values from 213 to 243) or Christmas (values from 356 to 365) can be clearly differentiated.

The percentage of days classified into 6 clusters for each building is shown in Table 1. The last row presents the average of the electricity consumption for all the

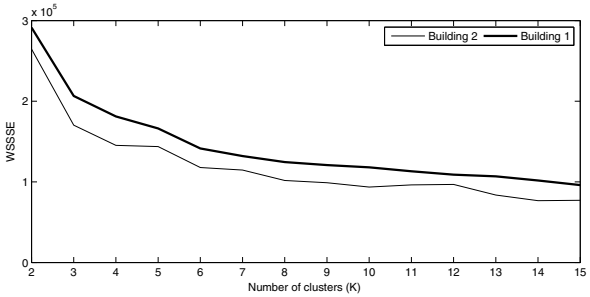

Fig. 3 Errors versus number of clusters for each building.

days associated with the cluster. Although it seems that the working days are equally distributed, a detailed analysis from Table 1 and Figure 4 reveals interesting patterns related to temperature and days with or with no scheduled teaching. For Building 1, teaching days with low and high temperatures (winter and summer) belong to the clusters of greater consumption, that is, cluster 4 and 5 respectively, and finally, teaching days with no extreme temperatures (autumn) are classified into cluster 2 of moderate consumption. Similar patterns can be found for Building 2.

Characteristic curves of each cluster are depicted in Figure 5. It can be observed that clusters 1 and 2 for Building 1, and clusters 1 and 2 for Building 2 are clusters composed of days of low consumption, namely weekends and holidays. Likewise, the remaining clusters correspond to working days, which are days of greater consumption.

Table 1 Percentage of days for each cluster in years 2011, 2012 and 2013.

Days	Building 1					
	Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Monday	5.73%	15.29%	12.74%	17.83%	10.83%	37.58%
Tuesday	5.73%	19.75%	12.10%	19.11%	8.92%	34.39%
Wednesday	5.77%	20.51%	12.18%	16.67%	8.33%	36.54%
Thursday	6.41%	16.67%	15.38%	17.95%	8.33%	35.26%
Friday	10.90%	11.54%	14.74%	14.74%	10.26%	37.82%
Saturday	42.68%	2.55%	48.41%	0.00%	0.64%	5.73%
Sunday	16.56%	1.27%	81.53%	0.64%	0.00%	0.00%
Average (in kW)	1.90	3.37	4.57	5.47	6.54	6.94

Days	Building 2					
	Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Monday	14.65%	38.22%	14.01%	16.56%	10.83%	5.73%
Tuesday	15.92%	32.48%	15.92%	19.11%	8.28%	8.28%
Wednesday	16.03%	35.90%	14.74%	15.38%	7.69%	10.26%
Thursday	19.87%	35.90%	15.38%	12.82%	7.69%	8.33%
Friday	19.23%	41.67%	13.46%	12.18%	8.97%	4.49%
Saturday	80.89%	17.83%	0.00%	0.64%	0.00%	0.64%
Sunday	96.82%	1.91%	0.00%	0.64%	0.00%	0.64%
Average (in kW)	1.41	2.39	3.96	4.71	5.43	6.39

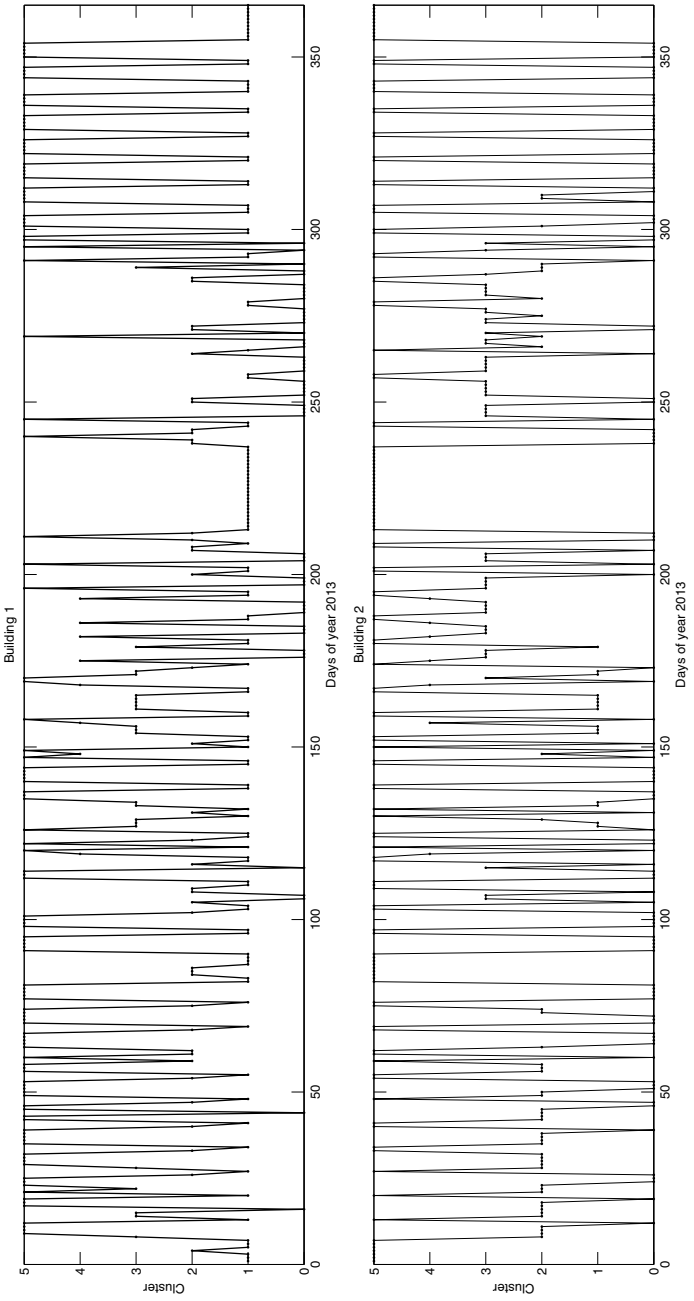

Fig. 4 Classification of the electricity consumption for each building.

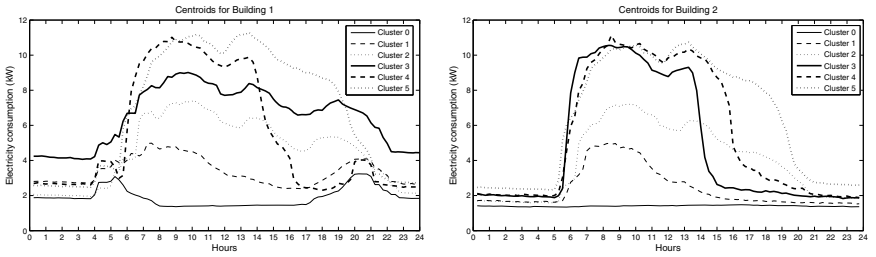

Fig. 5 Centroids for each building in years 2011, 2012 and 2013.

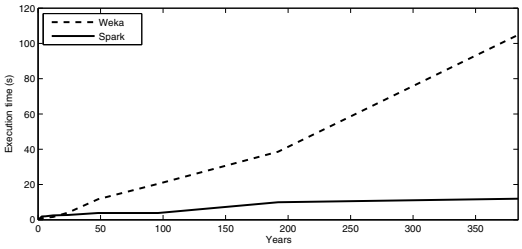

Fig. 6 CPU time for different sizes of datasets when using Weka and Spark.

Figure 6 shows the relation between the CPU time and the size of the dataset for k-means from Weka and Spark. This size has been set to 370 years, by synthetically generating such years. As a consequence of the results, it can be noticed that Weka has an exponential growth when the number of years comprising the time series increases, being remarkable the differences with Spark when the dataset is considered as big data.

4 Conclusions

In this work, a real big time series data composed of electricity consumptions has been analyzed by means of the k-means algorithm distributed version for Apache Spark. This parallelized version of the algorithm allows the discovery of daily consumption behaviors with a low computational cost. The results show different kinds of days according to the daily consumption as well as the identification of significant patterns related to working days with or with no scheduled teaching. Moreover, the CPU time of the proposed methodology has been compared to Weka, as a reference tool in data mining, proving to be a good solution for the big data clustering. Future works will be directed in the prediction of big time series once known the previous clustering, and the discovery of patterns for the classification of all the buildings of an organization in the context of smart cities.

Acknowledgments The authors would like to thank the Spanish Ministry of Economy and Competitiveness, Junta de Andalucía for the support under projects TIN2014-55894-C2-R and P12-TIC-1728, respectively.

References

1. Bahmani, B., Moseley, A., Vattani, R., Kumar, R., Vassilvitskii, S.: Scalable k-means++. Proceedings of the VLDB Endowment **5**(7), 622–633 (2012)
2. Capó, M., Pérez, A., Lozano, J.A.: A recursive k-means initialization algorithm for massive data. In: Proceedings of the Spanish Association for Artificial Intelligence (2015)
3. Dean, J., Ghemawat, S.: Mapreduce: Simplified data processing on large clusters. Communications of the ACM **51**(1), 107–113 (2008)
4. Ding, R., Wang, Q., Dan, Y., Fu, Q., Zhang, H., Zhang, D.: Yading: fast clustering of large-scale time series data. Proceedings of the VLDB Endowment **8**(5), 473–484 (2015)
5. Fahad, A., Alshatri, N., Tari, Z., Alamri, A., Zomaya, A.Y., Khalil, I., Sebti, F., Bouras, A.: A survey of clustering algorithms for big data: Taxonomy & empirical analysis. IEEE Transactions on Emerging Topics in Computing **5**, 267–279 (2014)
6. Fernández, A., del Río, S., López, V., Bawakid, A., del Jesús, M.J., Benítez, J.M., Herrera, F.: Big data with cloud computing: an insight on the computing environment, mapreduce, and programming frameworks. WIREs Data Mining and Knowledge Discovery **4**(5), 380–409 (2014)
7. Machine Learning Library (MLlib) for Spark (2015). http://spark.apache.org/docs/latest/mllib-guide.html
8. Hamstra, M., Karau, H., Zaharia, M., Knwinski, A., Wendell, P.: Learning Spark: Lightning-Fast Big Analytics. O' Really Media (2015)
9. Hernández, L., Baladrón, C., Aguiar, J.M., Carro, B., Sánchez-Esguevillas, A.: Classification and clustering of electricity demand patterns in industrial parks. Energies **5**, 5215–5228 (2012)
10. Keyno, H.R.S., Ghaderi, F., Azade, A., Razmi, J.: Forecasting electricity consumption by clustering data in order to decline the periodic variable's affects and simplification the pattern. Energy Conversion and Management **50**(3), 829–836 (2009)
11. Martínez-Álvarez, F., Troncoso, A., Riquelme, J.C., Riquelme, J.M.: Discovering patterns in electricity price using clustering techniques. In: Proceedings of the International Conference on Renewable Energy and Power Quality, pp. 245–252 (2007)
12. Martínez-Álvarez, F., Troncoso, A., Riquelme, J.C., Riquelme, J.M.: Partitioning-clustering techniques applied to the electricity price time series. In: LNCS, vol. 4881, pp. 990–991 (2007)
13. Minelli, M., Chambers, M., Dhiraj, A.: Big Data, Big Analytics: emerging business intelligence and analytics trends for today's businesses. John Wiley and Sons (2013)
14. Rakthanmanon, T., Campana, B., Mueen, A., Batista, G., Westover, B., Zhu, Q., Zakaria, J., Keogh, E.: Addressing big data time series: Mining trillions of time series subsequences under dynamic time warping. ACM Transactions on Knowledge Discovery from Data **7**(3), 267–279 (2014)
15. Van Wijk, J.J., Van Selow, E.R.: Cluster and calendar based visualization of time series data. In: Proceedings of the International IEEE Symposium on Information Visualization (1999)
16. Zhao, W., Ma, H., He, Q.: Parallel k-means clustering based on mapreduce. In: LNCS, vol. 5391, pp. 674–679 (2009)