VIPOC Project Research Summary (Discussion Paper)

Michelangelo Ceci[2], Roberto Corizzo[2], Fabio Fumarola[2], Michele Ianni[3], Donato Malerba[2], Gaspare Maria[4], Elio Masciari[1], Marco Oliverio[1], and Aleksandra Rashkovska[2]

[1] ICAR-CNR
[2] University of Bari A. Moro, Department of Computer Science
[3] UNICAL
[4] GFM-Integration
{masciari,oliverio@icar.cnr.it}, {name.surname}@uniba.it, michele.ianni@unical.it

Abstract. Predicting the output power of renewable energy production plants distributed on a wide territory is a valuable goal, both for marketing and energy management purposes. In this paper, we describe *Vi-POC* (Virtual Power Operating Center) – a distributed system for storing huge amounts of data, gathered from energy production plants and weather prediction services. Due to the heterogeneity and the high volume of data, it is necessary to exploit suitable Big Data analysis techniques in order to perform a quick and secure access to data that cannot be obtained with traditional approaches for data management. We use HBase over Hadoop framework on a cluster of commodity servers in order to provide a system that can be used as a basis for running machine learning algorithms. We perform one-day ahead forecast of PV energy production based on Artificial Neural Networks in two learning settings – structured and non-structured output prediction. Preliminary experimental results confirm the validity of the approach, also when compared with a baseline approach.

1 Introduction

Recently, renewable energy research is gathering a lot of attention due to the strategic and urgent need of reducing pollution emission. A key problem for renewable energy producers is the exact quantification of the amount of energy that can be pushed over the electric network. This problem arises as energy cannot be stocked efficiently; thus, they are forced to give it for free if they produce more energy than the network can use or they will pay a huge penalty if they do not provide the expected amount. In this perspective, the Vi-POC project has been developed in order to support renewable energy providers with a framework for collecting, storing, analyzing, querying and retrieving data coming from heterogeneous renewable energy production plants (such as photovoltaic, wind, geothermal, Sterling engine, water running) distributed on a wide territory. Moreover, Vi-POC features an innovative system for real-time prediction of the energy production which integrates data coming from production plants and weather production services.

The data coming from production plant are quite heterogeneous, they arrive at a continuously fast rate and their volume increases continuously. These features pose several challenges that can be solved using Big Data techniques [1, 6, 8, 9, 12–16].

Nowadays, dealing with a big volume of data is very challenging, since traditional technologies, like Relational Database Management System (RDBMS) or classical object oriented programming, are not well suited for this purpose. As Big Data also arise at high speed and variety, we need to cope with requirements like scalability. Traditional RBDMS are not suitable for the typical size and scalability requirements of Big Data. In order to meet these requirements, *distributed non-relational* database management systems have been proposed. The use of non-relational database approach, also known as NoSQL (Not Only SQL), allows for avoidance of unneeded complexity, high throughput, horizontal scalability and possible running on commodity hardware, and avoidance of expensive object-relational mapping.

Many open source technologies were developed in order to effectively handle massive amounts of data. The majority of these technologies are based on the MapReduce programming model. This paradigm makes easier to implement solutions based on the use of distributed systems for executing analysis tasks. However, although MapReduce is well suited for large distributed data processing where fast performance is not an issue, its high-latency batch model is not effective for fast computations or real data analysis. The most widespread implementation of the MapReduce programming model is Hadoop MapReduce, part of the Hadoop framework. Although Hadoop is a really pervasive technology, it has some drawbacks, especially when dealing with algorithms based on iterative operations.

In our project, we exploited HBase - an open source, non-relational, distributed database model. HBase is an Apache project and runs on top of Hadoop Distributed File System (HDFS), providing Google BigTable-like capabilities for Hadoop, i.e., it provides a fault-tolerant way of storing large quantities of sparse data. HBase main features include good compression performances, in-memory execution of operation, and bloom filters on a per-column basis as in BigTable.

In this paper, we describe our end-to-end framework that, starting from the data leaning step, allows a better engineering of data structures in order to support data analysis and prediction of energy production. As for the prediction of the energy production, we propose a method for long-term forecast (one-day ahead) of photovoltaic energy production based on Artificial Neural Networks (ANN) and investigate the performance of structured and non-structured output prediction models. While in non-structured output prediction, a prediction model generates the forecast for a specific hour in the future, in structured output prediction, a prediction model generates the forecast for 24 hours in the future [11][4]. In principle, the main advantage of structured output prediction is in the implicit consideration of the dependence of the predictions at two consecutive hours.

In [7], we described the high level architecture of our system. In this paper, we will describe the actual implementation of the prototype along with the implementation issues that are crucial for building a system for Big Data management and analysis.

2 HBase Data Model

HBase data model is “sparse, distributed, persistent, multi-dimensional sorted map”. HBase is *distributed* and *persistent* features are guaranteed by automatically storing

data in a redundant way by exploiting a specialized distributed file system as HDFS, that spread data across different machines that usually represent different nodes of a given cluster. Moreover, data are stored in a *multi-dimensional* map for fast indexing by row key, column and version. Furthermore, data are also partitioned across nodes of the cluster in *regions* composed of a contiguous range of row. Every region is served by only one machine, denoted as *Region Server*. However, this data model lacks some useful operations available for classical RDBMS solutions, like joins, foreign keys, referential integrity and transaction support. If the application being implemented requires these features, they need to be implemented ad-hoc.

HBase allows for significant reduction of the overall storage space required for saving relevant information. The operation performed for storing more information in a single key is referred to as *mashing* and can be performed in several ways, as value concatenation or by formatting data using a suitable delimiter. Mashing has several important advantages: i) Row key ordering allows fastest querying compared to alternative query patters such as time-stamp based querying or column qualifier based querying; ii) It allows partial scan of the HBase data based on mash ordering ; iii) As tables are partitioned in regions among nodes, tables should not contain a huge number of columns Properly exploiting row keys allows to define tables exhibiting the above mentioned feature as mashing causes column values to collapse in a single row key identifier.

Finally, row key design plays a crucial role for load balancing among region servers. When tuples are inserted in the data store, if they share a row key prefix, they will be stored in the same region server causing an unbalanced cluster loading. This drawback is referred to as *hot spotting*. However, it is possible to avoid hot spotting by exploiting strategies such as *salting* (adding a random value as row key prefix to make the value distribution uniform), *hashing* (the hash code of the key is used instead of the row key identifier). It is also possible to apply reversed row key identifier in order to obtain the least significant digit as prefix. The rationale of this choice relies in the fact that often last digits change more frequently than the ones preceding it (e.g. the time-stamp).

3 Renewable Energy Case Study

The Vi-POC project aims at designing and implementing a prototype which is able to manage renewable energy production plants distributed over the national territory. Vi-POC implements an innovative system for real-time prediction of the energy production capability of each plant. It exploits big data techniques explained above in order to deal with the heterogeneity of data coming from different renewable resources, such as photovoltaic (PV), wind, geothermal, Sterling engine, water running. Vi-POC is intended to make the prediction available from GSE (Gestore Servizi Energetici `http://www.gse.it/`) more efficient, effective and reliable. In particular, Vi-POC intents to predict real-time energy production with higher precision as it exploits historical information about production and weather conditions. The high accuracy and efficiency will allow energy market operators to implement a more effective purchasing strategy.

We exploited an HBase storage system designed for storing weather information and plant sensor data. These data are exploited by clients running data mining algo-

rithms to predict the output power of the plants. Every plant sends periodically all the data collected by installed sensors. The time granularity is set based on the type and the dimension of the plant. Data coming from the plants usually consists of different measures gathered from several sensors at a given time-stamp. Moreover, the number and the type of sensors may differ among plants. Forecast data instead, consists of various predicted weather parameters forecasted for a given time and location.

Separation between the plants and the computation cluster is a key concept. The plants, in fact, do not send their measurements directly to the computation cluster, but to a separated storage level, made of several file servers. Different fault tolerance strategies are applied among these levels in order to avoid the block of the entire system due to the failure of one of the components. Data is then taken by computation cluster's *Extract, Transform and Load (ETL)* tool and stored in a non-relational distributed database across the nodes of the cluster itself.

Our architecture stores the data on a HBase system consisting of three tables for storing: plant information, measurements from plants and forecasting information. To store data regarding a location, we use Geohash[5]. It is a standard way to represent latitude/longitude information as a string of characters having very useful properties. As an example, sites close to each another share the same prefix in the string.

HBase performances heavily decrease when more than three column families are used because flushing and compaction are performed on a per-region basis. Thus, if a column family is carrying the bulk of the data being flushed, the adjacent families will also be flushed even though the amount of data they carry is small. As a consequence, when many column families are exploited, the flushing and compaction interaction can heavily decrease system performances. In this respect, we designed column family schemas having at most two column families.

3.1 Table schemas

Based on the above considerations, we designed tables as described below (we do not report the actual name of each attribute as they are coded by the plant owner and they are not easily understandable).

Table *Plants*:

- *RowKey*: concatenation of the type of the plant (solar, wind, hydroelectric) and a plant identifier;
- *Column family 1*: contains as many attributes as the cardinality of data. Every attribute represents raw information as the configuration parameters or the coordinates of the plant;
- *Column family 2*: stores log information about maintenance operations for the specific plant.

Table *Measure*:

- *RowKey*: concatenation of the identifier of the plant, the reverse time stamp and the measurement type;

[5] www.geohash.org

– *Column family*: stores all collected measures. The number of attributes is equal to the cardinality of counters being collected.

Table *Predicted Measure*:
– *RowKey*: the same structure as the *Measure* table;
– *Column family*: stores the measures predicted by mining algorithms. The number of attributes is equal to the cardinality of predicted data.

Table *Weather Data*:
– *RowKey*: concatenation of Geohash, reverse time stamp, measurement type and server identifier, where server identifier is used to trace which server sent the prediction;
– *Column Family 1*: used to store collected weather data.
– *Column Family 2*: used to store predicted weather data (weather forecasts).

3.2 Long-term forecast of PV energy production

During the last years, the forecast of PV energy production has received significant attention since photovoltaics are becoming a major source of renewable energy for the world [2]. Forecasting methods depend on the tools and information available, the forecast horizon, the number of plants considered and the size of the geographic area they cover [17]. Diverse resources are used to generate solar and PV forecasts, ranging from measured weather and PV system data, satellite and sky imagery cloud observations, to Numerical Weather Prediction (NWP) models [10]. The best approaches make use of both measured data and NWP models.

In the literature, several data mining approaches have been proposed for renewable energy power forecasting. It has been noted that physical (e.g. wind speed and solar irradiation) property behavior exhibits a trail called concept drift, i.e., they change characteristics over time [5]. In this respect, adaptive models are generally considered to produce more reliable predictions regarding concept drift, but require a continuous training phase [3][18].

In this case study, we propose an adaptive method for long-term forecast (one-day ahead) of PV energy production based on ANN. The proposed approach exploits NWP to benefit from uncontrollable factors (such as weather conditions). We investigate structured (all hours of the forecasted day are outputs from a single ANN model) and non-structured output prediction models (each hour of the forecasted day is output from one ANN model).

Method The machine learning task is to predict the PV power generation using the following input attributes:

– the geographic coordinates of the plant: latitude and longitude,
– the sun positions at the location of the plant: altitude and azimuth, queried by SunPosition (`http://www.susdesign.com/sunposition/index.php`),
– the properties of the plant: site ID, brand ID, model ID, age in months,
– weather data: ambient temperature, irradiance, pressure, wind speed, wind bearing, humidity, dew point, cloud cover, descriptive weather summary.

Additionally, in the case of structured output prediction, also the day is passed as feature, while in the case of non-structured output prediction, besides the day, also the hour. In the training phase, we use historical weather information collected by sensors, while for prediction purposes, we use weather forecast data provided by NWP systems (data are normalized according to the z-score). The output is the prediction of the power production (KWh) for the next day at one hour intervals. The prediction models are updated on a daily basis.

Experiments In our empirical evaluation, we consider a real dataset collected at regular intervals of 15 minutes (measurements start at 2:00 AM and stop at 8:00 PM every day) by sensors located on 18 plants in Italy. The time period spans from January 1st, 2012 to May 4th, 2014. The weather data is queried from Forecast.io (`http://forecast.io/`), while the irradiance is queried from PVGIS (`http://re.jrc.ec.europa.eu/pvgis/apps4/pvest.php`). As anticipated before, the raw data are preprocessed and normalized according to the z-score normalization, before using them for ANN learning, in order to resolve measurement errors.

In this paper, we use the *encog* implementation of the Resilient Propagation (RPROP+) algorithm for training neural networks (`http://www.heatonresearch.com/wiki/Resilient_Propagation#Implementing_RPROP.2B`). RPROP+ is one of the best general-purpose neural network training methods implementing the back-propagation technique. We use RPROP+ since it has been proven effective for renewable energy prediction [5]. For the evaluation, the dataset is split into training days (85%) and testing days (15%). Experiments are run three times and average results are collected. For each run the network is trained incrementally on the training dataset until a testing day is found. Then, it is repeatedly first tested on the testing day and after that it is re-trained with the sample added to the training set, together with all the training days before the next testing day. At the end, the average performance over all the test samples is reported as a result.

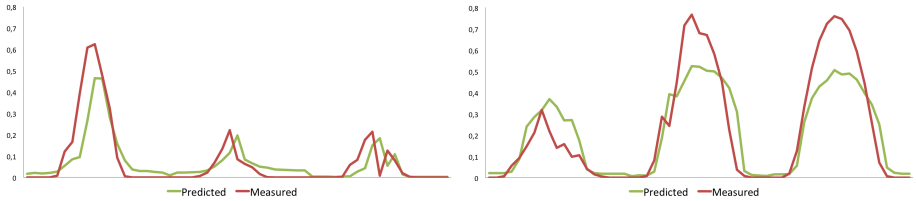
We distinguish between hourly (non-structured) and daily (structured output) settings. In the hourly setting, we investigate non-structured ANN models with single output - the production of the plant at a specified day and specified hour. In the daily setting, we investigate structured ANN models with 19 outputs - the productions of the plant for the hours from 2:00 AM to 8:00 PM on a specified day. Moreover, in each setting, we investigate the performance of models that do not take into account any spatial information about the data (No Spatial) and models that take into account the latitude and longitude of the plant as spatial information (Lat Lon).

Results and discussion The results for the investigated hourly and daily scenarios are reported in Table 1. We consider three indicators of the predictive performance, namely the Root Mean Squared Error (RMSE), the Mean Absolute Error (MAE) and the improvement with respect to the persistence model (i.e., the model that forecasts the same production observed 24 hours before).

The results clearly show improvement of the predictive performance over the persistence model, with the structured-output prediction model outperforming the non-structured one, both in No Spatial and Lat Lon models. From Table 1, we can also

Table 1. Performance results for one-day ahead PV power forecast for hourly (non-structured) and daily (structured output) settings. No spatial (Lat Lon) indicate results without (with) geographic coordinates of the plant.

	RMSE	MAE	Impr. over the persistence mod. [%]
No Spatial Hourly	0,121	0,080	16,622
No Spatial Daily	0,111	0,068	23,966
Lat Lon Hourly	0,119	0,078	17,745
Lat Lon Daily	0,109	0,067	25,369
Persistence model	0,146	0,085	



(a) January 4^{th}, 5^{th}, 6^{th}. (b) May 4^{th}, 5^{th}, 6^{th}.

Fig. 1. Predictions (green) and measurements (red) of the productions for three consecutive days of a single plant. The three consecutive days are taken from January and May. Results are obtained with the daily (structured) setting. We recall that the time intervals considered are 2:00 AM - 8:00 PM. Results are obtained including geographic coordinates.

notice that geographic coordinates play an important role in the prediction effectiveness - the Lat Lon models outperform the No Spatial models. This suggests that spatial autocorrelation phenomena should be taken into account when constructing predictive models for the production of geographically distributed plants [19].

The predictive performance of the model can also be graphically inspected from Figure 1, where the predicted vs. measured power production are presented for three consecutive typical cold (in January) and warm (in May) days. In both cases we report predictions for partially cloudy days. The predictions are obtained using the best performing model, i.e. structured output considering the latitude and longitude as input attributes. The plots confirm that the ANN models predict the production for the next day visually better in winter than in summer days.

4 Conclusions

In this paper, we have presented the project Vi-POC – a distributed system for storing, querying and analyzing data collected from renewable energy production plants. In particular, we have described its data model and its forecasting capabilities. We have empirically shown its predictive capabilities and compared cases with structured output prediction and non-structured output prediction. Results confirm that predictive capabilities are better in case of structured output prediction, probably because of the implicit consideration of the dependence of the predictions at consecutive hours.

References

[1] D. Agrawal et al. “Challenges and Opportunities with Big Data. A community white paper developed by leading researchers across the United States”. In: (2012).

[2] EPIA European Photovoltaic Industry Association. *Global Market Outlook for Photovoltaics 2014-2018*. http://www.epia.org/news/publications/global-market-outlook-for-photovoltaics-2014-2018. 2014.

[3] Peder Bacher, Henrik Madsen, and Henrik Aalborg Nielsen. “Online short-term solar power forecasting”. In: *Solar Energy* 83.10 (2009), pp. 1772 –1783. ISSN: 0038-092X.

[4] Gökhan H. Bakır et al., eds. *Predicting structured data*. The MIT Press, 2007.

[5] R.J. Bessa, V. Miranda, and J. Gama. “Entropy and Correntropy Against Minimum Square Error in Offline and Online Three-Day Ahead Wind Power Forecasting”. In: *Power Systems, IEEE Transactions on* 24.4 (2009), pp. 1657–1666.

[6] “Big Data”. In: *Nature* (Sept. 2008).

[7] Michelangelo Ceci et al. “Big Data Techniques For Renewable Energy Market”. In: *22nd Italian Symposium on Advanced Database Systems, SEBD 2014, Sorrento Coast, Italy, June 16-18, 2014*. Ed. by Sergio Greco and Antonio Picariello. 2014, pp. 369–377.

[8] “Data, data everywhere”. In: *The Economist* (2010).

[9] “Drowning in numbers - Digital data will flood the planet - and help us understand it better.” In: *The Economist* (2011).

[10] Jan Kleissl. *Solar Resource Assessment and Forecasting*. Elsevier, 2013. ISBN: 978-0-12-397177-7.

[11] Dragi Kocev et al. “Tree ensembles for predicting structured outputs”. In: *Pattern Recognition* 46.3 (2013), pp. 817–833.

[12] A. Labrinidis and H. V. Jagadish. “Challenges and Opportunities with Big Data”. In: *PVLDB* 5.12 (2012), pp. 2032–2033.

[13] S. Lohr. “The age of big data”. In: *nytimes.com* (2012).

[14] J. Manyika et al. “Big data: The next frontier for innovation, competition, and productivity”. In: *McKinsey Global Institute* (2011).

[15] Y. Noguchi. “Following Digital Breadcrumbs to Big Data Gold”. In: *National Public Radio* (2011).

[16] Y. Noguchi. “The Search for Analysts to Make Sense of Big Data”. In: *National Public Radio* (2011).

[17] Sophie Pelland et al. *Photovoltaic and Solar Forecasting*. Tech. rep. IEA PVPS, 2013.

[18] Navin Sharma et al. “Predicting solar generation from weather forecasts using machine learning.” In: *SmartGridComm*. IEEE, 2011, pp. 528–533. ISBN: 978-1-4577-1704-8.

[19] Daniela Stojanova et al. “Dealing with spatial autocorrelation when learning predictive clustering trees”. In: *Ecological Informatics* 13 (2013), pp. 22–39.