Article

A Data-Driven Multi-Regime Approach for Predicting Energy Consumption

Abdulgani Kahraman ^{1,*} , Mehmed Kantardzic ¹, Muhammet Mustafa Kahraman ² and Muhammed Kotan ³ ¹ Department of Computer Science and Engineering, University of Louisville, Louisville, KY 40292, USA; mmkant01@louisville.edu² Department of Mining Engineering, Gumushane University, Gumushane 29100, Turkey; kahraman@gumushane.edu.tr³ Department of Information Systems Engineering, Sakarya University, Sakarya 54050, Turkey; mkotan@sakarya.edu.tr

* Correspondence: a0kahr01@louisville.edu

Abstract: There has been increasing interest in reducing carbon footprints globally in recent years. Hence increasing share of green energy and energy efficiency are promoted by governments. Therefore, optimizing energy consumption is becoming more critical for people, companies, industries, and the environment. Predicting energy consumption more precisely means that future energy management planning can be more effective. To date, most research papers have focused on predicting residential building energy consumption; however, a large portion of the energy is consumed by industrial machines. Prediction of energy consumption of large industrial machines in real time is challenging due to concept drift, in which prediction performance deteriorates over time. In this research, a novel data-driven method multi-regime approach (MRA) was developed to better predict the energy consumption for industrial machines. Whereas most papers have focused on finding an excellent prediction model that contradicts the no-free-lunch theorem, this study concentrated on adding potential concept drift points into the prediction process. A real-world dataset was collected from a semi-autonomous grinding (SAG) mill used as a data source, and a deep neural network was utilized as a prediction model for the MRA method. The results proved that the MRA method enables the detection of multi-regimes over time and provides a highly accurate prediction performance, thanks to the dynamic model approach.

Keywords: energy efficiency; energy consumption prediction; concept drift; deep learning; industrial machines

Citation: Kahraman, A.; Kantardzic, M.; Kahraman, M.; Kotan, M. A Data-Driven Multi-Regime Approach for Predicting Energy Consumption. *Energies* **2021**, *14*, 6763. <https://doi.org/10.3390/en14206763>

Academic Editor: Antonio Morán

Received: 2 September 2021

Accepted: 14 October 2021

Published: 17 October 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Energy management is becoming vital for companies around the world, and energy prediction is necessary as an initial step to create an energy management system [1]. Predicting energy consumption is not only essential for energy management, but it is also crucial when considering climate change [2]. In the last decade, numerous researchers have applied several statistical analyses, such as data mining, machine learning and deep learning methods, on time series to predict energy consumption for buildings, cities and industrial machines [3–5].

The mining industry is extremely vulnerable from an economic position because of recession, economic uncertainty and the use of machines that are costly to maintain [6]. In mineral processing, comminution is one of the substantial operations especially for milling [7]. Moreover, milling machines have high-cost energy consumption and maintenance expenses [8,9]. Semi-autogenous grinding (SAG) mills have been more common for mining due to economic advantages, such as advanced processing capacity, low physical space necessity, relatively lower maintenance expenses, convenient configuration, and low investment [10–12]. Feed size, ore hardness and mill load are essential variables for productive SAG mill operation, but it is not always possible to have optimum values [13].

Additionally, SAG mills have the highest electricity consumption during the whole comminution process [14]. Predicting the machines' energy consumption can be very valuable for companies, as this may be used to estimate energy costs in advance and improve the efficiency of the mills [15]. Figure 1 illustrates sample inputs (density, fresh feed, feed size, sound and pressure) and output (energy consumption) variables over time for a SAG Mill. The dataset details will be shared in Section 4.

Figure 1. SAG Mill inputs and output, where the x -axes represent the time and the y -axes reflect the value.

Predicting accurate energy consumption is a complex and critical task for researchers. Detecting change points and integrating these unpredictable change points into the prediction process is one of the most challenging tasks [16]. Time series forecasting has become more popular in recent decades due to the significant applications in numerous fields [17], such as energy consumption [18–20], predicting financial variables [21] and wind power generation [22,23]. A single method cannot achieve satisfactory prediction results for all types of time series [24]. Various intricate models have been studied to examine the time series' nonlinear behavior [25,26]. However, one of the biggest challenges is that there may be frequently repetitive data variations called concept drift over time on the data stream [27].

The term “concept drift” refers to unanticipated shifts in the underlying distribution of streaming data over time. Furthermore, many researchers have tried to solve the multiple regime problem on time series data [16,26,28,29]. Nevertheless, there are not sufficient papers that investigate real industrial machines' time series with possible multi-regime solutions. To the authors' best knowledge, most papers employ a traditional approach to create their prediction model, which disregards the data stream with concept drift issues and results in less accurate prediction performance over time.

In this research, first, the conventional method was applied to show the performance declines for a real time SAG Mill dataset. Then, a data-driven model was developed to avoid these degradations in the model prediction performance. Furthermore, if there are multiple repetitive regimes in industrial machinery datasets, the MRA method aids in detecting these regimes with a high level of accuracy. Finally, we compared the performance of the MRA method with the traditional approach, and the results proved that the proposed model reduces the overall error rate and is useful in finding repetitive regimes.

1.1. Motivation

A large number of research papers have investigated the buildings' or cities' energy consumption [30–34]. However, industrial machines use a vast amount of energy [35]; the industrial sector consumes over half of the world's total energy, and its energy consumption has nearly doubled in the last 60 years [36]. For example, the average annual electricity consumption for a U.S. residential utility customer in 2019 was 10,649 kilowatt-hours (kWh) [37], and a SAG mill consumes the same amount of energy in an hour. Furthermore, industrial machines tend to have more complicated distributed time series with multi-regime running cases, which may be for reasons such as operator behavior, load amount, material type, or size. In this research, a novel data-driven method was developed to enhance the prediction performance of industrial machine energy consumption based on the variables.

1.2. Contribution

Due to the concept drift, which is one of the biggest problems in the real-time data stream, the prediction accuracy of the traditional approach is decreasing over time. A novel method has been proposed to handle concept drift issues and precisely estimate the energy consumption of industrial machines. In addition, instead of dividing the real-time data into the fixed size of chunks, the data were split into the variant size of chunks based on the machine's operating conditions. Change points and recurrent regimes have thereby been successfully detected over time.

The rest of the paper is organized as follows: Section 2 presents the related work and differences with the MRA method. Section 3 explains the MRA method in detail. The results and comparisons are shared in Section 4. Finally, conclusions and future work are given in Section 5.

2. Related Work

Over the past few decades, energy consumption and efficiency have attracted an increasing number of researchers, not only for energy saving and supply purposes, but for CO₂ emissions, which have a significant impact on climate change [30]. An artificial neural network (ANN) was implemented based on the principal physical method to estimate the building energy consumption [31]. Hamzacebi demonstrated the power of the ANN for the prediction of the seasonal time series [32]. A novel method named pattern sequence-based forecasting (PSF) was developed by Alvarez et al. First, a clustering method was applied to cluster the time series data. Second, the sequence of labeled groups was calculated to predict the next day group, which increased the model performance for the specified group of the time series [33]. Hill et al. compared the traditional statistical method and the neural network method on the time series forecasting [14]. Similarly, Tso et al. showed that the decision tree and the neural network outperformed the regression method for the Hong Kong energy-consumption prediction [34]. Kankal et al. used four independent variables, gross domestic product, population, and the amount of import and export, and implemented an ANN to forecast energy demand [38]. A genetic algorithm and ANN were integrated by Azadeh et al. to forecast electricity demand for agricultural activities by using stochastic procedures [39]. Wang et al. developed a method to select secondary variables data from the cooling energy consumption dataset, and the model discovered periodicity over the time series. As a result, the model could predict energy consumption more precisely compared to the conventional methods [40].

He et al. developed a novel data-driven energy prediction approach to predict the energy consumption of grinding and milling machines. They implemented several feature extraction methods to eliminate unnecessary features, and deep learning was used as a prediction method. The results increased prediction accuracy compared to the traditional approach [41]. Another similar study was carried out for the prediction of energy consumption of electric arc furnaces, and the results proved that deep neural networks outperformed support vector machines, linear regression, and decision trees [42]. Kant and

Sangwan implemented an ANN to predict the cutting energy of machining, and the results confirmed that higher feed rate and spindle speed use less energy [36]. Avalos et al. used a real-time operational variable feed tonnage, bearing pressure, and spindle speed from SAG mills. They implemented several deep learning and machine learning techniques to predict the energy consumption of the SAG mills. The results showed that neural networks achieved one of the best prediction performances for SAG mill energy consumption [7].

Several researchers have used the Markov regime-switching model to detect change points related to the multi-regime approach [25,43]. The disadvantage of the model is that the change points must be known before it is applied. However, each machine has specific properties and working conditions, requiring a unique approach for detecting change points more accurately [43]. Additionally, the model is less explainable and problematic to forecast, and it is broadly used in economics to define different structures [25].

3. Methodology

Industrial machines usually have complex designs and working conditions. Solutions must consider many aspects, such as feature selection, noisy data, trending data, stationarity, nonlinearity, seasonality, and multi-regimes. Moreover, accurately analyzing industrial time series requires an interdisciplinary approach to better understand the problem. In this paper, a novel data-driving model was developed, and working conditions and running cycles were considered based on a subject matter expert's (SME's) advice, which helped us to develop a better prediction model.

Figure 2 summarizes the main steps for the proposed method to predict SAG mill energy consumption with a multi-regime approach.

Figure 2. Steps followed for developing the proposed model.

Step 1: Understanding the data and running conditions of the machine is a crucial element to accurately discover the machine's potential change points over time. Furthermore, an SME was consulted to decide threshold values for the output and potential change points to investigate possible regime regions, named chunks. There were five factors that directly or indirectly impact energy consumption, and all those features were used as input variables.

Step 2: Real-time industrial data usually have several issues, such as missing values, outliers, noisy data, changing feature tag names by time, upgrading sensor quality, and sensitivity. As a preprocessing step, the data were cleaned for further processing. Missing values are a widespread problem due to the reliability of sensor quality, and there are

two common strategies in the literature [20]. If the majority of features were missing in a single record, the whole row was removed from the dataset. If the minority of the records were missing for a single line, they were replaced with their mean value. In this way, we attempted to use each record as much as possible. Each chunk has a different data size, and several chunks have a limited number of instances. Furthermore, when the data are split into several chunks, it is considered that any record would be valuable for their chunk.

Step 3: This step was mainly designed to discover potential regime areas. Several change points were selected when a daily cumulative energy consumption equaled zero for more than 24 h. Long-time inactivity is abnormal for a SAG mill as they work 24 h a day, seven days a week, except for regular maintenance or machine breakdowns. During inactive days, various operational changes on the machine that might significantly impact the machine running cycle are considered potential change points. Furthermore, the threshold timing should be updated according to machine type and working conditions.

Equation (1) is used for deciding the change points and the chunks. W_t represents the timing window threshold and is a minimum 24 h time period for the cumulative energy consumption. S_{OV} exemplifies the sum of the output variable, and $Ov(t)$ symbolizes hourly energy consumption for the selected duration.

$$S_{OV} = \sum_{t=1}^{W_t} Ov(t). \quad (1)$$

$$Time \geq W_t \begin{cases} S_{OV} = 0; & \text{Change Point=Yes, and New Chunk=Yes} \\ S_{OV} \neq 0; & \text{Change Point=No, and New Chunk=No} \end{cases}$$

After separating the data into several chunks based on the threshold, a deep neural network (DNN) model was developed based on the first chunk. The chunk data were divided into several training and testing percentages, and the final split ratio was decided according to performance. The remaining chunks were used as unseen testing data.

A DNN was selected as a prediction method for the machine's energy consumption since it provides one of the best accuracies in the literature [20]. There are several alternatives to the DNN model, but a comparison for different models was not investigated in this paper. Overall, the main goal is to improve prediction performance by discovering potential repetitive multi-regimes over time.

A DNN model has an input layer, output layer, and multiple hidden layers. It uses a multi-layer feed-forward neural network structure. It also has more enhanced features, such as dropout, early stopping and penalties on the l1 and l2 norms of the weights against overfitting problems. Many hidden layers containing neurons with hyperbolic tangent function (tanh), rectifier, and sigmoid activation functions can be adjusted in the network. A sample of the DNN structure is illustrated in Figure 3.

Following the computation of the DNN, the value of the output is calculated using a feed-forward method. The mathematical description of the relationship between the output (y_t) and the inputs ($y_{(t-n)}$) is as follows [44]:

$$y_t = W_b + \sum_{j=1}^h W_j f[W_{bj} + \sum_{i=1}^n W_{ij}(y(t-1))] + E_t \quad (2)$$

W_{ij} and W_j are model parameters commonly referred to as connection weights. n is the number of input nodes, and h is the number of hidden nodes. W_b and W_{bj} are bias unit weights that are distinctive to each process unit, and f is the activation function, which is widely used as the Rectifier Linear Unit (ReLU) function. The network structure and connection weights determine the function f . The output error E_t is calculated each time and used as negative feedback to adjust the incoming-weight connections and bias. This adjustment allows the DNN's computation accuracy to be improved by reducing output mistakes to a minimum.

Figure 3. DNN structure.

Step 4: Thresholds are subjective, and they may have a significant impact on the results. The results can be evaluated during this step to discover optimum threshold values for detecting multiple regimes more accurately and predicting energy consumption more precisely. If the results are not satisfactory, the thresholds for the machine should be updated accordingly.

Finally, the results and possible future work are discussed with an SME, as each industrial machine may have specific working conditions and require an interdisciplinary approach.

The MRA method is illustrated as a flow chart in Figure 4. The method divides the dataset into several chunks and optimizes necessary models based on chunks. NC, Th, Err, C, M, and NM represent a total number of chunks, threshold value, error rate, chunk no, model no, and the number of models, respectively. The DNN model is developed based on Chunk-1 data in the first step, and the following chunks are used as unseen testing data. The most recent model is used for the subsequent chunks until the error rate exceeds the threshold value. When the error rate is greater than the set threshold value ($Err > Th$), the MRA method, first, uses all available historical models to obtain an error rate lower than the threshold. All previous models are used in a loop represented by symbol i in Figure 4 to discover a suitable historical model for the current chunk. If a satisfying result ($Err < Th$) cannot be found, the current chunk is assumed to be a new regime chunk requiring a new model, and the MRA method builds a new model for the current chunk's (C) data.

Additionally, the traditional approach, also named the static model, was compared with the MRA method results to see improvement in prediction performance. The most common metrics to determine the model's accuracy for continuous variables are root mean square error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE) [45]. RMSE, MAE, and MAPE are given in Equations (3)–(5), respectively. m stands for the number of samples in the test set, Y_i stands for the sample's actual value, and $\hat{Y}_i$ stands for the sample's predicted value. The lower values of these parameters mean the higher model's accuracy.

$$RMSE = \sqrt{\frac{1}{m} \sum_{i=1}^m (Y_i - \hat{Y}_i)^2} \quad (3)$$

$$MAE = \frac{1}{m} \sum_{i=1}^m |Y_i - \hat{Y}_i| \quad (4)$$

$$MAPE = \frac{100}{m} \sum_{i=1}^m \left| \frac{Y_i - \hat{Y}_i}{Y_i} \right| \quad (5)$$

All three evaluation metrics were shared and used to avoid overfitting and under-fitting. Furthermore, MAPE was used for determining the threshold values and general evaluation of the model performance.

Figure 4. The flow chart of the MRA method.

4. Experiments

4.1. Dataset

The dataset was collected from a SAG Mill over three consecutive years, and the summary statistics were illustrated in Table 1. There are several time intervals between sequential records for each variable, but all features have hourly average values in the dataset. Furthermore, each year has slightly different cumulative active hours; 8744, 8674, and 8539, respectively.

Table 1. Summary statistics of the dataset.

	Feed Particular Size (cm)	Mill Density (%)	Fresh Feed Amount (TPH)	Mill Sound (dB)	Mill Pressure (psi)	Mill Energy Consumption (kWh)
Mean	3.47	80.65	1069.23	30.26	895.61	9756.94
Standard Deviation	1.09	3.10	372.78	11.6	66.09	780.95
Minimum	0.28	60.02	0	7.04	240.895	492.60
Maximum	16.54	98.11	2196.72	99.1	1054.39	13,009.76

There are six input variables (feed particular size, fresh feed amount, mill density, mill sound, mill speed, and mill pressure) and one output variable (mill energy consumption) in the dataset. The mill speed data were removed from the input variables as a preprocessing step since they had around 80% missing values. The remaining inputs had less than 20%

missing values, which were preprocessed according to step two. The distribution of the inputs and the output are illustrated in Figure 5, where x -axes are the actual value and y -axes are the frequency.

Figure 5. The distribution of the feature variables, where the x -axes represent the actual value, and the y -axes reflect the frequency.

Time series problems can be distinguished from the more common classification and regression problems. If a time series has no pattern or seasonal impact, it is classified as stationary. It can be seen that all the three-year data have similar distributions, and the dataset appears to be stationary.

4.2. Experimental Results

Figure 6 shows sudden daily cumulative changes in the output value, and the following three graphics are separated according to the years. When the machine's energy consumption equals to zero and exceeds the threshold time duration (24 h), it is marked with red rectangular shapes.

The data between each consecutive red rectangular shape is described as a chunk. There are 23 different chunks in total over the three years. The first chunk in June of the third year has a small number of data that are counted as one marked shape. The first eight chunks occurred in the first year, chunks nine to 16 were observed during the second year, and the remaining seven chunks were seen in the third year. After the dataset was divided into different chunks, the MRA method was implemented to detect possible multiple repetitive regime areas based on these chunks. Furthermore, each chunk has a different sample size as they were not divided into a fixed size. Therefore, the MRA method has a more flexible and dynamic approach compared to the static model.

For all DNN structures, we used the standardization function since our features have different range scales. Numerous combinations were tried to discover the optimum hyperparameter values for the DNN prediction performance. The best model was found by varying the number of hidden layers in the set of three, four, five, and the number of neurons in the set of 50, 100, and 150 selected for hyperparameter tuning. The numbers of epochs were selected to be 10, which is the number of passes over the training dataset. ReLU and tanh functions were used as the activation functions, but ReLU outperformed the tanh. We also used the early stopping criterion and dropout function for hidden layers to avoid overfitting when it is required. Different split ratios were also tried as a training and testing part, which is illustrated in Appendix A Table A1. According to prediction performance, the split ratio was decided for each model accordingly. In addition, epsilon, which provides forward progress, was selected as $(1.0) \times 10^{(-8)}$. Rho was chosen as 0.99, the gradient moving average decay factor used for the learning rate decay over each update. Whereas $l1$, a regularization method that constrains the absolute value of the weights, was selected as $(1.0) \times 10^{(-5)}$, $l2$, which constrains the sum of the squared weights, was chosen as 0.0. DNN model parameters details are shown in Appendix A Table A1.

Figure 6. The daily cumulative SAG mill energy consumption values over three years, where the x-axes represent the time, and the y-axes reflect the value.

Building a new model for each chunk can provide us higher prediction performance. However, it is not efficient for the time complexity aspect since tuning and training of hyperparameters for each model separately requires extra time. In order to show the efficiency of the developed model, the conventional approach was also applied to the dataset, and the results were compared. As a traditional approach, a DNN model named Model-1 was developed based on Chunk-1 data, and the remaining 22 chunks were used as unseen testing data. Eighty percent of Chunk-1 data were used as the training set and the remaining 20% as a testing part. Table 2 illustrates Model-1's performance for each chunk. In addition, we calculated the general *MAPE* moving average to see overall model prediction performance. The last column, named Data Size, shows the sample size of each chunk.

According to Table 2, several chunks have a similar *MAPE* rate for static Model-1, indicating that several consecutive chunks have a similarity based on their error rate for the same model. However, discovering specific regimes will be changed according to the carefully chosen *MAPE* threshold value.

The MRA method uses the old models before building a new one when the *MAPE* exceeds the threshold. When the error rate is higher than the threshold value, a dynamic approach in which a new model is created immediately can be considered. However, testing the old models before creating a new one enables us to optimize the number of developed models and detect possible regime groups. Additionally, it may offer less complexity and save time in regard to the computing aspect. For this study, the *MAPE* threshold value was decided as 10%, which is accepted as high accuracy for similar research papers in the literature [45]. The MRA method applies the old models sequentially until it finds an error rate lower than the threshold value. If a satisfactory result is not found, a new model is created for the current chunk of data. The MRA method gives a regime number

based on the used model. Table 3 shows the results of the MRA method. Compared to the static approach, the results have greater accuracy as the MRA method creates a new model according to chunks with a high error rate for the current model. Chunk-18, Chunk-20, and Chunk-23 exceed the *MAPE* threshold for the current model, and they are shown in bold in Table 3.

Table 2. Results for static Model-1 (Traditional Approach).

Chunk No.	RMSE (kW)	MAE (kW)	MAPE	General MAPE Moving Average	Data Size
Chunk-1	558.606	399.851	4.07%	4.07%	884
Chunk-2	482.829	377.286	3.76%	3.92%	1134
Chunk-3	619.539	508.141	5.27%	4.37%	1006
Chunk-4	636.746	478.183	6.72%	4.96%	918
Chunk-5	471.063	385.239	3.77%	4.72%	449
Chunk-6	599.276	477.449	5.03%	4.77%	2338
Chunk-7	501.044	383.753	3.91%	4.65%	381
Chunk-8	927.837	723.436	8.02%	5.07%	535
Chunk-9	855.624	701.537	7.74%	5.37%	919
Chunk-10	857.830	672.027	7.28%	5.56%	624
Chunk-11	891.772	704.841	8.07%	5.79%	435
Chunk-12	531.324	421.940	4.46%	5.68%	306
Chunk-13	510.956	387.609	3.99%	5.55%	906
Chunk-14	700.204	544.330	5.22%	5.52%	689
Chunk-15	496.442	385.021	3.84%	5.41%	1138
Chunk-16	957.800	811.572	8.63%	5.61%	2871
Chunk-17	685.724	501.897	5.27%	5.59%	1711
Chunk-18	1644.924	1464.499	16.67%	6.21%	1272
Chunk-19	1647.166	1537.029	17.76%	6.81%	406
Chunk-20	1771.213	1556.966	18.42%	7.40%	1719
Chunk-21	1718.737	1616.370	18.36%	7.92%	462
Chunk-22	1767.231	1628.439	18.43%	8.40%	959
Chunk-23	1053.187	811.229	7.33%	8.35%	228

Table 3. Results for the MRA method.

Chunk No.	RMSE (kW)	MAE (kW)	MAPE < Threshold (10%)	General MAPE Moving Average	Data Size	Regime-Model No.	Are There Any Old Models?
Chunk-1	558.606	399.851	4.07%	4.07%	884	1	No
Chunk-2	482.829	377.286	3.76%	3.92%	1134	1	No
Chunk-3	619.539	508.141	5.27%	4.37%	1006	1	No
Chunk-4	636.746	478.183	6.72%	4.96%	918	1	No
Chunk-5	471.063	385.239	3.77%	4.72%	449	1	No
Chunk-6	599.276	477.449	5.03%	4.77%	2338	1	No
Chunk-7	501.044	383.753	3.91%	4.65%	381	1	No
Chunk-8	927.837	723.436	8.02%	5.07%	535	1	No
Chunk-9	855.624	701.537	7.74%	5.37%	919	1	No
Chunk-10	857.830	672.027	7.28%	5.56%	624	1	No
Chunk-11	891.772	704.841	8.07%	5.79%	435	1	No
Chunk-12	531.324	421.940	4.46%	5.68%	306	1	No
Chunk-13	510.956	387.609	3.99%	5.55%	906	1	No
Chunk-14	700.204	544.330	5.22%	5.52%	689	1	No
Chunk-15	496.442	385.021	3.84%	5.41%	1138	1	No
Chunk-16	957.800	811.572	8.63%	5.61%	2871	1	No
Chunk-17	685.724	501.897	5.27%	5.59%	1711	1	No
Chunk-18	1644.924	1464.499	16.67%	6.21%	1272	1	No

Table 3. Cont.

Chunk No.	RMSE (kW)	MAE (kW)	MAPE < Threshold (10%)	General MAPE Moving Average	Data Size	Regime-Model No.	Are There Any Old Models?
Chunk-18	450.178	331.914	3.70%	5.49%	1272	2	Yes
Chunk-19	387.165	287.935	3.17%	5.36%	406	2	Yes
Chunk-20	1172.610	981.431	10.67%	5.63%	1719	2	Yes
Chunk-20	1771.213	1556.966	18.42%	6.02%	1719	1	No
Chunk-20	465.403	345.34	3.99%	5.30%	1719	3	Yes
Chunk-21	623.416	525.236	5.74%	5.32%	462	3	Yes
Chunk-22	1125.931	784.768	8.20%	5.45%	959	3	Yes
Chunk-23	2354.083	2272.379	20.14%	6.09%	228	3	Yes
Chunk-23	1053.187	811.229	7.33%	5.53%	228	1	Yes

The MRA method enhances the prediction quality due to the dynamic model approach. The machine may have several running modes for different input combinations, and the MRA method assists in discovering those distinct potential regimes by predicting the energy consumption more precisely compared to the traditional method. For Chunk-23, it can be seen that the MRA method used Model-1 and achieved a *MAPE* lower than the threshold, which is an example of a repetitive regime. However, Chunk-18 and Chunk-20 required a new model based on the agreed threshold. Figure 7 illustrates the MRA method's prediction performance for each distinctive chunk from traditional approach.

Figure 7. Prediction performance for the MRA method over the distinctive chunks, where the *x*-axes represent the time, and the *y*-axes reflect the value.

According to the results, most new regimes occurred in the last year of the dataset, which is from Chunk-16 to Chunk-23. A performance comparison of the traditional approach and the MRA method for the last eight chunks is shown in Figure 8. The results show that the MRA method outperformed the traditional approach.

Applying old models rather than building a new one has several advantages. First, it facilitates detection of possible repetitive regimes. Second, where the traditional approach's general prediction performance *MAPE* rate was around 8.35%, the MRA method's general prediction performance was around 5.53%. It also reduces the total number of models by applying old models before building a new one. As a result, the MRA method provides a better prediction performance for the energy consumption of the SAG Mill with the detection of potential repetitive regime areas.

Figure 8. Comparison of the traditional approach and the MRA method, where the x -axes represent the time and the y -axes reflect the value.

5. Conclusions and Future Work

In this paper, a novel data-driven method named the MRA method was developed to predict the energy consumption of a SAG mill. The MRA method allows us to discover potential change points over time and enhance the prediction performance. In addition, the performance of the proposed method was compared with the traditional approach. The MRA method reduces the overall error rate and is useful in finding repetitive regimes. Furthermore, we also showed the importance of understanding the dataset rather than just focusing on the quality of the prediction models. More complex systems, such as industrial machines, require more interdisciplinary solutions to obtain better prediction results.

It is obvious that typical machine learning algorithms assuming data are stationary have difficulty with real-world variations in streaming data. Enormous numbers of data can be generated, necessitating dispersed processing over time. The results show that the proposed model effectively predicts the energy consumption of industrial machines with concept drift difficulties. Instead of using a traditional static model, which does not provide acceptable prediction performance after a concept drift occurs, the MRA method detects concept drift points over time to maintain highly accurate prediction performance thanks to the dynamic approach. In addition, the results proved that instead of using a fixed size of chunks, separating the data based on machine types and working conditions is more efficient to discover concept drift points.

In future work, the MRA method could be applied to different industrial machines' time series to see whether there is an improvement in energy consumption prediction. Additionally, the dataset has one-hour time interval records, but different time interval records may boost the accuracy of the results. Finally, in this research, a DNN was used as a prediction model. However, several prediction methods, such as SVR or RF, can be integrated into the MRA method and possibly increase the prediction performance.

Author Contributions: Conceptualization, A.K., M.K. (Mehmed Kantardzic); methodology, A.K., M.K. (Mehmed Kantardzic); software, A.K., M.K. (Muhammed Kotan); validation, M.K. (Mehmed Kantardzic), M.M.K.; formal analysis, A.K., M.K. (Mehmed Kantardzic), M.M.K., M.K. (Muhammed Kotan); investigation, A.K.; resources, A.K.; data curation, M.M.K.; writing—original draft preparation, A.K., M.K. (Muhammed Kotan); writing—review and editing, A.K., M.M.K., M.K. (Muhammed Kotan); visualization, A.K., M.K. (Muhammed Kotan); supervision, M.K. (Mehmed Kantardzic); project administration, A.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Nomenclature

ANN	Artificial neural network
β	Bias
C	Chunk no
CO ₂	Carbon dioxide
dB	Decibel
DNN	Deep neural network
Err	Error rate
E	Error rate of the neural network
f	activation function
h	The number of hidden nodes
kWh	Kilowatt-hour
M	Model no
MAE	Mean absolute error
MAPE	Mean absolute percentage error
MRA	Multi-Regime approach
n	The number of input nodes
NC	Total number of chunks
NM	Number of models
psi	Pounds per square inch
PSF	Pattern sequence-based forecasting
ReLU	Rectified linear unit
RF	Random forest
RMSE	Root mean square error
SAG	Semi-autonomous grinding mill
SME	Subject matter expert
SVR	Support vector regression
S_{OV}	The sum of the output variable
t	Time index
tanh	Hyperbolic tangent function
Th	Threshold value
TPH	Ton per hour
O_V	Mill energy consumption values
W_b, W_j, W_{bj}, W_{ij}	The weights for the neural network connections
Wt	Window timing
y_t	The output variable
y_{t-i}	Input variables

Appendix A

In Table A1, a DNN model was developed for each chunk of data separately, which is referred to as fully dynamic modeling. The results were shared as *RMSE*, *MAE*, and *MAPE*. Additionally, distinctive parameters of each DNN model's details are given in the last column.

Figure A1 illustrates each chunk's actual and predicted energy consumption according to the chunk number. All *x*-axes illustrate the time index, and *y*-axes show the amount of energy consumption within an hour.

Table A1. Results for fully dynamic modeling with DNN model details.

Chunk No.	RMSE (kW)	MAE (kW)	MAPE	Model No.	Data Size	DNN Model Details
Chunk-1	558.606	399.851	4.07%	1	884	4 layers each with 50 neurons, 80% training, 20% testing
Chunk-2	369.197	277.147	2.75%	2	1134	3 layers each with 50 neurons, 80% training, 20% testing
Chunk-3	374.185	289.057	2.96%	3	1006	3 layers each with 100 neurons, 80% training, 20% testing
Chunk-4	286.719	234.354	2.27%	4	918	5 layers each with 50 neurons, 80% training, 20% testing
Chunk-5	251.755	191.603	1.85%	5	449	4 layers each with 50 neurons, 70% training, 30% testing
Chunk-6	391.644	305.965	3.17%	6	2338	5 layers each with 50 neurons, 80% training, 20% testing
Chunk-7	434.867	290.972	3.05%	7	381	4 layers each with 100 neurons, 70% training, 30% testing
Chunk-8	374.29	270.084	2.86%	8	535	4 layers each with 150 neurons, 70% training, 30% testing
Chunk-9	455.702	338.252	3.61%	9	919	4 layers each with 50 neurons, 80% training, 20% testing
Chunk-10	293.248	235.917	2.48%	10	624	4 layers each with 50 neurons, 70% training, 30% testing
Chunk-11	471.422	354.039	3.87%	11	435	3 layers each with 50 neurons, 70% training, 30% testing
Chunk-12	336.583	261.434	2.74%	12	306	3 layers each with 100 neurons, 70% training, 30% testing
Chunk-13	372.154	266.002	2.68%	13	906	5 layers each with 50 neurons, 80% training, 20% testing
Chunk-14	527.264	389.147	3.90%	14	689	4 layers each with 150 neurons, 70% training, 30% testing
Chunk-15	397.105	310.283	3.06%	15	1138	4 layers each with 50 neurons, 80% training, 20% testing
Chunk-16	405.892	310.186	3.19%	16	2871	4 layers each with 50 neurons, 70% training, 30% testing
Chunk-17	488.183	339.273	3.51%	17	1711	3 layers each with 100 neurons, 70% training, 30% testing
Chunk-18	450.178	331.914	3.70%	18	1272	4 layers each with 100 neurons, 70% training, 30% testing
Chunk-19	334.007	253.484	2.75%	19	406	4 layers each with 50 neurons, 70% training, 30% testing
Chunk-20	465.403	345.34	3.99%	20	1719	4 layers each with 100 neurons, 70% training, 30% testing
Chunk-21	283.137	227.15	2.53%	21	462	5 layers each with 100 neurons, 70% training, 30% testing
Chunk-22	596.904	407.922	4.48%	22	959	5 layers each with 100 neurons, 70% training, 30% testing
Chunk-23	272.855	209.01	1.83%	23	228	3 layers each with 100 neurons, 70% training, 30% testing

Figure A1. Prediction performance for static Model-1 (Traditional Approach), where the x -axes represent the time and the y -axes reflect the value.

References

- Kim, J.Y.; Cho, S.B. Electric energy consumption prediction by deep learning with state explainable autoencoder. *Energies* **2019**, *12*, 739. [\[CrossRef\]](#)
- Zhao, G.; Liu, Z.; He, Y.; Cao, H.; Guo, Y. Energy consumption in machining: Classification, prediction, and reduction strategy. *Energy* **2017**, *133*, 142–157. [\[CrossRef\]](#)
- Kim, M.; Choi, W.; Jeon, Y.; Liu, L. A hybrid neural network model for power demand forecasting. *Energies* **2019**, *12*, 931. [\[CrossRef\]](#)
- Tan, M.; Yuan, S.; Li, S.; Su, Y.; Li, H.; He, F. Ultra-short-term industrial power demand forecasting using LSTM based hybrid ensemble learning. *IEEE Trans. Power Syst.* **2019**, *35*, 2937–2948. [\[CrossRef\]](#)
- Li, K.; Xue, W.; Tan, G.; Denzer, A.S. A state of the art review on the prediction of building energy consumption using data-driven technique and evolutionary algorithms. *Build. Serv. Eng. Res. Technol.* **2020**, *41*, 108–127. [\[CrossRef\]](#)
- Wang, X.; Yi, J.; Zhou, Z.; Yang, C. Optimal Speed Control for a Semi-Autogenous Mill Based on Discrete Element Method. *Processes* **2020**, *8*, 233. [\[CrossRef\]](#)
- Avalos, S.; Kracht, W.; Ortiz, J.M. Machine learning and deep learning methods in mining operations: A data-driven SAG mill energy consumption prediction application. *Min. Metall. Explor.* **2020**, *37*, 1197–1212.
- Silva, M.; Casali, A. Modelling SAG milling power and specific energy consumption including the feed percentage of intermediate size particles. *Miner. Eng.* **2015**, *70*, 156–161. [\[CrossRef\]](#)
- Curilem, M.; Acuña, G.; Cubillos, F.; Vyhmeister, E. Neural networks and support vector machine models applied to energy consumption optimization in semiautogenous grinding. *Chem. Eng. Trans.* **2011**, *25*, 761–766.
- Yuwen, C.; Sun, B.; Liu, S. A Dynamic Model for a Class of Semi-Autogenous Mill Systems. *IEEE Access* **2020**, *8*, 98460–98470. [\[CrossRef\]](#)

11. Hoseinian, F.S.; Abdollahzadeh, A.; Rezai, B. Semi-autogenous mill power prediction by a hybrid neural genetic algorithm. *J. Cent. South Univ.* **2018**, *25*, 151–158. [\[CrossRef\]](#)
12. Hoseinian, F.S.; Faradonbeh, R.S.; Abdollahzadeh, A.; Rezai, B.; Soltani-Mohammadi, S. Semi-autogenous mill power model development using gene expression programming. *Powder Technol.* **2017**, *308*, 61–69. [\[CrossRef\]](#)
13. Jnr, W.V.; Morrell, S. The development of a dynamic model for autogenous and semi-autogenous grinding. *Miner. Eng.* **1995**, *8*, 1285–1297.
14. Hill, T.; O'Connor, M.; Remus, W. Neural network models for time series forecasts. *Manag. Sci.* **1996**, *42*, 1082–1092. [\[CrossRef\]](#)
15. Park, J.; Law, K.H.; Bhinge, R.; Biswas, N.; Srinivasan, A.; Dornfeld, D.A.; Helu, M.; Rachuri, S. A generalized data-driven energy prediction model with uncertainty for a milling machine tool using Gaussian Process. In Proceedings of the International Manufacturing Science and Engineering Conference, Charlotte, NC, USA, 8–12 June 2015; American Society of Mechanical Engineers: Charlotte, NC, USA, 2015; Volume 56833, p. V002T05A010.
16. Ceci, M.; Corizzo, R.; Japkowicz, N.; Mignone, P.; Pio, G. Echad: Embedding-based change detection from multivariate time series in smart grids. *IEEE Access* **2020**, *8*, 156053–156066. [\[CrossRef\]](#)
17. Xu, W.; Peng, H.; Zeng, X.; Zhou, F.; Tian, X.; Peng, X. A hybrid modelling method for time series forecasting based on a linear regression model and deep learning. *Appl. Intell.* **2019**, *49*, 3002–3015. [\[CrossRef\]](#)
18. Singh, S.; Yassine, A. Big data mining of energy time series for behavioral analytics and energy consumption forecasting. *Energies* **2018**, *11*, 452. [\[CrossRef\]](#)
19. Demirel, Ö.F.; Zaim, S.; Çalışkan, A.; Özuyar, P. Forecasting natural gas consumption in Istanbul using neural networks and multivariate time series methods. *Turk. J. Electr. Eng. Comput. Sci.* **2012**, *20*, 695–711.
20. Torres, J.F.; Hadjout, D.; Sebaa, A.; Martínez-Álvarez, F.; Troncoso, A. Deep Learning for Time Series Forecasting: A Survey. *Big Data* **2021**, *9*, 3–21. [\[CrossRef\]](#)
21. Sezer, O.B.; Gudelek, M.U.; Ozbayoglu, A.M. Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Appl. Soft Comput.* **2020**, *90*, 106181. [\[CrossRef\]](#)
22. Mishra, S.; Bordin, C.; Taharaguchi, K.; Palu, I. Comparison of deep learning models for multivariate prediction of time series wind power generation and temperature. *Energy Rep.* **2020**, *6*, 273–286. [\[CrossRef\]](#)
23. Manero, J.; Béjar, J.; Cortés, U. “Dust in the wind...”, deep learning application to wind energy time series forecasting. *Energies* **2019**, *12*, 2385. [\[CrossRef\]](#)
24. Xiao, J.; Li, Y.; Xie, L.; Liu, D.; Huang, J. A hybrid model based on selective ensemble for energy consumption forecasting in China. *Energy* **2018**, *159*, 534–546. [\[CrossRef\]](#)
25. Liu, X.; Chen, R. Threshold factor models for high-dimensional time series. *J. Econom.* **2020**, *216*, 53–70. [\[CrossRef\]](#)
26. Battaglia, F.; Protopapas, M.K. Multi-regime models for nonlinear nonstationary time series. *Comput. Stat.* **2012**, *27*, 319–341. [\[CrossRef\]](#)
27. Hu, H.; Kantardzic, M.; Sethi, T.S. No Free Lunch Theorem for concept drift detection in streaming data classification: A review. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2020**, *10*, e1327. [\[CrossRef\]](#)
28. McCandless, T.; Dettling, S.; Haupt, S.E. Comparison of implicit vs. explicit regime identification in machine learning methods for solar irradiance prediction. *Energies* **2020**, *13*, 689. [\[CrossRef\]](#)
29. Lu, Z.; Xia, J.; Wang, M.; Nie, Q.; Ou, J. Short-term traffic flow forecasting via multi-regime modeling and ensemble learning. *Appl. Sci.* **2020**, *10*, 356. [\[CrossRef\]](#)
30. Divina, F.; Garcia Torres, M.; Gómez Vela, F.A.; Vazquez Noguera, J.L. A comparative study of time series forecasting methods for short term electric energy consumption prediction in smart buildings. *Energies* **2019**, *12*, 1934. [\[CrossRef\]](#)
31. Neto, A.H.; Fiorelli, F.A.S. Comparison between detailed model simulation and artificial neural network for forecasting building energy consumption. *Energy Build.* **2008**, *40*, 2169–2176. [\[CrossRef\]](#)
32. Hamzaçebi, C. Improving artificial neural networks’ performance in seasonal time series forecasting. *Inf. Sci.* **2008**, *178*, 4550–4559. [\[CrossRef\]](#)
33. Alvarez, F.M.; Troncoso, A.; Riquelme, J.C.; Ruiz, J.S.A. Energy time series forecasting based on pattern sequence similarity. *IEEE Trans. Knowl. Data Eng.* **2010**, *23*, 1230–1243. [\[CrossRef\]](#)
34. Tso, G.K.; Yau, K.K. Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks. *Energy* **2007**, *32*, 1761–1768. [\[CrossRef\]](#)
35. Hu, Y.C. Electricity consumption prediction using a neural-network-based grey forecasting approach. *J. Oper. Res. Soc.* **2017**, *68*, 1259–1264. [\[CrossRef\]](#)
36. Kant, G.; Sangwan, K.S. Predictive modelling for energy consumption in machining using artificial neural network. *Procedia CIRP* **2015**, *37*, 205–210. [\[CrossRef\]](#)
37. U.S. Energy Information Administration (EIA). Available online: www.eia.gov (accessed on 9 June 2021).
38. Kankal, M.; Akpınar, A.; Kömürçü, M.İ.; Özşahin, T.Ş. Modeling and forecasting of Turkey’s energy consumption using socio-economic and demographic variables. *Appl. Energy* **2011**, *88*, 1927–1939. [\[CrossRef\]](#)
39. Azadeh, A.; Ghaderi, S.F.; Tarveredian, S.; Saberi, M. Integration of artificial neural networks and genetic algorithm to predict electrical energy consumption. *Appl. Math. Comput.* **2007**, *186*, 1731–1741. [\[CrossRef\]](#)
40. Wang, J.Q.; Du, Y.; Wang, J. LSTM based long-term energy consumption prediction with periodicity. *Energy* **2020**, *197*, 117197. [\[CrossRef\]](#)

-
41. He, Y.; Wu, P.; Li, Y.; Wang, Y.; Tao, F.; Wang, Y. A generic energy prediction model of machine tools using deep learning algorithms. *Appl. Energy* **2020**, *275*, 115402. [[CrossRef](#)]
 42. Chen, C.; Liu, Y.; Kumar, M.; Qin, J. Energy consumption modelling using deep learning technique—A case study of EAF. *Procedia CIRP* **2018**, *72*, 1063–1068. [[CrossRef](#)]
 43. Lin, L.; Wang, F.; Xie, X.; Zhong, S. Random forests-based extreme learning machine ensemble for multi-regime time series prediction. *Expert Syst. Appl.* **2017**, *83*, 164–176. [[CrossRef](#)]
 44. Khashei, M.; Bijari, M. An artificial neural network (p, d, q) model for timeseries forecasting. *Expert Syst. Appl.* **2010**, *37*, 479–489. [[CrossRef](#)]
 45. Wei, N.; Li, C.; Peng, X.; Zeng, F.; Lu, X. Conventional models and artificial intelligence-based models for energy consumption forecasting: A review. *J. Pet. Sci. Eng.* **2019**, *181*, 106187. [[CrossRef](#)]