


Article

A Spatio-Temporal Deep Learning Network for the Short-Term Energy Consumption Prediction of Multiple Nodes in Manufacturing Systems

Jianhua Guo ^{*}, Mingdong Han, Guozhi Zhan and Shaopeng Liu

School of Computer Science, Guangdong Polytechnic Normal University, Guangzhou 510665, China; hanmingdong173@163.com (M.H.); guozhi_zhan@163.com (G.Z.); l-shaopeng@live.cn (S.L.)

* Correspondence: gjh209@163.com; Tel.: +86-020-3825-6730

Abstract: Short-term energy prediction plays an important role in green manufacturing in the industrial internet environment and has become the basis of energy wastage identification, energy-saving plans and energy-saving control. However, the short-term energy prediction of multiple nodes in manufacturing systems is still a challenging issue owing to the fuzzy material flow (spatial relationship) and the dynamic production rhythm (temporal relationship). To obtain the complex spatial and temporal relationships, a spatio-temporal deep learning network (STDLN) method is presented for the short-term energy consumption prediction of multiple nodes in manufacturing systems. The method combines a graph convolutional network (GCN) and a gated recurrent unit (GRU) and predicts the future energy consumption of multiple nodes based on prior knowledge of material flow and the historical energy consumption time series. The GCN is aimed at capturing spatial relationships, with the material flow represented by a topology model, and the GRU is aimed at capturing dynamic rhythm from the energy consumption time series. To evaluate the method presented, several experiments were performed on the power consumption dataset of an aluminum profile plant. The results show that the method presented can predict the energy consumption of multiple nodes simultaneously and achieve a higher performance than methods based on the GRU, GCN, support vector regression (SVR), etc.

Keywords: energy consumption prediction; green manufacturing; deep learning network; graph convolutional network; gated recurrent unit



Citation: Guo, J.; Han, M.; Zhan, G.; Liu, S. A Spatio-Temporal Deep Learning Network for the Short-Term Energy Consumption Prediction of Multiple Nodes in Manufacturing Systems. *Processes* **2022**, *10*, 476. <https://doi.org/10.3390/pr10030476>

Academic Editor: Jiaqiang E

Received: 13 February 2022

Accepted: 25 February 2022

Published: 26 February 2022

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



Copyright: © 2022 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

In the context of global warming and fossil fuel depletion, green manufacturing has drawn more and more attention. Energy saving in manufacturing processes is one of the core goals of green manufacturing, and it has been well supported by industrial internets, which enable real-time collection of energy consumption from smart energy meters. Short-term (i.e., hourly or minute-by-minute) energy prediction plays an important role in green manufacturing and has become the basis of energy wastage identification, energy-saving plans and energy-saving control [1–4]. However, short-term energy prediction in manufacturing systems is still a challenging issue owing to its complex spatial and temporal relationships between production nodes.

The spatial relationship means the topological structure composed of the material flow. From the perspective of machines, the production status at the upstream production node is transferred to the downstream production node [5]. From the perspective of management, the production node is a hierarchical concept composed of a plant, workshop, line, machine group or individual machine [2,6,7], and the energy consumption at the upper production node is transferred to the lower production node. Moreover, these transfers are very fuzzy in many flexible manufacturing systems [3,8]. In general, the output from one upstream/upper production node may randomly be transferred to a few

downstream/lower nodes, and one faulty production node must be frequently replaced by another parallel node.

The temporal relationship means the dynamic production rhythm that reflects production periodicity and trend. The energy consumption time series at different granularities can reflect different production laws. The minute-by-minute series of machine nodes generally reflect process law, and the hourly series of workshop nodes generally reflect worker law [9]. These production laws are easily affected by machine failure, lack of material, changeovers, etc.

According to input data, the energy consumption prediction methods for manufacturing systems can be categorized into the parametric method and the sequential method. The former generally inputs impact parameters such as yield parameters, technological parameters, weather parameters, etc., and tries to capture the relationship between input and output through a building mechanism model or a machine learning model [1,10,11]. Generally, the relationship between parameters and energy consumption can be regarded as a kind of spatial relationship between the machine and its surroundings. The authors [1,12] presented a parametric method for energy consumption prediction of tire vulcanization, in which a few technological parameters were inputted and which was then used to detect heat loss anomalies during the process. However, we find that such methods generally require rich prior technical knowledge and complex multi-data fusion, and hence they are hard to implement in many real cases. Moreover, their prediction precision may be unsatisfactory owing to ignoring of the temporal relationship.

The sequential methods depend on the historical time series of energy consumption, and they are still prevalent owing to high flexibility and availability. The classical sequential methods include the autoregressive integrated moving average (ARIMA) model [13], the XGBoost model [14], the Kalman filtering model [15], the support vector regression (SVR) model [16–18] and the back-propagation (BP) neural network model [19]. However, the above methods cannot fit well with capturing complex features from big data. In recent years, deep learning networks have become very prevalent owing to their strong feature extraction capabilities from big data.

A recurrent neural network (RNN) [20] is a typical deep learning network for feature capturing from time series. In this network, hidden unit patterns are fed back to themselves and previous states are memorized. However, it is not able to keep long-term memory information since the gradients tend to either vanish or explode. To this end, two improved RNNs, long short-term memory (LSTM) [21] and gated recurrent unit (GRU) [22,23], were proposed. In LSTM, three gates, i.e., input gate, output gate and forget gate, are added to serve as three switches. They determine what is to be memorized or forgotten and facilitate the keeping, utilizing, or destroying of a previous state in an appropriate time. The GRU can be regarded as an updated version of LSTM with a simpler structure. It merges the input gate and the forget gate into the update gate while the memory units and hidden units are also combined, and modulates the information flow inside the units without separated memory cells [24].

In recent years, LSTM, GRU and their improvements have been widely applied in energy consumption prediction of time series and other industrial data predictions. Some methods focus on the prediction of a single node considering only the temporal relationship. Wang et al. [25] implemented a day-ahead photovoltaic power prediction through integrating LSTM with time correlation principles. He and Tsang [26] achieved short-term load prediction of colleges and universities through integrating LSTM with the periodic pattern decomposition of time series. Similar methods based on LSTM and pattern decomposition can be seen in the energy prediction of solar-assisted water heating systems [27], in regional natural gas consumption prediction [28], in the power prediction of universities [26], in the heating energy prediction of non-residential buildings [29], etc. Wang, Yan, Li, Gao and Zhao [24] integrated local feature knowledge into a deep heterogeneous GRU model and implemented tool wear prediction in manufacturing. The above methods can achieve high prediction precision for time series through extracting

local features and then reducing local noise, but they did not consider spatial relationships between multiple nodes and have not been applied in the simultaneous prediction of multiple nodes. For a manufacturing system with hundreds or thousands of nodes, it is too time-consuming to build a prediction model for each node.

Some methods consider the spatial relationship in the time-series prediction of multiple nodes. Liu et al. [30] proposed a dual-stage two-phase attention-based RNN for long-term and multivariate time-series prediction, in which spatial correlations are captured through adding attention to the node dimensions. The method was employed to predict the consumption energy of multiple appliances in a low-energy building. A few methods integrated convolutional neural network (CNN) with GRU (or LSTM) to predict the energy consumption time series of multiple nodes, and they have been applied in regional integrated energy systems [31–34] and building energy management systems [35]. In these methods, the spatial relationship is extracted through a convolutional operation on high-dimensional time series. However, the CNN is commonly used for Euclidean data such as images, regular grids and so on, and cannot work well in the context of a manufacturing system with a complex topological structure. Thus, the CNN is inapposite for describing spatial relationships. On the other hand, an attention mechanism is commonly used for sequence data with clear contexts such as nature language processes and cannot work well in the context of manufacturing systems without describing topological structure.

Therefore, short-term energy consumption prediction of multiple nodes in manufacturing systems is still a challenging issue. The main challenge of the issue lies in the extraction of spatial relationships and in spatio-temporal collaborative learning. Graph convolutional networks (GCNs) [36] inspired the solution for spatial-relationship learning issues. The GCN presented a scalable approach for learning on graph-structured data based on an efficient variant of convolutional neural networks which operate directly on graphs and can be used to capture the structural features of a graph network. Initially, the GCN was designed for semi-supervised classification of graph data. Currently, it is being extended to time-series prediction through integration with RNN, LSTM and GRU [37,38], and has presented spatio-temporal graph convolutional networks for forecasting traffic flow. In the two methods, the GCN is combined with the GRU, the GCN is used to learn complex topological structures to capture spatial dependence and the GRU is used to learn dynamic changes of traffic data to capture temporal dependence. The framework combining the GCN with the GRU has good transferability to the short-term energy consumption prediction of multiple nodes. However, different from traffic systems with visible topological structure, manufacturing systems have fuzzy topological structures and consume various forms of energy, and hence both the topological model and the learning algorithm need to be reconstructed for the latter.

Based on the above background, this paper focuses on short-term energy consumption prediction of multiple nodes in manufacturing systems. Three hypotheses are under consideration. First, a manufacturing system is a hierarchical structure composed of a workshop, line, machine group and individual machine. Second, material flow in the manufacturing system is a mixture mode consisting of parallel machines and flow lines. Third, data on energy consumption of production nodes are collected minute-by-minute, and prediction depends only on the collected time series. Thereafter, a spatio-temporal deep learning network (STDLN) combining the GCN with the GRU is presented. The main contributions of this paper are as follows:

- (1) A topology modeling method is presented to make clear the topological structure of the manufacturing system. The method constructs the explicit spatial relationships between the production nodes through hierarchically abstracting parallel machines and flow lines and finally describes it as an adjacency matrix.

- (2) A deep learning network combining the GCN with the GRU is presented. The GCN is used to capture the implicit spatial relationship between production nodes through learning based on the adjacency matrix and historical time series. The GRU is used to capture the temporal relationship through learning based on the time series from the GCN.

(3) The case of an aluminum profile plant is provided to evaluate the model presented. A total of 140 nodes were chosen and the power consumption dataset was extracted from the energy management system of the plant studied. Several experiments were performed, and the results show that the method presented can predict the energy consumption of multiple nodes simultaneously, and can achieve a higher performance than models based on the GRU, the GCN, support vector regression (SVR), etc.

The rest of this paper is organized as follows. The methods are presented in Section 2. The experiments and results are described in Section 3. The interpretations of the experimental results are discussed in Section 4, and the conclusions drawn are presented in Section 5.

2. Methods

2.1. Problem Definition

Given a manufacturing system with prior knowledge of the technology process and minute-by-minute energy consumption time series of production nodes, the goal of this paper is to predict the energy consumption of multiple nodes in a certain time period. The prediction is short-term (minute-by-minute or hourly) and mainly used to support energy wastage identification, energy-saving plans and energy-saving control.

Definition 1. *Production Nodes V .* A production node means a workshop, a line, a machine group or an individual machine, and it is associated with an energy consumption meter. $V = \{v_1, v_2, \dots, v_N\}$, V is the set of production nodes in the manufacturing system, N is the size of V and v_i is the i th production node.

Definition 2. *Edges between Production Nodes E .* $E = \{e_{ij} \mid v_i, v_j \in V, i \neq j\}$, where e_{ij} is the edge between v_i and v_j , which only exists if there is direct material flow between v_i and v_j .

Definition 3. *Energy Consumption Time Series $X^{N \times P}$.* $X = \{x_{it} \mid v_i \in V, t = 0, 1, \dots, P\}$, where x_{it} is the energy consumption value of v_i at t th time, P is the length of the time series, $X \in \mathbf{R}^{N \times P}$. For simplicity, it is assumed that only one type of energy needs to be predicted for one production node.

Definition 4. *Material Flow Network G .* An unweighted graph $G = (V, E)$ is used to describe the topological structure of the material flow network of the manufacturing system. G can be represented by an adjacency matrix $A \in \mathbf{R}^{N \times N}$ which contains only elements of 0 and 1: the element in the i th row and the j th is 1 if e_{ij} exists, and 0 otherwise.

Therefore, the short-term energy consumption prediction of multiple nodes in the manufacturing system can be described as Equation (1):

$$[X_{t+1}, X_{t+2}, \dots, X_{t+T}] = f(A; (X_{t-n}, X_{t-n+1}, \dots, X_t)) \quad (1)$$

where $X_t \in \mathbf{R}^N$ is the energy consumption values of V at t -th time, n is the length of the historical time series and T is the length of the time series that needs to be predicted.

2.2. Framework of Method

Based on the definitions above, the framework of the STDLN method is designed and shown in Figure 1. The method consists of three tasks: topology modeling, spatial relationship learning network modeling and temporal relationship learning network modeling. The topology modeling is to construct material flow network G based on prior knowledge of the technology process, and the spatial relationship learning network modeling and temporal relationship learning network modeling are to construct a prediction algorithm based on G and X . It is assumed that the data-collecting task for collecting energy consumption data from the energy meters was implemented in the industrial internet. The three tasks are described in the following subsections, respectively.

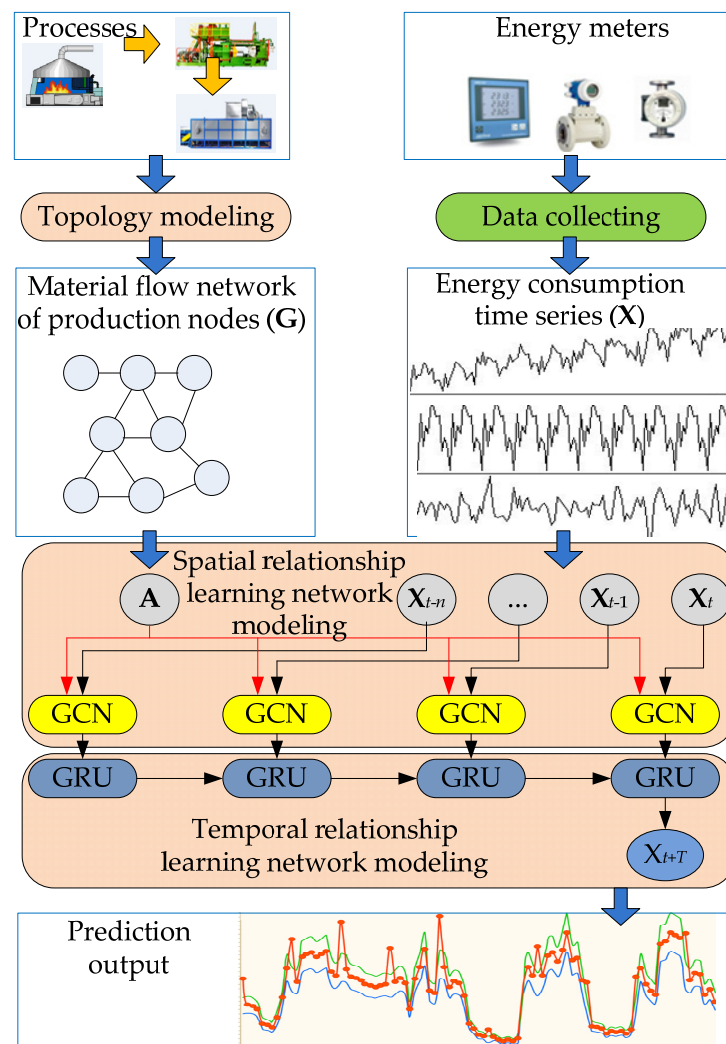


Figure 1. The framework of the STDLN method.

2.3. Topology Modeling

In general, material flow between machines is intricate and needs to be made clear through hierarchical abstracting. In production management, the manufacturing systems are categorized into flow shop system and parallel machine system based on the material flow dependency between the machines. Correspondingly, the material flow in a manufacturing system is categorized into liner flow and distributing flow in the topology modeling method. Liner flow means that the materials flow through each production node in sequence. Distributing flow means that the materials are distributed from one upper production node to multiple lower production nodes.

In this section, a 2-level abstracting method is presented to construct the topology model of a certain plant. The method works as follows:

1. Identify visible production nodes including workshops and machines and build the distributing flow between the workshop and the machine based on ownership.
2. Abstract production nodes at the workshop level.
 - (a) Merge workshops with a similar process into one process node.
 - (b) Build the distributing flow between the process node and the corresponding lower workshops.
 - (c) Build the liner flow between the workshops (or process nodes if there are merged workshops).
3. Abstract production nodes at the machine level.

- (a) Merge parallel machines into one process node and merge the flow machines into one process node.
 - (b) Build the distributing flow between the process node and the corresponding downstream machines.
 - (c) Build the liner flow between the machines (or process nodes if there are merged machines).
 - (d) Build the liner flow between the flow machines.
4. Merge redundant nodes. If one upper node owns only one lower node, the two nodes are redundant and need to be merged as one node.

After 2-level abstracting, the intricate material flow in a certain manufacturing system is decomposed into simple distributing flows and liner flows and finally converted into a composite flow.

An example of the modeling production topology is shown in Figure 2. A plant consists of three workshops with different processes, and Workshop 2 owns a production line with three machines. The three workshops are modeled by Node 1, Node 2 and Node 3. The three machines are modeled by Node 2.1, Node 2.2 and Node 2.3. The distributing flow and liner flow are shown on the left, and the composite flow is shown on the right. The composite flow is just the material flow network G and it can be represented by an adjacency matrix $A \in \mathbb{R}^{N \times N}$.

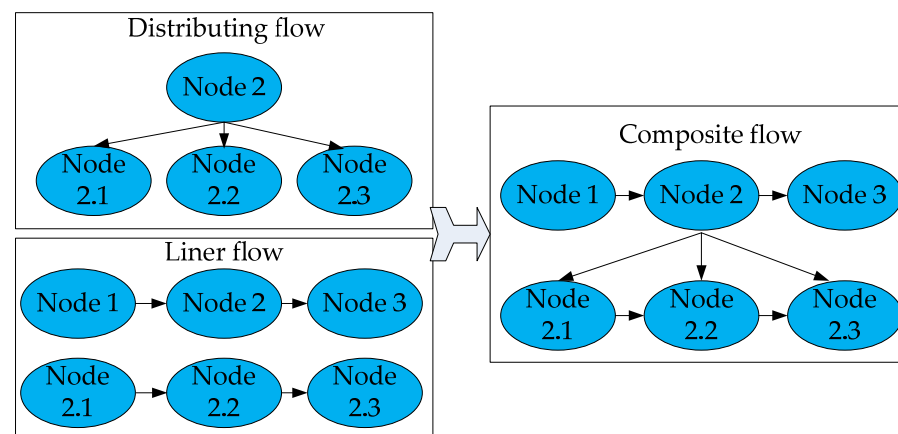


Figure 2. An example of modeling production topology.

2.4. Spatial Relationship Learning Network Modeling

The GCN has been successfully applied in traffic flow prediction [37,38] and graph node classification [36], and has shown a strong capacity for capturing spatial features. Thus, the GCN was chosen to construct the spatial relationship learning network, and the schematic depiction of the GCN for energy consumption prediction is shown in Figure 3. The input of a GCN cell includes an adjacency matrix A ($\in \mathbb{R}^{N \times N}$) and a periodical energy consumption vector X_t ($\in \mathbb{R}^N$), and the output is a vector X'_t ($\in \mathbb{R}^N$) that represents the implicit spatial relationship between production nodes. The GCN model constructs a filter in the Fourier domain, and the filter acts on the nodes of the graph and its first-order neighborhood to capture spatial features between the nodes. The GCN can be built by stacking multiple convolutional layers [36,38]. As shown in Figure 3, the GCN model can obtain the implicit spatial relationship between Node 2 and its first neighbor nodes, and it represents the energy consumption on all nodes. In fact, the adjacency matrix A represents the explicit spatial relationship between production nodes based on embedding prior knowledge. The GCN can convert X_t to X'_t and represents the implicit spatial relationship between production nodes based on machine learning.

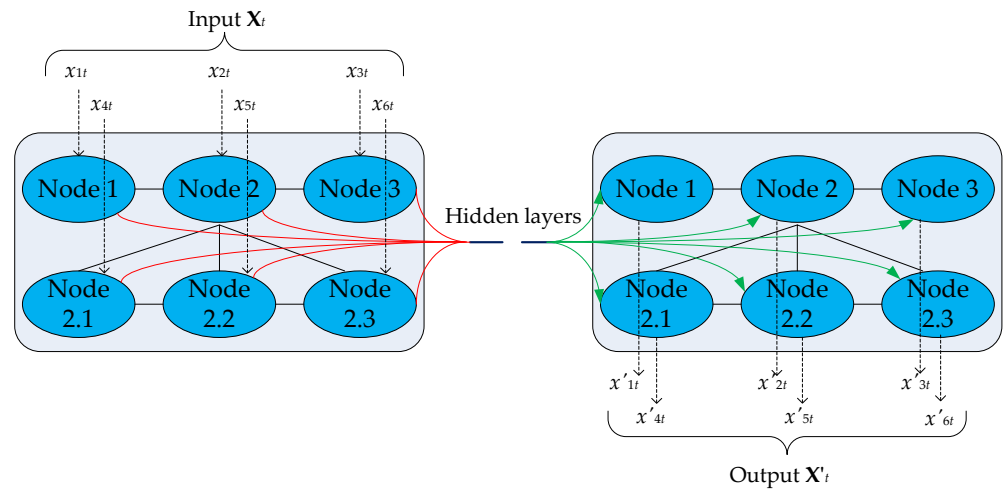


Figure 3. The schematic depiction of GCN for energy consumption prediction.

In this paper, a 2-layer GCN model is used to capture spatial relationships. The forward algorithm from \mathbf{A} and \mathbf{X}_t to \mathbf{X}'_t can be described as follows [36]:

$$\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I} \tag{2}$$

where $\tilde{\mathbf{A}}$ is a matrix with a self-connection structure, and \mathbf{I} is an identity matrix with the same size as \mathbf{A} .

$$\tilde{\mathbf{D}} = \sum_j \tilde{\mathbf{A}}_{ij} \tag{3}$$

where $\tilde{\mathbf{D}}$ is the degree matrix of $\tilde{\mathbf{A}}$.

$$\hat{\mathbf{A}} = \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \tag{4}$$

$$\mathbf{X}'_t = f(\mathbf{X}_t, \mathbf{A}) = \sigma(\hat{\mathbf{A}} \text{Relu}(\hat{\mathbf{A}} \mathbf{X}_t \mathbf{W}_0) \mathbf{W}_1) \tag{5}$$

where $\sigma(\cdot)$ and $\text{Relu}(\cdot)$ represent two activation functions, and \mathbf{W}_0 and \mathbf{W}_1 represent the weight matrix in the first and second layer, respectively.

It is not necessary to define loss function in the GCN model since the \mathbf{X}'_t is not the final output.

2.5. Temporal Relationship Learning Network Modeling

As an improved RNN, the GRU has a strong capacity for capturing temporal relationships from time series. Moreover, the GRU has simpler structure and more efficient training process than LSTM. Thus, the GRU was chosen to construct the temporal relationship learning network. The schematic depiction of the GRU for energy consumption prediction is shown in Figure 4. In the middle-GRU cell, the input \mathbf{X}'_t is the output of a GCN cell, and the output \mathbf{h}_t is the prediction vector corresponding to observing vector \mathbf{X}_{t+T} . \mathbf{u}_t and \mathbf{r}_t are the update gate and reset gate, respectively.

The forward algorithm from \mathbf{X}'_t to \mathbf{h}_t can be described as follows [22,23]:

$$\mathbf{u}_t = \sigma(\mathbf{W}_u[\mathbf{X}'_t, \mathbf{h}_{t-1}] + \mathbf{b}_u) \tag{6}$$

$$\mathbf{r}_t = \sigma(\mathbf{W}_r[\mathbf{X}'_t, \mathbf{h}_{t-1}] + \mathbf{b}_r) \tag{7}$$

$$\mathbf{c}_t = \tanh(\mathbf{W}_c[\mathbf{X}'_t, (\mathbf{r}_t * \mathbf{h}_{t-1})] + \mathbf{b}_c) \tag{8}$$

$$\mathbf{h}_t = \mathbf{u}_t * \mathbf{h}_{t-1} + (1 - \mathbf{u}_t) * \mathbf{c}_t \tag{9}$$

where W_u , W_r and W_c represent the weights in the training process, and b_u , b_r and b_c represent the deviations in the training process.

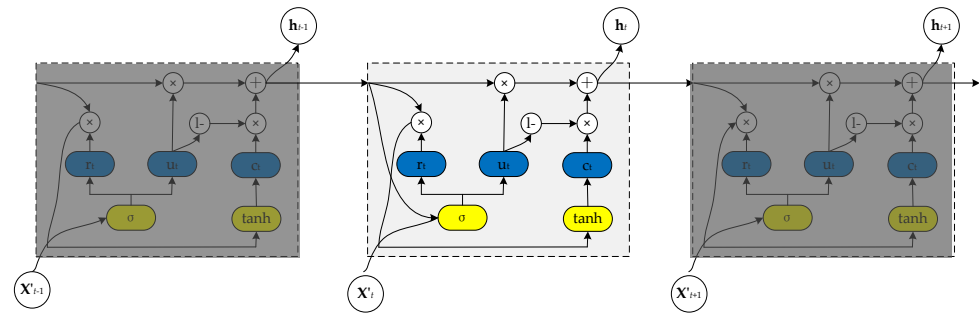


Figure 4. The schematic depiction of GRU for energy consumption prediction.

2.6. Loss Function

In the training process, the goal is to minimize the error between the observing of the energy consumption of the nodes and the predicted value. Y_t and \hat{Y}_t are used to denote the observing vector and the predicted vector of the nodes at time t . The loss function of the STDLN is defined as Equation (10).

$$loss = \sqrt{\frac{1}{NT} \sum_{i=1}^N \sum_{t=1}^T (y_{it} - \hat{y}_{it})^2} + \lambda L_{reg} \quad (10)$$

where $y_{it} \in Y_t$ represents the observing value of the i th node, $\hat{y}_{it} \in \hat{Y}_t$ represents the predicted value of the i th node, N represents the number of nodes, T is the length of the time series that needs to be predicted, L_{reg} is the L2 regularization term that helps to avoid an over-fitting problem and λ is a hyperparameter.

3. Experiments and Results

3.1. Application Case

A large-scale aluminum profile plant located in Guangdong, China, was chosen as experimental object. The plant turns aluminum ingots into aluminum profiles applied in building or industrial products, and consumes a large amount of energy through power, gas, diesel oil, water, compressed air, etc. The authors anticipated the development of its energy management system, in which over 1000 energy meters were installed and the data were collected minute-by-minute. At the same time, the authors learned about its technology and production management and were authorized to access an energy consumption dataset.

3.1.1. Technology Topology Model

According to the two-level abstracting method presented in Section 2.3, the process model of the studied plant can be described as in Figure 5. At workshop level, the processes can be abstracted as a flow line consisting of melting, extruding and surface treating. At the machine level in the melting workshop, the processes can be abstracted as a flow line consisting of a melting surface and a homogenizer. At the machine level in the extruding workshop, the processes can be abstracted as a flow line consisting of a heating surface, extruder, straightener and aging furnace. The surface treating workshop consists of four parallel workshops, namely, the oxidation workshop, the electrophoretic workshop, the painting workshop and the fluorocarbon workshop. The abbreviated names of the workshops and the machines are written in parentheses after the full names in the figure.

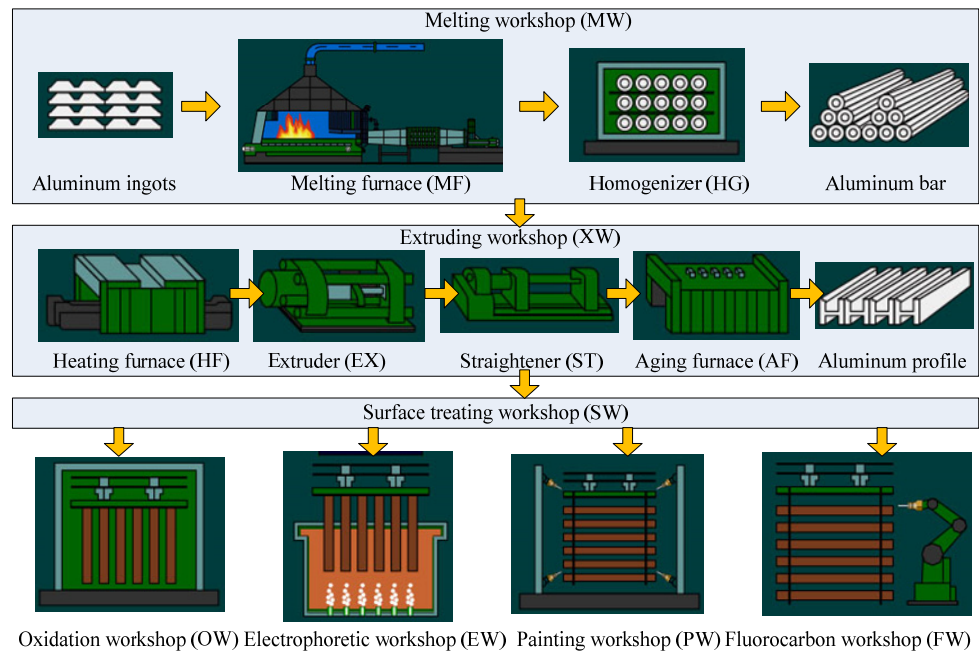


Figure 5. Process model of the studied aluminum profile plant.

The partial topology model of the studied plant is shown in Figure 6. The blue nodes are visible and the yellow are abstract. The extruding workshop (XW) node is the abstract of three parallel XWs (XW1, XW2 and XW3). Each XW node includes a flow line consisting of HF, EX, ST and AF, and an EX node includes parallel visible machines. The figure depicts a part of the topology model, and some nodes are not shown; e.g., the details of XW2 and XW3 are omitted.

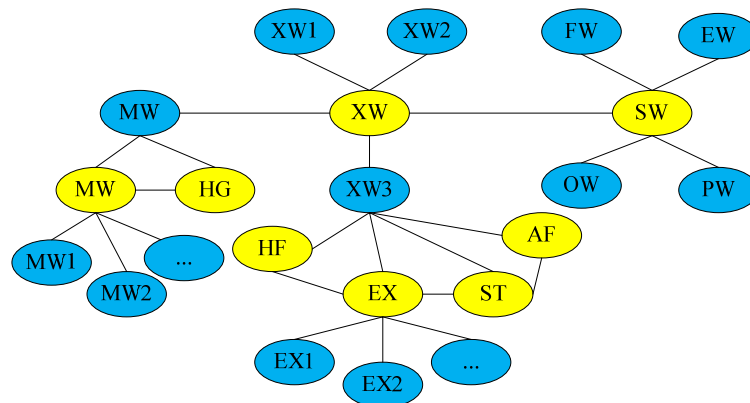


Figure 6. Partial topology model of the studied plant.

3.1.2. Time Series Datasets

To evaluate the model presented, a dataset was extracted from the energy management system of the studied plant. The dataset consists of power consumption of 140 nodes, and the time span is one month. The energy consumption of each node was collected minute-by-minute and aggregated every 5 min, 15 min, 30 min and 60 min. Each aggregated dataset can be regarded as an independent dataset, and the sets were named the 5 min set, the 15 min set, the 30 min set and the 60 min set in the ensuing experiments.

In the experiments, the input data was normalized to the interval (0, 1). For every dataset, 80% of the data was used as a training set and the remaining 20% was used as a testing set. The energy consumption in the next three periods was the predicted object.

3.2. Evaluation Metrics

To evaluate the performance of the STDLN method, four metrics were used to measure prediction performance. They are defined as follows:

(1) Root mean squared error (*RMSE*):

$$RMSE = \sqrt{\frac{1}{NT} \sum_{i=1}^N \sum_{t=1}^T (y_{it} - \hat{y}_{it})^2} \quad (11)$$

(2) Mean absolute error (*MAE*):

$$MAE = \frac{1}{NT} \sum_{i=1}^N \sum_{t=1}^T |y_{it} - \hat{y}_{it}| \quad (12)$$

(3) Accuracy (*Acc*):

$$Acc = 1 - \frac{\sqrt{\sum_{i=1}^N \sum_{t=1}^T (y_{it} - \hat{y}_{it})^2}}{\sqrt{\sum_{i=1}^N \sum_{t=1}^T y_{it}^2}} \quad (13)$$

(4) Coefficient of determination (R^2):

$$R^2 = 1 - \frac{\sum_{i=1}^N \sum_{t=1}^T (y_{it} - \hat{y}_{it})^2}{\sum_{i=1}^N \sum_{t=1}^T (y_{it} - \bar{y}_{it})^2} \quad (14)$$

where the symbols have the same meanings as in Equation (10).

Specifically, the *RMSE* and the *MAE* were used to measure the prediction error; the smaller the value, the better the prediction effect. The *Acc* was used to detect the prediction precision; the larger the value, the better the prediction effect. The R^2 was used to calculate the correlation coefficient, which measures the ability of the prediction result to represent the actual data; once more, the larger the value, the better the prediction effect [38].

3.3. Baseline Methods

The following baseline methods were used to compare the performance with the STDLN:

(1) **Autoregressive integrated moving average (ARIMA)** [13]: a well-known method for time-series prediction, which is an improvement on the autoregressive moving average model.

(2) **Support vector regression (SVR)** [18]: the kernel method based on support vectors has performed well in time-series prediction; the SVR with a radial basis function kernel was used as a the baseline method.

(3) **XGBoost** [14]: one of the most popular boosting tree algorithms for gradient boosting machines, which has been widely applied in prediction.

(4) **GCN**: Only the spatial relationship was considered. See Section 2.4 for details.

(5) **GRU**: Only the temporal relationship was considered. See Section 2.5 for details.

(6) **STDLN_F**: the STDLN with a fully connected adjacency matrix. It is assumed that there exists a spatial relation between each pair of production nodes. With the fully connected adjacency matrix, the graph convolution is similar to the convolution in CNN. Hence, the effect of the STDLN_F is also similar to combining the CNN with the GRU.

3.4. Implementation and Parameter Design

The STDLN method presented was developed using the Python language based on TensorFlow 2.0 and reusing some code provided by [38] on github.com. The baseline

methods GCN, GRU and STDLN_F were implemented through reconfiguring STDLN code, and the baseline methods ARIMA, SVR and XGBoost were developed based on TensorFlow 2.0. The experiments were performed on a super-deep learning server with NVIDIA Quadro RTX8000 48G GDDR6 GPU *4, Intel Xeon Gold 5218R CPU *2 and DDR4 32G RAM *8.

Hyperparameters such as input length, learning rate, batch size, training time and number of hidden units needed to be set in the STDLN. Empirically, input length was set to 12, learning rate to 0.001, batch size to 32 and training time to 1000. Additionally, the Adam optimizer was used in the training process.

The number of hidden units was set through dichotomy experiments. A 15 min set was chosen to evaluate the performance, and the steps are described as follows:

1. Three empirical numbers $0 < n_1 < n_2 < n_3$ were initially chosen to perform the experiments.
2. Experiments with n_1 , n_2 and n_3 were performed and their performances were evaluated.
3. The three numbers were updated in the new range according to the performance of experiments. If the best was n_1 , the new range was $(0, n_1)$; if the best was n_3 , it was $(n_3, +\infty)$; if the best was n_2 and the second best was n_1 , it was (n_1, n_2) , otherwise it was (n_2, n_3) .
4. If the new range is small enough, return to the best number; otherwise, repeat Step 2 and Step 3.

The dichotomy experiments were performed and the performance comparison of the number of hidden units is shown in Table 1. The three numbers 16, 64 and 128 were chosen for Step 1, and the new range was 64–128 based on the previous performances. The three numbers 80, 90 and 100 were chosen in the second repetition; the number 80 was the best number of hidden units and was used in the ensuing experiments.

Table 1. Performance comparison of the number of hidden units.

Number of Hidden Units	RMSE	MAE	Acc Ratio (-)	R ² Ratio (-)
16	10.160	5.269	0.939	0.996
64	8.606	4.801	0.945	0.997
80 *	6.301	4.320	0.953	0.998
90	9.560	5.108	0.940	0.996
100	8.384	4.263	0.946	0.997
128	8.741	4.261	0.944	0.997

* The best parameter was 80.

3.5. Experimental Results

The prediction performances of the STDLN method and the baseline methods on the datasets are shown in Table 2. It can be seen that the STDLN method obtained the best prediction performance under all evaluation metrics and all datasets, and the result proves the effectiveness of the STDLN method for short-term energy consumption prediction of multiple nodes in the case studied. The results were expanded according to algorithm, period and production node level in the following subsections.

3.5.1. Performance Comparison between the STDLN Method and the Traditional Methods

The XGT, ARIMA and SVR are classical algorithms which emphasize the importance of modeling the relation between input attributes and output attributes. Observing Table 1, it can be seen that the STDLN method obtained significantly better performance than the traditional methods above. For example, compared with the XGT, ARIMA and SVR for the prediction of the 5 min set, the RMSEs of the STDLN method were reduced by 48.69%, 56.69% and 19.26%, respectively; the MAEs were reduced by 66.15%, 71.10% and 21.21%, respectively; the Acc values were higher, at 76.70%, 30.90% and 32.70%; and the R² values were higher, at 82.60%, 57.70% and 35.60%. In fact, the Acc values of less than 70% shows that the performances of the XGT, ARIMA and SVR were rather unsatisfactory. The main reason for this result is that these traditional methods lack the ability to process complex

high-dimensional time series. In addition, the *Acc* of the XGT for the 15 min set was only 0.194, and the result shows that it is hard to find a regression mode from over 100 dimension data points through assembling multiple simple prediction models.

Table 2. The prediction performances of the STDLN model and baseline methods on the datasets.

Dataset	Metrics	XGT	ARIMA	SVR	GRU	GCN	STDLN *	STDLN_F
5 min	RMSE	12.496	14.804	7.942	11.7691	7.877	6.412	17.793
	MAE	10.199	11.944	4.381	8.282	12.232	3.452	18.571
	Acc ratio (-)	0.694	0.652	0.634	0.910	0.610	0.961	0.173
	R ² ratio (-)	0.172	0.421	0.642	0.821	0.823	0.998	0.392
15 min	RMSE	56.475	76.31	46.147	35.67	33.651	31.840	134.345
	MAE	80.761	69.46	60.792	56.47	27.290	16.522	146.694
	Acc ratio (-)	0.478	0.314	0.516	0.671	0.742	0.945	0.172
	R ² ratio (-)	0.247	0.430	0.317	0.426	0.783	0.996	0.137
30 min	RMSE	68.537	51.483	28.383	51.773	447.318	57.734	106.108
	MAE	56.428	37.522	21.306	16.953	132.971	29.244	295.101
	Acc ratio (-)	0.412	0.391	0.637	0.955	0.611	0.957	0.155
	R ² ratio (-)	0.225	0.407	0.154	0.997	0.831	0.999	0.112
60 min	RMSE	73.512	84.376	66.146	99.561	894.196	132.485	1173.267
	MAE	84.374	77.153	44.046	31.646	256.101	73.247	517.062
	Acc ratio (-)	0.284	0.377	0.476	0.957	0.611	0.963	0.156
	R ² ratio (-)	0.117	0.305	0.331	0.997	0.831	0.998	0.105

* The STDLN method obtained the best prediction performance under all evaluation metrics and all datasets.

3.5.2. Performance Comparison between the STDLN method and the Deep Learning Methods

The GRU is a novel deep learning algorithm which emphasizes the importance of modeling temporal features. The GCN is a novel deep learning algorithm which emphasizes the importance of modeling spatial features. The STDLN is a combination of the two algorithms above with material flow topology. The STDLN_F is a special form of STDLN with full connection topology in which material flow is ignored. To verify whether the STDLN method has the ability to capture spatial and temporal features from the energy consumption time series, the performance comparison between the deep learning networks for the 5 min set and the 15 min set are shown in Figure 7.

Observing Figure 7, it can be seen that the STDLN method obtained better performance than the GRU, the GCN and the STDLN_F methods, and the result shows that the STDLN method can improve prediction performance through capturing both spatial features and temporal features from both time series and production nodes. The GRU performed satisfactorily for the 5 min set and obtained an *Acc* of 0.910, but that of the STDLN method was still better by 0.05. The GCN performed satisfactorily for the 15 min set and obtained a lower *RSME*, but that of the STDLN was still better by 5.3%.

As shown in Table 1, the prediction performance of the GRU was generally better than that of the GCN, and this shows that the impact of temporal features is more significant than that of spatial features for the energy consumption of multiple production nodes. However, the stable *Acc* of the GCN under all datasets shows that the impact of spatial features should not be ignored.

All the prediction performances of the STDLN_F were the worst among the methods, and the results show the importance of the topology model of production nodes. It is usually thought that there exists a relationship between any two production nodes since they work together in a manufacturing system, and thus that the CNN may be effective in capturing spatial features without prior knowledge of material flow. However, the significant difference between the STDLN and the STDLN_F shows that the STDLN method only works well when inputting the appropriate topology model. In this paper, the two-level abstracting method was presented to construct the topology model, and the

improvement in the topology model may be an interesting study to improve the prediction performance of the STDLN in the future.

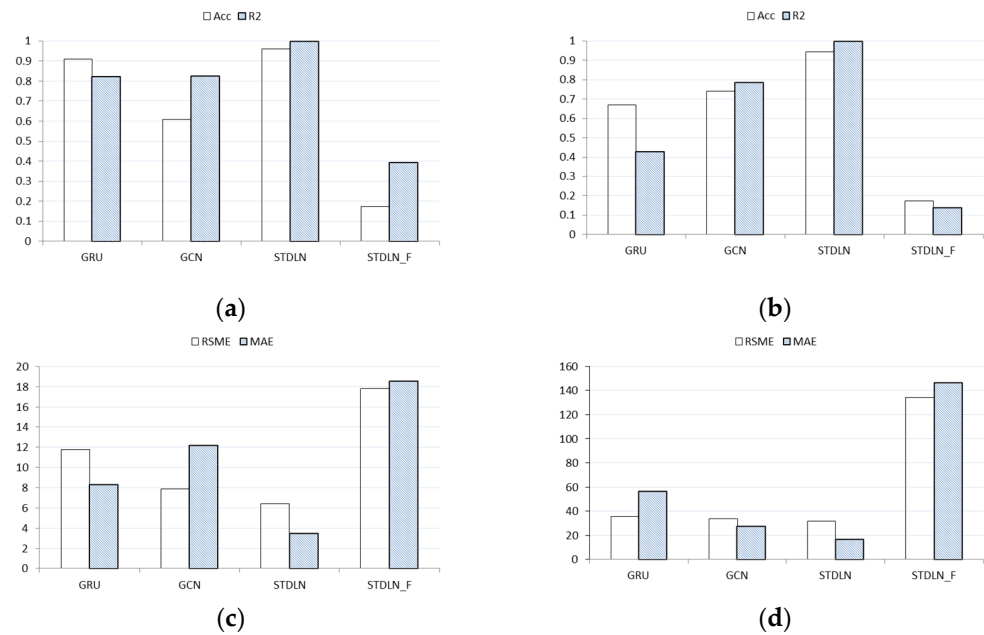


Figure 7. Performance comparison between deep learning networks. (a) The histogram of *Acc* and *R*² for the 5 min set; (b) The histogram of *Acc* and *R*² for the 15 min set; (c) The histogram of *RSME* and *MAE* for the 5 min set; (d) The histogram of *RSME* and *MAE* for the 15 min set.

3.5.3. Performance Comparison between the Datasets

The *RSME* and *MAE* metrics are not comparable between periodical datasets, and hence the normalized *Acc* was used. The scatter diagram of the *Acc* for all methods and all datasets is shown in Figure 8. The *Acc*s of the STDLN for all datasets were the highest, and they fluctuated in a small range (0.94, 0.97). The results show that the STDLN method is insensitive to prediction periods. Thus, the STDLN can be applied for minute-by-minute and hourly energy consumption prediction. Although the GRU obtained satisfactory *Acc* for the 5 min, 30 min and 60 min sets, it fluctuated downward significantly for the 15 min set. The other methods show instability and an unsatisfactory performance with the datasets.

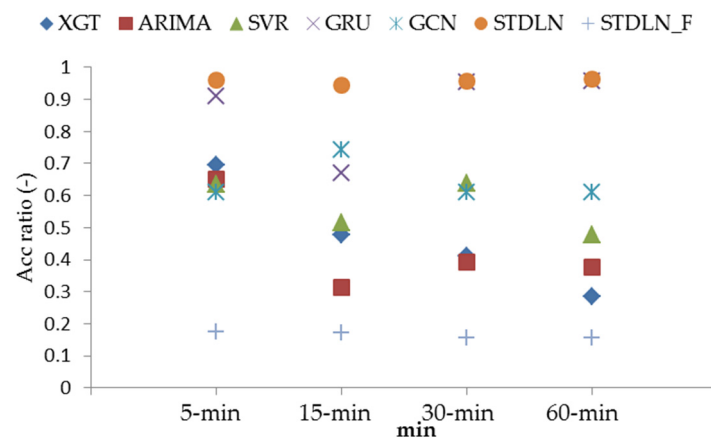


Figure 8. The scatter diagram of the *Acc* for all methods and all datasets.

3.5.4. Performance Comparison between Production Nodes

Four production nodes XW, XW3, EX and EX2, which can be seen in Figure 6, were chosen to evaluate the effect of the STDLN method on production nodes at different

levels. XW is an abstract workshop node, EX is an abstract machine node, XW3 is a visible workshop node and EX2 is a visible machine node. There is distributing flow from the top node XW to the bottom node EX2. The chart of *Acc* for the above four production nodes for the 5 min set and the 15 min set is shown in Figure 9. The prediction accuracy was found to decrease with the decline of the node level. In real manufacturing systems, the energy consumption of the lower node tends to have a more complex temporal law, and hence the corresponding prediction led to a worse performance. Even so, the *Acc* of the EX2 is still acceptable for application in energy wastage identification and energy-saving plans.

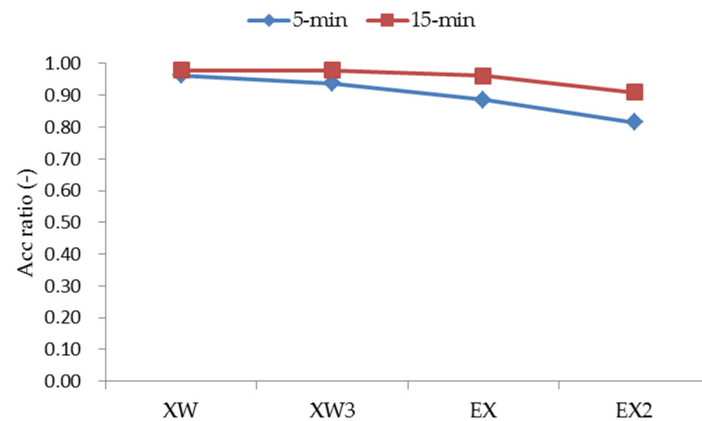


Figure 9. Chart of *Acc* for production nodes at different levels.

4. Discussion

4.1. Discussion of the Effect of GCN and Topology Model of Production Nodes

To better understand the effect of the GCN and the topology model of production nodes, the continuous prediction of all methods for the 15 min set of Node XW3 is shown in Figure 10. In the figure, the true curve and the STDLN prediction curve are highlighted by a heavy red line and a heavy green line, respectively. The STDLN prediction curve was found to fluctuate more gently than the true curve and the other prediction curves. There are two possible causes for this finding: (1) that the GCN model captures spatial features through the smooth filter in the Fourier domain; (2) that the topology model representing material flow between the production nodes leads the prediction curve to represent not the trend of individual change but rather that of overall change.

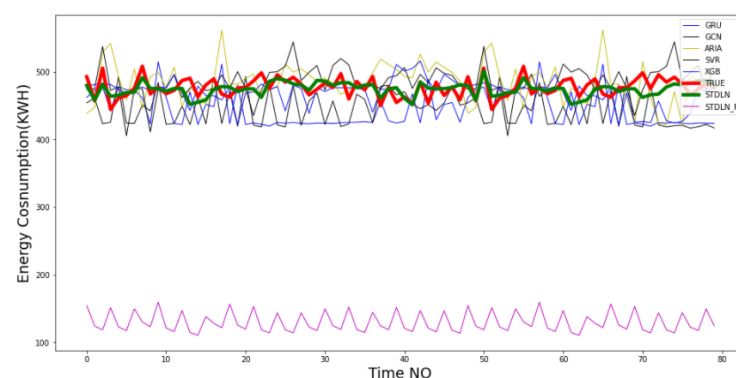


Figure 10. The continuous prediction of all methods for the 15 min set of Node XW3.

On the other hand, there was an obvious gap between the prediction curve of the STDLN_F and the true curve. The result proves the importance of the topology model of production nodes. The application case studied in this paper can be regarded as a typical manufacturing system consisting of parallel machines and flow lines. The parallel machines seem to be closely related because they work together in the same shop, but in

fact they are independent of each other. In the STDLN_F, the unreasonable connections between the parallel nodes introduce serious noise interference and lead to worse prediction performance. By contrast, in the STDLN, the topology model of production nodes only includes dispatching flow and line flow; it filters the noise from unrelated nodes through graph convolution and leads to high prediction performance.

4.2. Discussion of the Effect of Prediction Period and Production Node Level

To better understand the effect of prediction period and production node level, the continuous prediction of the STDLN for all datasets of Node XW3 is shown in Figure 11, and the continuous prediction of the STDLN for all datasets of Node EX2 is shown in Figure 12.

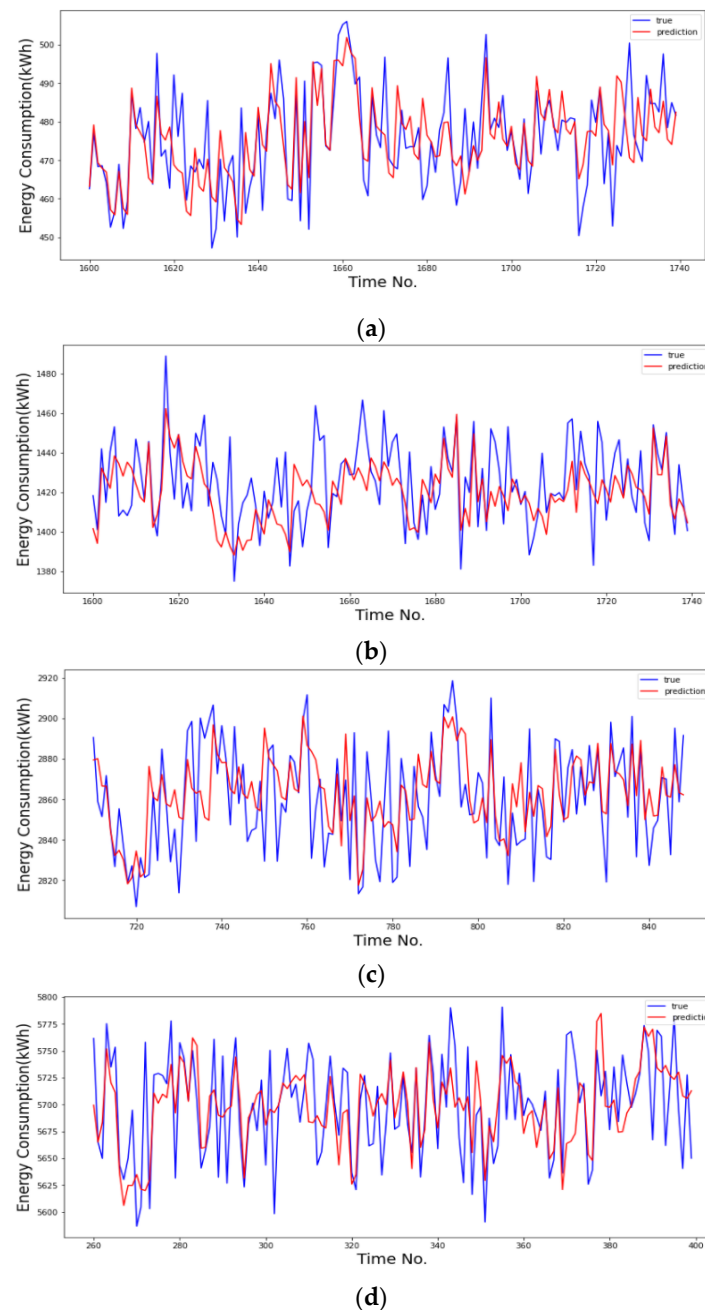


Figure 11. The continuous prediction of STDLN for all datasets of Node XW3. (a) 5 min set; (b) 15 min set; (c) 30 min set; (d) 60 min set.

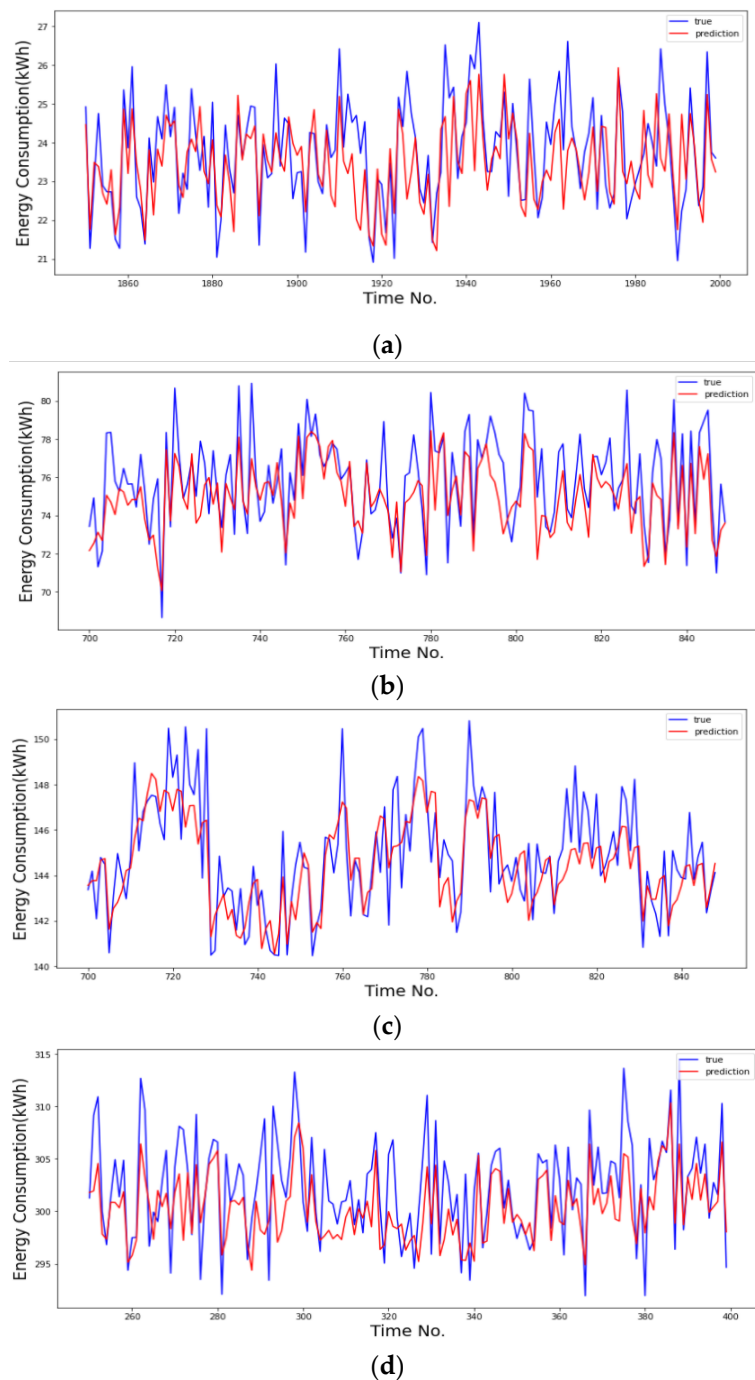


Figure 12. The continuous prediction of the STDLN for all datasets of Node EX2. (a) 5 min set; (b) 15 min set; (c) 30 min set; (d) 60 min set.

As mentioned in the application case, XW3 is a node at the workshop level and EX2 is a node at the machine level. Observing Figures 11 and 12, the STDLN method was always found to achieve satisfactory prediction results for the four prediction periods and two production nodes. As mentioned before, the energy consumption time series in different periods or at different node levels represent different temporal laws, but the STDLN can simultaneously capture these laws through spatio-temporal collaborative learning.

A certain error was also found between the true energy consumption and the prediction energy consumption. The error was mainly caused by the randomness of the machine work state and meter measurement error. Moreover, the error may have been caused by anomalies such as long idleness, machine fault and over-processing. This type of error

is simply the basis of energy consumption anomaly diagnosis. Generally, prediction is the main method for building a baseline for energy consumption anomaly diagnosis. The authors of [1,12] studied energy consumption anomaly diagnosis using the parametric prediction method. Sequential prediction methods for a single node are also frequently seen in the literature related to anomaly diagnosis. Although parametric and sequential prediction methods for a single node can achieve high prediction precisions for time series, the STDLN method is expected to have two advantages: (1) efficiency, which means simultaneously predicting the energy consumption of multiple nodes, and (2) effectiveness, which means that the anomalies can be detected by the state of adjacent nodes. Hence, the application of the STDLN method in energy anomaly diagnosis in manufacturing systems is an interesting study topic for the future.

5. Conclusions

This paper presented a spatio-temporal deep learning network (STDLN) model for short-term energy consumption prediction of multiple nodes in a manufacturing system. The model combines a graph convolutional network (GCN) and a gated recurrent unit (GRU) and predicts the short-term energy consumption of multiple nodes based on a topology model of material flow and the historical energy consumption time series. The GCN is aimed at capturing spatial relationships with the adjacency relation, and the GRU is aimed at capturing dynamic change laws with the time series. Several experiments were performed on the power dataset from an aluminum profile plant. The results show that the model presented can predict the energy consumption of multiple nodes simultaneously and achieve a higher accuracy than models based on the GRU, GCN, SVR, etc.

The analyses of the results show that the STDLN can improve prediction performance through capturing both spatial features and temporal features from the energy consumption series of multiple nodes, obtain a stable performance on minute-by-minute and hourly datasets and obtain an acceptable performance for production nodes at different levels. The STDLN method is expected to support energy wastage identification, energy-saving plans and energy-saving control.

To build on this research, we suggest the construction of a topology model supporting more than one energy consumption type in order to improve the STDLN model through adding an attention mechanism and apply the STDLN in the anomaly diagnosis of energy consumption.

Author Contributions: Conceptualization, J.G.; methodology, J.G.; software, M.H., G.Z. and S.L.; validation, J.G. and M.H.; formal analysis, J.G.; investigation, J.G.; resources, J.G. and M.H.; writing—original draft preparation, J.G.; writing—review and editing, J.G. and G.Z., M.H.; visualization, M.H.; supervision, J.G.; project administration, J.G.; funding acquisition, J.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Natural Science Foundation of Guangdong Province (CN) (Grant No.: 2018A0303130187) and the National Natural Science Foundation of China (Grant No.: 62072123).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available upon request from the corresponding author.

Acknowledgments: The authors would like to thank the data providers and the open-source code providers.

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

References

1. Guo, J.; Yang, H. A fault detection method for heat loss in a tyre vulcanization workshop using a dynamic energy consumption model and predictive baselines. *Appl. Therm. Eng.* **2015**, *90*, 711–721. [[CrossRef](#)]
2. Li, Y.; Hea, Y.; Wang, Y.; Yan, P.; Liu, X. A framework for characterising energy consumption of machining manufacturing systems. *Int. J. Prod. Res.* **2014**, *52*, 314–325. [[CrossRef](#)]
3. Li, H.; Yang, H.; Yang, B.; Zhu, C.; Yina, S. Modelling and simulation of energy consumption of ceramic production chains with mixed flows using hybrid Petri nets. *Int. J. Prod. Res.* **2017**, *10*, 3007–3024. [[CrossRef](#)]
4. Adenuga, O.T.; Mpofu, K.; Ramatsetse, B.I. Exploring energy efficiency prediction method for Industry 4.0: A reconfigurable vibrating screen case study. *Procedia Manuf.* **2020**, *51*, 243–250. [[CrossRef](#)]
5. Rahimifard, S.; Seow, Y.; Childs, T. Minimising Embodied Product Energy to support energy efficient manufacturing. *CIRP Ann.-Manuf. Technol.* **2010**, *59*, 25–28. [[CrossRef](#)]
6. Wang, Q.; Liu, F.; Li, C. An integrated method for assessing the energy efficiency of machining workshop. *J. Clean. Prod.* **2013**, *52*, 122–133. [[CrossRef](#)]
7. Liu, B.; Verbraeck, A. Multi-resolution modeling based on quotient space and DEVS. *Simul. Model. Pract. Theory* **2017**, *70*, 36–51. [[CrossRef](#)]
8. Hu, L.; Liu, Z.; Hua, W.; Wang, Y.; Tan, J.; Wu, F. Petri-net-based dynamic scheduling of flexible manufacturing system via deep reinforcement learning with graph convolutional network. *J. Manuf. Syst.* **2020**, *55*, 1–14. [[CrossRef](#)]
9. Han, Z.; Zhao, J.; Leung, H.; Wang, W. Construction of prediction intervals for gas flow systems in steel industry based on granular computing. *Control Eng. Pract.* **2018**, *78*, 79–88. [[CrossRef](#)]
10. Brillinger, M.; Wuwer, M.; Hadi, M.A.; Haas, F. Energy prediction for CNC machining with machine learning. *CIRP J. Manuf. Sci. Technol.* **2021**, *35*, 715–723. [[CrossRef](#)]
11. Al-Hajj, R.; Assi, A.; Fouad, M.; Mabrouk, E. A Hybrid LSTM-Based Genetic Programming Approach for Short-Term Prediction of Global Solar Radiation Using Weather Data. *Processes* **2021**, *9*, 1187. [[CrossRef](#)]
12. Guo, J.; Yang, H. An anti-jamming artificial immune approach for energy leakage diagnosis in parallel-machine job shops. *Comput. Ind.* **2018**, *101*, 13–24. [[CrossRef](#)]
13. Kuranga, C.; Pillay, N. A comparative study of nonlinear regression and autoregressive techniques in hybrid with particle swarm optimization for time-series forecasting. *Expert Syst. Appl.* **2022**, *190*, 116163. [[CrossRef](#)]
14. Dong, W.; Huang, Y.; Lehane, B.; Ma, G. XGBoost algorithm-based prediction of concrete electrical resistivity for structural health monitoring. *Autom. Constr.* **2020**, *114*, 103155. [[CrossRef](#)]
15. Nguyen, Q.-C.; Vu, V.-H.; Thomas, M. A Kalman filter based ARX time series modeling for force identification on flexible manipulators. *Mech. Syst. Signal Process.* **2022**, *169*, 108743. [[CrossRef](#)]
16. Zhao, Y.; Wang, S.; Xiao, F. A statistical fault detection and diagnosis method for centrifugal chillers based on exponentially-weighted moving average control charts and support vector regression. *Appl. Therm. Eng.* **2013**, *51*, 560–572. [[CrossRef](#)]
17. Wang, L.; Liu, Z.; Chen, C.L.P.; Zhang, Y.; Lee, S. Support Vector Machine based optimal control for minimizing energy consumption of biped walking motions. *Int. J. Precis. Eng. Manuf.* **2012**, *13*, 1975–1981. [[CrossRef](#)]
18. Zhang, X.-X.; Yuan, H.-Y.; Li, H.-X.; Ma, S.-W. A Spatial Multivariable SVR Method for Spatiotemporal Fuzzy Modeling with Applications to Rapid Thermal Processing. *Eur. J. Control* **2020**, *54*, 119–128. [[CrossRef](#)]
19. Jo, T. VTG schemes for using back propagation for multivariate time series prediction. *Appl. Soft Comput.* **2013**, *13*, 2692–2702. [[CrossRef](#)]
20. Elman, J.L. Finding Structure in Time. *Cogn. Sci.* **1990**, *14*, 179–211. [[CrossRef](#)]
21. Hochreiter, S.; Schmidhuber, J. Long-Short Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)] [[PubMed](#)]
22. Cho, K.; Merriënboer, B.v.; Gulcehre, C.; Bahdanau, D. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. *arXiv* **2014**, arXiv:1406.1078. [[CrossRef](#)]
23. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv* **2014**, arXiv:1412.3555.
24. Wang, J.; Yan, J.; Li, C.; Gao, R.X.; Zhao, R. Deep heterogeneous GRU model for predictive analytics in smart manufacturing: Application to tool wear prediction. *Comput. Ind.* **2019**, *111*, 1–14. [[CrossRef](#)]
25. Wang, F.; Xuan, Z.; Zhen, Z.; Li, K.; Wang, T.; Shi, M. A day-ahead PV power forecasting method based on LSTM-RNN model and time correlation modification under partial daily pattern prediction framework. *Energy Convers. Manag.* **2020**, *212*, 112766. [[CrossRef](#)]
26. He, Y.; Tsang, K.F. Universities power energy management: A novel hybrid model based on iCEEMDAN and Bayesian optimized LSTM. *Energy Rep.* **2021**, *7*, 6473–6488. [[CrossRef](#)]
27. Heidari, A.; Khovalyg, D. Short-term energy use prediction of solar-assisted water heating system: Application case of combined attention-based LSTM and time-series Decomposition. *Sol. Energy* **2020**, *207*, 626–639. [[CrossRef](#)]
28. Laib, O.; Khadir, M.T.; Mihaylova, L. Toward efficient energy systems based on natural gas consumption prediction with LSTM Recurrent Neural Networks. *Energy* **2019**, *177*, 530–542. [[CrossRef](#)]
29. Jang, J.; Han, J.; Leigh, S.-B. Prediction of heating energy consumption with operation pattern variables for non-residential buildings using LSTM networks. *Energy Build.* **2022**, *255*, 111647. [[CrossRef](#)]

30. Liu, Y.; Gong, C.; Yang, L.; Chen, Y. DSTP-RNN: A dual-stage two-phase attention-based recurrent neural network for long-term and multivariate time series prediction. *Expert Syst. Appl.* **2020**, *143*, 113082. [[CrossRef](#)]
31. Xuan, W.; Shouxiang, W.; Qianyu, Z.; Shaomin, W.; Liwei, F. A multi-energy load prediction model based on deep multi-task learning and ensemble approach for regional integrated energy systems. *Electr. Power Energy Syst.* **2021**, *126*, 106583. [[CrossRef](#)]
32. Khan, N.; Haq, I.U.; Khan, S.U.; Rho, S.; Lee, M.Y.; Baik, S.W. DB-Net: A novel dilated CNN based multi-step forecasting model for power consumption in integrated local energy systems. *Electr. Power Energy Syst.* **2021**, *133*, 107023. [[CrossRef](#)]
33. Kim, T.-Y.; Cho, S.-B. Predicting residential energy consumption using CNN-LSTM neural networks. *Energy* **2019**, *182*, 72–81. [[CrossRef](#)]
34. Sajjad, M.; Khan, Z.A.; Ullah, A.; Hussain, T.; Ullah, W.; Lee, M.; Baik, S.W. A Novel CNN-GRU-Based Hybrid Approach for Short-Term Residential Load Forecasting. *IEEE Access* **2020**, *8*, 143759–143768. [[CrossRef](#)]
35. Lu, C.; Li, S.; Lu, Z. Building Energy Prediction Using Artificial Neural Networks: A Literature Survey. *Energy Build.* **2021**, 111718. [[CrossRef](#)]
36. Kipf, T.N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. In Proceedings of the International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
37. Guo, S.; Lin, Y.; Feng, N.; Song, C.; Wan, H. Attention Based Spatial-Temporal Graph Convolutional Networks for Traffic Flow Forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; pp. 922–929.
38. Zhao, L.; Song, Y.; Zhang, C.; Liu, Y.; Wang, P.; Lin, T.; Deng, M.; Li, H. T-GCN: A Temporal Graph Convolutional Network for Traffic Prediction. *IEEE Trans. Intell. Transp. Syst.* **2019**, *21*, 3848–3858. [[CrossRef](#)]