Abstract: Predictive maintenance, production scheduling, and machine health monitoring (MHM) are just a few of the areas of smart factories where timeseries forecasting is used. Machine speed prediction can optimize production throughput, lower energy consumption, and dynamically modify production processes depending on different system variables in smart factories. Making precise, data-driven predictions about machine speeds is difficult, though. Due to the complexity of the data generated by industrial production processes, predictive models that are robust to noise and can reflect the temporal and spatial distributions of timeseries signals are required for successful forecasting. Inspired by current deep learning efforts in smart manufacturing, this paper proposes an end-to-end framework for multi-step machine speed prediction. The model, also known as the 2D-Convolutional LSTM Autoencoder, is composed of the deep convolutional LSTM (ConvLSTM) encoder-decoder framework. When contrasted to cutting-edge predictive models, the utility of the suggested method is demonstrated by extensive empirical evaluations utilising real-world data from a metal packing plant in the United Kingdom.

Keywords: Smart Manufacturing, LSTM, 2DConvLSTMAE, CNN

1.1 Smart manufacturing systems for Industry 4.0

The “Industry 4.0” concept is being adopted by the industrial sector in a number of sectors, including product design and logistics. Mechatronics, a foundational concept in the designing of production environments [3], has been revised to fit CPS. It has been proposed to develop smart products with specific specifications for customized goods. Predictive maintenance and its application to machine health forecasts are hot topics in Industry 4.0-based CPS. The most recent generation of machine tools, Machine Tools 4.0, is being used at the machining sites. Energy monitoring systems are now independent systems with self-optimized energy usage thanks to Energy Management 4.0, which has also been proposed for decision-oriented energy data.
Research has also been done on the potential impact of Industry 4.0 technology on logistical systems. All potential applications cannot be covered inside one document. Therefore, the applications for designing, monitoring, manufacturing, controlling, and scheduling are the only ones covered in this paper. Figure 1 depicts the framework for Industry 4.0 smart production environments.

![Diagram of Industry 4.0 smart manufacturing models' architectural design](image)

**Fig. 1 Industry 4.0 smart manufacturing models' architectural design**

Common Industry 4.0 issues are shown on the horizontal axis. These issues are the core subjects of this work. The vertical axis shows issues with several aspects of Industry 4.0, such as the use of sensors and actuators in addition to information gathering, analysis, and decision-making. Data collection and analysis are the primary causes of intelligence for the activities displayed on the horizontal axis of Industry 4.0.

i. **Smart design:** Traditional design has been enhanced and made smart (AR) as a result of the speedy advancement of innovative technologies including augmented reality (AR) and virtual reality (VR) [5]. Virtual reality techniques to hybrid prototyping have been used in addition manufacturing. Design tools like computer-aided manufacture (CAM) and computer-aided design (CAD) may now connect in real time with sophisticated physical prototype systems because to the combination of 3D printing with CPS and AR. Thus, by fusing technological breakthroughs with physical realizations, a smart design concept may be attained.

ii. **Smart machining:** Smart machining is possible in Industry 4.0 with the aid of sentient robots and other smart objects that could also observe and interact among themselves in genuine. In order to provide smart manufacturing solutions, real-time data can be gathered by CPS-enabled smart mechanical equipment and delivered to a central cloud-based system. There, it can be synchronized with the linked services. Self-optimization control systems also enable in-process quality control, eliminating the need for post-process quality check [6].

iii. **Smart monitoring:** Monitoring is essential for the proper operation, maintenance, and planning of Industry 4.0 manufacturing systems. The ubiquitous usage of various sensors has made smart monitoring possible. For example, real-time data can be collected on the temperature, electricity use, vibrations, and speed of a range of manufacturing items. Smart monitoring displays this data graphically and alerts users when a machinery or piece of equipment behaves abnormally. Industry 4.0 smart production solutions use CPS and IoT as advanced components for smart monitoring.

iv. **Smart control:** Industry 4.0 can accomplish high-resolution, adaptable production control by developing cyber-physical production planning and control systems. Smart control is primarily employed to physically control several smart instruments or devices through a system that is cloud-enabled. Customers can turn off equipment or robots using their smartphones [7]. Then, decisions can be swiftly put into practice at industrial facilities employing cutting-edge machinery, such as automated assembly lines or machine intelligence.

v. **Smart scheduling:** Smart scheduling mostly uses sophisticated models and algorithms to draw info from sensor data. The use of sophisticated decision structure and data-driven approaches can enable smart scheduling. Distributed smart systems with a multilayer interactive design are a viable option for reliable genuine scheduling and executions [8]. The implementation of production behaviors and processes is then automatic and effective thanks to the existing frameworks and services. With the use of data input devices, the output resolutions are returned in a variety of ways to the parties involved.

vi. **Industrial applications:** The ultimate goal of Industry 4.0 is to focus on various industry deployments of innovations that have the potential to change industrial processes. Business 4.0's options are easily versatile to enable customized design and creation in compliance with the uniqueness and specific requirements of a range of industries, for example the food business, which deals with a sizable amount of perishable items. As a result, adaptive manufacturing networks have the ability to regulate...
their operating and supply modes. Applications that use adjustable abilities from layers of intelligent design, engineering, and decision-making can achieve a broad view by factoring in practical issues, including such process performance, logistics availability, time constraints, and countless criteria.

The following is a summary of a few of the major research areas covered by this paradigm.

i. Intelligent manufacturing and design: This degree of study includes smart design, smart prototyping, smart controllers, and smart sensors [9]. Genuine control and monitoring enable the adoption of smart manufacturing. Instances of enabling technology include the Internet of Things, STEPNC, 3D printing, robotic arms, and wireless networking.

ii. Smart decision-making: Industry 4.0 is built around making informed decisions. The widespread use of sensors is intended to facilitate informed decision-making through comprehensive data collection. To achieve intelligent decision-making, there must be a real-time interchange of information and collaboration. Big data and related analytics are a key component of smart decision-making processes including data-driven modelling and data-enabled predictive maintenance. CPS, big data analytics, cloud computing, modelling, and simulation are just a few of the technologies that enable smart decision-making.

iii. Big data analytics: Due to the massive amounts of data that Industry 4.0's CPS and IoT-assisted manufacturing create, big data analytics is crucial for the design and operations of production systems [10]. The big data analytics technique has been used to provide a thorough methodology for data-driven risk analysis using genuine information for industrial manufacturing systems. A issue like this promotes production CPS visualization and quality production, according to popular consensus.

iv. Industrial implementations: Industrial applications are Industry 4.0's ultimate goal. Almost every industry, particularly manufacture, farming, information and entertainment, services, transportation, and aviation, can benefit from the upcoming industrial revolution. Numerous new opportunities will be available for industrial parties. Companies may focus on their fundamental business tenets or challenges that Industry 4.0-enabled innovations could enhance or resolve [11].

1.2 Existing challenges

The majority of manufacturing systems carry out multiple processes in accordance with predetermined production logics using standard machinery. To support these operations, manual and paper-based working techniques are frequently used. The utilization of these mechanisms faces a number of difficulties, which are outlined below:

i. When a large number of employees are employed, shop floor activities, interactions, and executions take a long time, which initially results in low operating efficiency. For instance, while reengineering designs, it is often necessary to hold a conference with machine operators, professional engineers, chief engineering technicians, and shop floor managers to discuss and come up with a solution. Because information or data must be presented and current conditions must be examined to provide an appropriate solution, such sessions typically go longer than half a day.

ii. Secondly, paper sheets or record cards are typically used for data collecting [12]. Critical information, including working components, quality information, and WIP level, must be recorded by various personnel. Workers are frequently preoccupied with operating machines and resent devoting time to data recording, a procedure that adds no value.

iii. Third, shop floor managers must use data to make decisions regarding manufacturing, including scheduling and production planning. Given the time and effort required to deal with numerous paper sheets and cards, plus the fact that the information acquired is virtually always outdated, decisions based on data from so many paper sheets or cards are more likely to become unreasonable and unworkable. For the majority of industrial organizations, real-time data collection is necessary to keep up with the Industry 4.0 age. These problems may have solutions thanks to IoT and CPS.

Manufacturing businesses are currently having trouble exhibiting and perceiving a variety of manufacturing services. In Industry 4.0, information transparency is crucial for making accurate decisions. Implementing factory virtualization and visualisation presents a number of difficulties. First, to guarantee production quality and safety, manufacturing objects should be visualised in real time. The sole available option, closed-circuit television (CCTV), however, is unable to display a functional machine's status [13]. In order to be shared as a service, manufacturing resources should be virtualized into distinct services. Sharing models and virtualization techniques have seldom ever been studied and reported. Finally, a new data modelling strategy that can merge disparate data into a common format is necessary to improve the visibility of distinct industrial items. Such information can then be shown for various end users who are worried about the visibility of various pieces of equipment.

2. Literature Review

J. C. Serrano-Ruiz, et.al (2022) suggested a conceptual framework called SMS (smart manufacturing scheduling) such as a SOC, a hierarchical agent model, and DRL technique [14]. This model emphasized on creating a conceptual and structured association framework in enhancing the efficacy of the procedure to schedule the job shop. SMS approach described this model. Hence, this task made the system more flexible and the ML (machine learning) for attaining autonomous operation. This approach led to integrate the efficacy of the particular set of Industry 4.0 principles with the expertise which used to build the suggested framework. The suggested framework assisted the researchers in moving toward the digital alteration of job shops.

K. Nagorny, et.al (2020) presented a data pipelining technique on the basis of a semantic model to define SM assets itself and for accessing their data with their living process [15]. Earlier work employed the reference architectures such as RAMI 4.0 to describe the meta-data or classify the quality. These results were based on these and a semantic algorithm based DIN Spec
91345 compliant data pipelining method was suggested with SM domain as ideal case. The presented technique was useful to explore the data, discover, access and extract the data. Moreover, the data was saved and transferred using the presented technique.

A. Sajjad, et.al (2020) projected a conceptual model of SMS (smart manufacturing system) [16]. A mathematical algorithm was employed to compute the operational flexibility for SM in Open Platform based CPPS at shop floor level. The projected model was useful to enhance the system with regard to flexibility, and higher capacity. This model led to enhance the process to manufacture the product and offered accuracy of 30.4% with SCNCM (Smart Computer Numeric Control Machining), 53.6% with SARM, 55% with SAM and 65% with SHASM on industry 4.0 paradigm.

A. Essien, et.al (2020) introduced an E2E (end-to-end) framework in order to predict the MMS (multistep machine speed) to accomplish SM (smart manufacturing) [17]. A DC-LSTM (deep convolutional long short-term memory)-ED (encoder-decoder) model was exploited for this purpose. The input sequence restricted to a SL way with the help of sliding-window technique. The issue of predicting the multistep time-series was tackled using Two Dimensional-Convolutional Long Short-Term Memory Autoencoder algorithm. This model utilized the potential of CNN (convolutional neural network) to extract the features. The AD model was effective to illustrate the representation learning, which led to mitigate the computing demand and training time. The experimental outcomes indicated that the introduced framework was more effective as compared to others.

X. Zhou, et.al (2022) established a hybrid DNN (deep neural network) in which MobileNetwork version 2, You Only Loo Once version 4, and Openpose, algorithms were adopted for recognizing the real-time status from physical manufacturing environment to visual space [18]. Thereafter, an LA was put forward to detect an effective multi-type little object depending upon the process of integrating the feature and fusion from both shallow and deep layers. Hence, the entire manufacturing procedure was monitored, modeled and optimized in the DT system. The experimental results revealed that the established algorithm was effectual and reliable as well as offered higher accuracy in SM (smart manufacturing).

C.-C. Chen, et.al (2018) developed a new MRSJO method based on MB [19]. Initially, a DJP model put forward which assisted the distributed edge devices in accomplishing an SHC job for supporting the manufacturing jointly. Subsequently, a MRSJO algorithm was utilized to split an SHC job into multiple portions. The selected edge devices were employed to process it in parallel. Such job was accomplished using a model which was useful to choose suitable edge devices with regard to prior manufacturing behaviors of every edge device to attain higher offloading efficacy. In the end, the testing was performed on a factory with 20 machines. The outcomes exhibited that the developed method performed efficiently.

H. Tang, et.al (2018) designed a CASO (cloud-assisted self-organized) model for which smart agents and cloud were included for communicating and negotiating via networks [20]. Ontological representations of knowledge base were developed for offering the information basis to make the decisions of agents so that the dynamic reconfiguration as established among agents in a collaborative way. In addition, this model aimed to model the interaction behavior of agents for structuring the agents hierarchically with the objective of mitigating the complexity. The experimental outcomes reported that designed model was capable of generating a SMS (smart manufacturing system) and making the MS more adaptive and robust against the mixed multi-product tasks.

X. Wu, et.al (2021) emphasized on the issue related to real-time HFSP [21]. The first stage was executed to analyze the attribute of hybrid flow shop in a SM scenario. After that, a real-time scheduling method, to tackle the Hybrid Flow Shop Scheduling, was recommended. The GEP (gene expression programming) model was deployed in the core module for creating a novel and effective scheduling rule. This rule was adopted to compute the priorities of the waiting job and schedule the task having higher priority. The experimental results validated the adaptability of the recommended method to schedule the HFS in a SM (smart manufacturing) environment.

K. I.-K. Wang, et.al (2022) intended a novel technique called FTL (federated transfer learning) in order to deal with the issue regarding the lack of data and security occurred in ML technique in MSM (modern smart manufacturing) [22]. A central server and various sets of smart devices were adopted for handling a diverse application. The novel applications deployed a TL (transfer learning) method for transforming a base system into their target-domain systems. In the meantime, the intended technique adopted for enhancing the accuracy of the application-based system. This technique resulted in protecting the data privacy. The intended technique computed on COCO and PETS2009 datasets. The simulation results depicted that the accuracy of the intended technique was found higher as compared to other methods.

S. Malik, et.al (2020) suggested an effectual TM (task management) model on the basis of ACM-FEF algorithm [23]. In this, ACM was integrated with FEF algorithm for managing the job. The fundamental objective of the initial algorithm was to maximize the manufacturing and focus on the productivity objectives of all the machine networks which the smart factory has contained. The tasks starvation rate was alleviated and the machine usage was increased on the basis of the latter technique. The suggested model was applicable for executing the plan tasks, maximizing the resource usage, productivity, alleviating the production delays, handling the exceptions and efficiently controlling the smart factory actuators.

P. R. R. Paiva, et.al (2021) presented an online diagnoser depending upon Petri framework of a SMS (smart manufacturing system) to make the decision related to the fault event for which the sequence of monitored events was stored [24]. Only simple
inequality verification was required in this model. This model assigned labels to diverse transitions. A hypothetical machine, whose embedding was done in a smart manufacturing line, employed to represent the efficiency of the presented framework. The findings indicated the supremacy of the presented framework the existing ones.

M-K. Kazi, et.al (2020) investigated a technique to accomplish the SM on Industry 4.0 for predicting the load and displacement curve of objective cotton fiber composite stuffs so that the essential aimed properties were fulfilled [25]. Experimental data was employed to deal with the composite fiber percentage which is varied to characteristic load and prior generated ANN techniques were exploited as the feed. In the end, the exact data driven generated model was recognized using a PySINDy. The investigated technique offered superior efficiency.

Research Methodology
This section explains that suggested deep ConvLSTM auto-encoder algorithm for predicting the machine speed in univariate time-series way, is defined in a smart factory.

1. ConvLSTM Encoder
This ConvLSTM is implemented to perform the time-series classification in order to detect the anomaly. For this, video sequences are utilized. Thus, its performance is alleviated with the maximization of sequence length. This issue is resolved using an attention-based system for determining and retaining the relevant hidden states over the time steps. These equations are expressed as:

\[ i^{t_l} = \sigma (W_{xz} x^{t_l} + W_{hh} h^{t_l-1} + b_i) \]
\[ f^{t_l} = \sigma (W_{xf} x^{t_l} + W_{hf} h^{t_l-1} + b_f) \]
\[ c^{t_l} = i^{t_l} \cdot \tan (W_{xc} x^{t_l} + W_{hc} h^{t_l-1} + b_c) + f^{t_l} \cdot c^{t_l-1} \]
\[ o^{t_l} = \sigma (W_{xo} x^{t_l} + W_{ho} h^{t_l-1} + b_o) \]
\[ h_c = o^{t_l} \cdot \tan (c^{t_l}) \]

In this, the Hadamard product is denoted with \( \circ \), the sigmoid function with \( \sigma \). The convolutional kernels illustrated in the model using \( W_{xz}, W_{xf}, W_{xc}, W_{hh}, W_{hf}, W_{hc}, W_{ho}, W_{ho} \in \mathbb{R}^{n \times 8 \times 64} \). The bias metrics are represented in the \( t \)-th layer of the ConvLSTM through \( b_i, b_f, b_c \) and \( b_o \). The figure 2 demonstrates the framework of the suggested algorithm.

![2DConvLSTMAE Model Architecture](image)

A layered structure employs to arrange the layers of ConvLSTM for extracting the temporal attributes in hierarchical way. The hyper-parameter defines the length of sequences which leads to affect the efficacy of the algorithm and its optimization is required. A grid search model is adopted to determine the optimal length of sequences. Moreover, three of these lengths employed in the training regime. A FCN of 10-units is obtained in the output.

B. Bi-directional LSTM Decoder
The result of the encoder stage of the suggested algorithm is considered as a set of feature map vectors of dimension \( (n \times 1 \times 8 \times 64) \) in which \( n \) is employed to illustrate the amount of samples utilized to train the system. The feature maps generated through the encoder layers are decoded using a repeat vector layer. This layer emphasizes on repeating the final output vector from the encoding layer in a shape that is available as a stable input for every time step of the decoder. Hence, the decoding layer has potential for reconstructing the original input sequence. A layered Bi-LSTM (Bidirectional-Long Short Term Memory) algorithm employs the output taken from this repeat vector layer. There are 200 Long Short Term Memory units deployed with ReLU to develop every LSTM layer. The next layer deploys the output taken from the earlier layer for input hierarchically. Thus, the decoder layer is effective of incorporating the encoded output vector from the Convolutional Long...
Short Term Memory Auto-encoder for enhancing the efficacy of the predictive model when the RL is adopted at the single layers [12].

C. Hyper-parameter Optimisation

The efficiency of DL algorithms is relied on the predetermined hyper-parameters that an optimization procedure has offered. Different from the model metrics, that an optimization function learned, for diminishing the LF (loss function), the learning of hyper-parameters is not done when the model is trained. Various hyper-parameters are available in Deep Learning techniques. For this study, the optimization is performed on eight hyper-parameters which is demonstrated in Table I.

<table>
<thead>
<tr>
<th>Layer</th>
<th>Hyper-parameter(s)</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConvLSTM1</td>
<td>Filter</td>
<td>16, 32, 64, 128, 256, 512, 1024</td>
</tr>
<tr>
<td></td>
<td>Kernel Size</td>
<td>(1 x 3), (1 x 4), (1 x 8), (1 x 12)</td>
</tr>
<tr>
<td>ConvLSTM2</td>
<td>Filter</td>
<td>16, 32, 64, 128, 256, 512</td>
</tr>
<tr>
<td></td>
<td>Kernel Size</td>
<td>(1 x 3), (1 x 4), (1 x 8), (1 x 12)</td>
</tr>
<tr>
<td>LSTM 1</td>
<td>Units</td>
<td>100, 200, 300, 400, 500, 600</td>
</tr>
<tr>
<td>LSTM 2</td>
<td>Units</td>
<td>100, 200, 300, 400, 500</td>
</tr>
<tr>
<td>N/A</td>
<td>Dropout</td>
<td>0.1, 0.2, 0.3, 0.4, 0.5</td>
</tr>
<tr>
<td>N/A</td>
<td>Learning Rate</td>
<td>1e-4, 1e-5, 1e-6, 1e-7</td>
</tr>
<tr>
<td>N/A</td>
<td>Batch Size</td>
<td>8, 16, 32, 64, 128, 256, 512</td>
</tr>
<tr>
<td>N/A</td>
<td>Optimiser</td>
<td>Adam, SGD, AdaDelta, RMSProp, Nadela, Nadam, RAadam</td>
</tr>
</tbody>
</table>

Various techniques, of optimizing the hyper-parameter, namely RS (Random Search), GS (Grid Search), BO (Bayesian Optimization), etc. are present. Though, this study makes the deployment of GS model to optimize the hyper-parameters of the suggested algorithm as well as of individual existing model. This model is adopted as it is more reliable in spaces of lower dimensionality as compared to existing model. Thereafter, the implementation of grid search model and the configuration of parallelisation is easy. This study projects a technique to optimize the hyper-parameter. Let set \( \forall \) in which an index of \( n \) possible configuration hyperparameters \( h \) is comprised. This model majorly focuses on selecting a set of values for every hyper-parameter \( (h^1 ... h^k) \) for mitigating the validation loss. Moreover, all the integrations of values are integrated in a grid format to offer the amount of trials in a GS is

\[
S = \prod_{n=1}^{m} h^{(2)}
\]

The given equation illustrates the implementation of the hyper-parameter SS in the GS model for optimizing the model hyper-parameters. Moreover, this study deploys a similar technique to optimize other benchmark algorithms.

D. Optimiser

The DL (Deep Learning) techniques exploits the SGBO (stochastic gradient based optimization) algorithm to optimize the metrics of framework. DL techniques is consisted of various optimization algorithms known as RMSProp, AdaGrad, and vSGD. The suggested work suggests and deploys a SGBO algorithm called Adam. The GS model determines the learning rate value around \( 1 \times 10^{-6} \).

E. Loss Function

This study considers the RMSE as the LF. Thus, the computation and back propagation of these values, is done for updating the metrics of the framework with every epoch. This metric is computed as:

\[
RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2}
\]

In this, the predicted values illustrated with \( \hat{y}_i \), target variable with \( y_i \) and the sample size with \( n \). The optimizer of the earlier subsection is employed to implement the MSGD (mini-batch stochastic gradient descent) for alleviating this loss. A number of epochs consider to train the system. Afterward, the next 10 time steps are predicted with regard to network metrics on a sequence of input preceding observations.

4. Result & Discussion

In this section, the introduced Deep Learning sequence-to-sequence time-series prediction framework in predicting machine speed is described along with the experimental setup and evaluation procedure. Using historical data from a body-maker machine, the 2D-ConvlSTMAE model’s performance is empirically assessed.

A. Data Preparation
The historical machine-collected speed data for this study were obtained at a frequency of 1/60Hz. The information shows the rate at which an aluminium of higher speed leads to operate the machine, as recorded internally by the machine and expressed as the amount of strokes at every minute. The full length body of can is produced by this bodymaker machine in the production of metal cans from a small cup which is pushed through a succession of iron rings. Bodymaker machine throughput and yield are influenced by operating speed, which typically displays an assortment of periodic patterns tied to "regular" manufacture schedules and episodic, erratic patterns brought on by atypical activities. Additionally, procedures like the cupper and the machines to clean or sterilize the can have an impact on (and are affected by) this machine.

The dataset used in this study included 525,600 observations of minute-by-minute machine speed over the time period of 31/08/2017 00:00 to 30/08/2018 23:59. From this dataset, 463,978 observations were utilised to train the model, 51,553 for testing, and 10,000 for validation. Because the data needs to be translated into a (sequential) supervised learning-friendly structure. In this investigation, Fig. 3 employs a sliding window of size $w$ of 60 and a repeated step size of 1.

![Figure 3: Transforming input timeseries to samples](image)

The prediction window of $k=10$ min is selected. As a result, the trajectory of the univariate input timeseries data was used to transform the training data from that input sequence of shape $(N \times 1)$, $x(t) = \{x_1, x_2, x_3, ..., x_N\}$, to a matrix of shape $(n \times 60 \times 1)$, where $N$ represents the total observations in the dataset (i.e., 525,600) and $n$ denotes the size of the training dataset $n=(N-k-w+1)=525,531$.

B. Baseline models

This paper analyses the efficacy against naive, statistical, and 3 cutting-edge DL standard techniques for quantifying two-dimensional Convolutional Long Short Term Memory Auto-encoder algorithm to predict the speed of multi-step machine.

1) Persistence Model

It is a popular timeseries predictive approach operated under the presumption that the forecasted value of the target variable doesn't vary over time. In other words, $\hat{y}_t = y_{t-1}$ for all times is the projected value at time $t$. This naive model demonstrates exceptional accuracy, particularly in short-term forecasting, although it shows weaknesses in prediction encompassing multiple steps.

2) Autoregressive Integrated Moving Average (ARIMA)

It is a popular algorithm to predict time-series. The primary tenet of this algorithm is that the mean and variance are stationary and that the intervals and the future state have a linear connection. This constraint is reflected in the fundamental assumptions of ARIMA.

3) Residual-Squeeze Net

The squeeze operation which combines the data based on the most effective set of channels discovered during model training creates the 1-dimensional CNNs that make up the RSNet utilising the residual-squeeze net architecture. In this study, the model is trained using data from a single input channel.

4) Deep LSTM Encoder-Decoder

Stacked design of Long Short Term Memory layers coupled to a time-distributed dense layer are used in this study's architecture.

5) CNN-LSTM Encoder-Decoder

The third benchmark is a CNN-LSTM autoencoder model. Although a classifier was described in the paper, the model is altered to add a regression layer.

C. Model Performance Evaluation

This work uses a method known as walk-forward confirmation or back testing to quantify the model. Assuming the observations are not related to each other, that is not possible in case of time-series data, at which it is essential to keep the sequential dimension, traditional prediction validation techniques like k-fold cross-validation is incapable of function effectively.
As shown in Figure 4, the various models of smart manufacturing are compared in which the proposed model gives low RMSE and MAE.

**Conclusion**

The prediction of machine speed in a SM procedure has been proposed in this study using a revolutionary deep ConvLSTMAutoencoder architecture. The issue related to multi-step timeseries prediction is addressed in the predictive model 2DConvLSTMAE by reorganising the input sequence in a SL way. For this, a sliding window technique is implemented. The introduced algorithm takes advantage of the CNN's superiority in automatically extracting features and the LSTM's superiority in SRL. An ED design in the suggested model encourages RL when the system is trained, to offer least computing demand and training time. It also serves as a dimensionality reduction strategy. Experimental outcomes of analysis demonstrate that the introduced algorithm performed more efficiently in contrast to traditional techniques and 3 cutting-edge Deep Learning algorithms, at superior predictive performance. Our model not only had better predictive performance but also required a lot less time to train. This makes 2DConvLSTMAE a more useful strategy for implementation in actual manufacturing processes, as well as better at predicting machine speed. The findings of this study can be used to improve production scheduling and planning by directly applying MSTS forecasting to operations in SM processes. For instance, when an indication of future production output is obtained such that the operational needs are changed appropriately, anticipating machine speed can be utilised to support just in time (JIT) manufacturing. The suggested model will be expanded in subsequent iterations of this study to include multivariate timeseries that contain information from external sensors and machine states.

**REFERENCES**

8. C. Monsone and Á. B. Csapó, "Charting the State-of-the-Art in the Application of Convolutional Neural Networks to Quality Control in Industry 4.0 and Smart Manufacturing," 2019 10th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2019, pp. 463-468
25. M-K. Kazi, F. Eljack and E. Mahdi, Data-driven modeling to predict the load vs. displacement curves of targeted composite materials for industry 4.0 and smart manufacturing”, Composite Structures, vol. 23, no 1, pp. 832-836, 28 October 2020