An Advanced LSTM Model for Optimal Scheduling in Smart Logistic Environment: E-Commerce Case

YASSINE ISSAOUI1, AZEDDINE KHIAT1, AYOUB BAHNASSE2, AND HASSAN OUAJJI1

1SSDA Laboratory, Hassan II University of Casablanca, Casablanca 20000, Morocco
2Ensam Casablanca, Hassan II University of Casablanca, Casablanca 20000, Morocco

Corresponding author: Yassine Issaoui (issaouiyassine818@gmail.com)

ABSTRACT At present, most logistics systems, especially those dedicated to e-commerce, are based on artificial intelligence techniques to offer better services and increase outcomes. However, the variety and complexity of resource allocation, as well as task scheduling, denote that dynamic environments have still great challenges to overcome. So advanced models based on strong algorithms are required. Introducing advanced models into scheduling solutions is a promising way to enhance logistics efficiency. As a result, managing system resources remain essential to optimize task scheduling respecting the interactive impacts, and logistics systems requirements. In response to these challenges, in this paper, a powerful solution based on a Long short-term memory (LSTM) model is proposed to optimize resource allocation and to enhance task scheduling in a smart logistics framework. This paper explores some of the most important scheduling techniques and hypothesizes that deep learning techniques might be able to afford accurate approaches. The proposed smart logistics model lays on strong techniques, for that, experimental simulations were conducted using various project instances. The validation tests demonstrated competitive results with important performance rates i.e.: accuracy of 92.44% with a precision of 93.83, a recall of 95.18%, F1-score of 94.92%, and an AUC of 88.17%. These results reveal the proof-of-principle for using LSTM models for effective and truthful logistics operations.

INDEX TERMS Artificial intelligence, deep learning, LSTM, optimization, smart logistics, task management, task scheduling.

I. INTRODUCTION
Global companies are often subject to a huge number of constraints that affect their processes. This environment is characterized by growing competition, more personalized demand, enlarged operating costs, continuous climate change, and endless evolution. Besides, logistics systems have become a major focus for smoothing flow management, advancing transactions, and improving the added value along the supply chain in the global economy. On this matter, smart logistics systems ensure the maximization of productivity, which has been the backbone of the previous industrial revolutions, the automation of processes, and the well-organized use of resources for greater efficiency.

Logistics environments encounter increasing requirements, operations, and issues. However, it is quite difficult to cover all these challenges. The goal is to manage accurately logistics activities to ensure useful decisions by moving from meeting simple requirements to the adequate prediction of future activities. Initially, resource allocation and task scheduling form the important processes for logistics systems. Although much research interest has been intensely increased, still complex problems have not been sufficiently studied. In conventional methods, real-time performance remains essential to warrant strong operative processes within a dynamic logistics environment, such as e-commerce. These systems are based on a set, of human resources, hardware, software, techniques..., to manage the basic logistics infrastructures. One of the main objectives of smart logistics (SL) is to use these resources efficiently and increased profit. Accordingly, scheduling is the main issue affecting task processing. For that, scheduling algorithms must balance the system performance with service quality while maintaining efficiency and fairness between functionalities [1].

Being an active logistics environment, e-commerce is speedily evolving, and therefore affecting global economic
development. E-commerce represents a sales transaction through an electronic payment transaction. adopting intelligent technologies can ease these operations among customers and companies through information flows, trade flows, capital flows, and logistics [2]. As a result, many companies have adopted e-commerce activities to outperform their competitors and improve their market position. The emergence of e-commerce then brought new channels for worldwide expansion [3]. E-commerce allows more fluid economic exchanges. However, its efficiency always depends on logistic flows. So far, the evolution of logistics industries is significantly consolidating e-commerce activities [4].

A key factor in logistics systems is between the distribution of tasks and the minimization of the execution time. This refers to the optimization problem to be able to manage dynamic systems with a high frequency of operations. Also, the growing adoption of smart logistics has led to an increased reliance on artificial intelligence technologies for resource and cost-efficient decision-making. These technologies must be used to effectively manage system tasks and resources. Therefore, a scheduling approach to setting up and sequencing the completion of different activities is necessary to ensure the optimal use of resources and improve execution performance.

In logistics systems, each task, belonging to an activity, receives the required resources. In contrast, the same resource can be used, at the same time, for different tasks or activities, which can damage task processing. This raises two important problems:

- Allocating resources is extremely time-consuming, which can affect quality control;
- Scheduling a task is usually extreme. Some tasks may be used more frequently than others. This can lead to undesirable biases in the learned models.

This is particularly relevant when we want to schedule logistics tasks to manage a new project or mission. The mentioned problems can be solved by selecting, effectively and efficiently the resources. Task scheduling is the process of distinguishing between the different resources that form a task, which can be used to set up the needed planning and detect system key points. For example, task scheduling in logistics systems may not work well because the environment is eventful, and then flexible scheduling is required. Instead, it is vital to adopt a technique that can predict the exact schedule of any logistics project so that the logistics system can safely process all the required tasks. Indeed, we develop a novel scheduling method based on the LSTM model to enhance logistics systems.

The objective of this work is to cover a large part of the logistic service in real-time projects based on deep learning algorithms. The LSTM network offers memory blocks instead of RNN units to solve the problem of the disappearance and explosion of the gradient [5]. Indeed, LSTM [6] addresses this gradient problem by replacing self-connected hidden units with memory blocks. Each memory block uses a specially designed memory cell to store information, and it is more efficient at finding and exploiting a long-term context. Memory units allow the network to know when to learn new information and when to forget old information.

As e-commerce is one of the most dynamic logistics systems, we attempt, in this paper, to study the scheduling problem to enhance the delivery service. To this end, we present an advanced solution based on a strong deep learning model. The approach is both straight and steady in meeting the current logistics needs. In this regard, the contributions of this work can be summarized as follows:

- Optimize the maximum workers’ tasks, and makespan simultaneously by adopting scheduling in the dual resource, which leads to strong customized logistics environments;
- Propose an efficient system to improve resource management and task scheduling, by using an advanced LSTM network, to predict task states, optimize the delivery process, and warrant better customer satisfaction;
- Improve the prediction of task accomplishments, which intelligently integrates tasks, inventory, and optimization;
- Cover a large and active part of the logistic service in a real-time scenario based on deep learning algorithms;
- Reduce the waiting time to deliver the ordered goods, by introducing an automatic and forecasting approach.

The rest of this study is organized as follows. Section II presents a general overview of the existing works that discussed smart logistics issues. Section III explains the problem statement. Section IV describes the proposed methodology. In Section V, we present the results and discuss the founding of the proposed study. The conclusion is presented in Section VI.

II. LITERATURE REVIEW

Today’s companies are influenced by the fast variations of the global market, the evolving customers’ requirements, and sustainability concerns [7]. Accordingly, e-commerce businesses are constantly looking for new advanced strategies to expand their logistics performance and warrant strong competitiveness. Particularly, the delivery distribution network, remains a key component in all logistics systems [8]. In this context, advanced delivery solutions are the most effective and operative mechanisms to increase logistics efficiency. In these last decades, industrials and researchers have increasing interests in artificial intelligence (AI) techniques and collaborative solutions to increase benefits in different logistics environments [9]. These works have been involved at varying levels of logistics task planning or scheduling. The use of AI solutions allowed designing smart logistics networks to consolidate traditional flows and reach higher economies scales [10]–[13]. In the literature, various decent approaches issues were widely proposed to smooth workflow in logistics contexts, such as route optimization [14], [15],...
warehouse and inventory management [16], traffic management [17], [18], packaging [19], and order tracking [7], [20].

Traditional e-commerce environments allow goods and services exchanged, in a digital model, while including all the stakeholders. Nevertheless, that involves numerous background tasks, that must be well accomplished to increase e-commerce performance [21]. These tasks may affect the growth of e-commerce businesses, that need to overcome different challenges and problems (e.g., system requirements and complexity, data quality, technological robustness, interoperability, the lack of connectivity and post services, and data security) [22], [23]. For that, using new advanced techniques is crucial for innovative and effective solutions. In that topic, several research practices increasingly emphasized the importance of scheduling as well as resource and task management as an effective means to balance the system requirements and increase global efficiency [24]–[27]. Based on that, a cloud-based model, using a genetic algorithm, was presented by Leung et al [28] for online order pre-processing. Another important physical e-commerce platform was introduced by Kong et al [29] to inspect the real-time processes. Also, an advanced IoT-based synchronization was proposed by Qu et al [30] to improve the efficiency of customer orders.

Much of the literature provided assistance and profits to e-commerce and logistics systems. However, the problem of centralization and real-time scheduling is still persistent. This encourages logistics companies to use adapted strategies to consolidate their flows and achieve higher economies of scale [31]–[33]. E-commerce involves strong business plans, business structures, logistics, and managed supply chain [34]. Some authors also studied the minimization of costs and the required time to accomplish logistics tasks management mechanisms. In this context, several algorithms have been presented to expand the task scheduling as well as allocation processes widely e.g.: complex manufacturing [35], freight transportation [36], product recycling [37], intelligent scheduling [38], etc. Full or dynamic scheduling driven a huge need to solve resource allocations issues in real-time systems [39], [40].

Beyond adopting traditional algorithms, advanced methods aim to unlock the potentials of logistics services by enhancing the system capacities and ensuring an equitable and effective delivery to promote e-commerce growth. Zhang et al. [41] used a classification approach based on a machine learning model to analyze the resource allocation problem in logistics systems. Similarly, the model presented by Xiaofeng et al. [42] adopted machine learning to deal with multistage random logistic tasks for better resource allocation.

As e-commerce is an evolving sector, it needs continually novel techniques to manage the web stores, users’ needs, logistics services, and resources. For that, deep learning methods seem promising to optimize the whole system and maximize the incomes. Planning, synchronizing, and scheduling have become much more based on intelligent systems for task automation, processing time minimization, and efficiency improvement [45]. To that end, mostly, resource management is related to software platforms’ needs to administrate e-commerce servers and increase the web system performance. An optimal decision-making approach was introduced by Ren et al. [46], using a deep learning architecture based on both the CNN and LSTM networks. Also, a two-phase framework was presented by Liu et al. [47] to address both resource allocation and power management on servers. The authors used deep reinforcement learning to allocate the resources, then they used an LSTM for local power management. Qiu et al. [48] designed a deep belief network (DBN) to pull out high-level information from tasks data to predict the future workload.

All over the world, e-commerce services, especially delivery, rely on various logistics operations. The adoption of advanced approaches based on AI and Deep Learning (DL) techniques afford operative solutions to solve delivery issues and increase customers’ satisfaction. On this basis, several relevant methods, table 1, introduced advanced systems from a different perspective. However, none of these methods yet addressed the automating task scheduling problem in large-scale and dynamic e-commerce systems. In particular, no research work has used LSTM for dynamic scheduling in

| Authors          | Research problem            | Method                                                                                                                                 |
|------------------|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| Zhang et al. [41] | Resource allocation         | A machine learning model for multi-dimensional cloud resource allocation issues. This approach is based on two resource allocation prediction algorithms i.e.: linear and logistic regressions. |
| Xiaofeng et al. [42] | Resource allocation      | Fuzzy classification model for multi-stage random logistics tasks. This presented multi-objective method aims to lower risks pressure and ease decision-making to optimize costs, time, and quality. |
| Zhang et al. [43] | Multi-task scheduling      | An extended NSGA-II for multi-task scheduling. This approach approximatively detects the optimal Pareto solutions considering the utilities of both manufacturers and customers. |
| Khan et al. [25]  | Resource Management        | "HeptorCloud" to empower the management of numerous tasks. This framework is based on both a single scheduler and orchestrator to ensure datacenters’ resources management, performance, and cost-efficiently. |
| Ali et al. [27]   | Resource Management        | Workload positioning and diverse migration policies i.e.: maximizing the revenues, ensuring the workload performance, reducing energy consumption, as well as reducing users’ costs and ecological impacts. |
| Chen et al. [44]  | Real-Time Scheduling        | ANN-based model task completion prediction to manage resource allocation. The proposed method offers promising potentials for smart manufacturing or Industry 4.0. |

TABLE 1. Relevant former works.
III. PROBLEM STATEMENT

In this section, we describe the scheduling problem in the logistic process and present the realization of the optimal In this section, we describe the scheduling problem in the logistic process. Recently, e-commerce businesses are experiencing rapid and evolving progress, offering a huge potential market. Unlike traditional methods, today’s e-commerce systems receive orders from customers, who purchase products online. Next, the stage of collection and packaging of the various products is carried out. Finally, the orders can be delivered.

The order fulfillment process is shown in Figure 1 includes both customers and logistics systems. Eventually, on the customer side, buying online is a fairly practical and economical way to save time, effort, and cost, especially during sales or pandemic crises. On the logistics side, task optimization can expressively improve the efficiency of the whole system activities. Compared to conventional systems, the main feature of order processing, in e-commerce frameworks, is that orders can be placed randomly, which means that collecting, packing, delivering orders are increased at all times.

An e-commerce platform brings together a set of tasks in each step of the supply chain. For that, the orders department requires a large number of resources to manage the tasks of large-scale orders. As scalability and speed are more required in evolving logistics systems, the practicable solutions are enormous and demanding many techniques, which after a certain time they are no longer applicable. In the traditional e-commerce models, demand is assumed to represent a normal distribution flow. We address the problem of order fulfillment tasks as an SL process: studying the problem of optimally scheduling tasks to ensure an efficient SL process (in other words, “what” and “how many” different resources should be distributed to properly manage many tasks while ensuring smooth operations, activities, warehouse management, and cost optimization). Then, predict task achievement using a deep learning model to ensure an innovative, adaptive, optimal, and sustainable solution using advanced AI techniques. For this, we present a real-time decision-making process for the task scheduling problem to strengthen the SL system and overcome the challenges that may occur in the future.

IV. THE PROPOSED METHODOLOGY

Inspired by the relevant literature works, in this paper, we treat resource allocation as a vital procedure for real-time planning and scheduling. For this, we present an evolutionary approach based on an LSTM network. This model allows the prediction of task status for better resource allocation. The LSTM model performs well in terms of optimization objectives, including total cost, customer satisfaction, and delivery time.

A. THE ADOPTED NOMENCLATURE

This methodology is mainly implemented to order and manage tasks in the e-commerce process to provide a solid platform for decision making and flow optimization. This paper
TABLE 2. Paper Terminology.

| Expression | Description |
|------------|-------------|
| SLO | SL Order |
| SLT | SL Delivery Tasks |
| SLR | SL resources |
| SLS | SL services |
| o | SL order (or ∈ SLO) |
| t | SL task (t ∈ SLT) |
| s | SL service (s ∈ SLS) |
| or | SL current order |
| or’ | SL tasks of the current order or |
| StartT | The start time of the task T |
| EndT | The end of the task T |
| R̄o | SL resources for the order o (o, s) |
| D̄o | Logistic duration (between o and the successor of the task T of or) |
| IST | The immediate successor set of tasks T |
| Cost | Unit cost of R̄o |
| Cap | Maximum Capacity of R̄o |
| SQ | Service Quality of R̄o |
| Pq | Processing pending queue of SLT in time t |
| PDI | The whole processing duration (make-span) |

describes the proposed methodology, using domain terminology, which is described in Table 2:

Being larger and more dynamic than traditional logistics, the SL environment is much more evolved and requires adequate scheduling for good decision-making. Therefore, we present in this paper a well-designed and effective strategy. The keys to our approach can be applied directly to the scheduling problem in the e-commerce environment.

B. TASK MANAGEMENT

Today’s market is transforming continuously, which influences the consecutively of smart logistics platforms, including e-commerce sales operations. This requires the automation of business operations as well as services. E-commerce requires logistics elements operating 24/7, with fewer human resources and further technical tools. However, to ensure the success and sustainability of the current adopted business models, it is necessary to adopt smarter supportive infrastructures. This may afford adequate management of logistics activities, and refined flows [49], [50] in the supply chain. Practically, e-commerce does not only offer a simple distribution channel but moderately involves revolutionary and intelligent changes, that impact the entire logistics process [51].

Smart logistics framework, Figure 2, offers appropriate technological solutions for good task management. It ensures the improvement of traditional logistics services, and customer satisfaction, which corresponds to the real needs of the market for such solutions. Numerous works have focused on modeling, solving, and optimizing the task scheduling problem in the traditional logistics environment. The results of their studies cannot generally be applied in any environment, as the magnitude of some problems is greatly increased. According to the academic consensus, the main types of actors involved in the tasks, the e-commerce platforms, can be described by a so-called “tri-group” model [52], [53], which can be summarized as follows:

- Suppliers must meet the increased needs of feeding stocks in reality while involving different resources, such as equipment, materials, etc;
- Online customers select several items and then place an order. This triggers different tasks, opting for resources from the SL platform;
- Operators who operate and manage all the tasks forming the process of the e-commerce platform. They decompose orders into picking tasks, encapsulate resources through technologies, and match sales departments by providing decision support tools.

C. SPECIFICITIES OF THE RESOURCE ALLOCATION MODEL

The e-commerce sales flow, in the context of our research, can be briefly described, as shown in Figure 3. The tasks, of the SL architecture, respect a priority sequence, which determines the order of level. Then, resources can be allocated to the corresponding tasks, taking into account the SLS services. These tasks will be carried out according to the following rules:

- The implementation time of the SLT is included in the processing time;
- No interruption is envisaged in the treatment of SLT;
- The capacity of the SLR resources occupied by the processing of tasks will be released when the processing is completed;
- Transportation must be considered before and after SLT treatment.

The real-time scheduling problems come mainly from the decision layer. Therefore, the purchasing process of customers, using the services provided by the electronic platform, is organized as follows:

1) Customers place an order o (o ∈ SLO) on the platform, which may contain one or more products;
2) The “o” is divided into SLT (Smart logistics Tasks);
3) A real-time update of stocks is carried out and then correspondence is established between the SLTs and the corresponding SLSs (smart logistics services);
4) Resources are allocated for the processing of tasks;
5) Tasks are organized and executed to complete the sales process.

The scheduling problem can be expressed as follows: how to reasonably allocate emerging SLT tasks to the corresponding SLR resources, in real-time, while respecting the constraints of priority, resource occupancy, and delivery time to be able to reduce costs, improve service quality, and optimize the completion time of each “o” order?

Smart logistics are gradually surpassing the capabilities of traditional logistics systems by using highly dynamic networks and advanced technological tools (such as cloud computing, geographic systems, machine learning algorithms, IoT techniques, etc.). Therefore, how to define resources to
serve real-time order delivery tasks is not only an urgent requirement but also a challenging problem for smart logistics models.

As SLR and SLR are the main components of the smart logistics environment, it is quite important to manage them accurately. Suppliers register all SLR resources (in-stock and
virtual) on the e-commerce platform. Then, after the validation of an order, the SLRs are classified according to SLS services, which is an integrated form of SLR to be used by operators later in the process. Specifically, SLS of type “s” can be expressed as a tuple, as shown in equation (1).

\[ S_i \cong (Rl_i, Ql_i) \quad \forall i \in I \]  

(1)

\( Rl_i \) represents the class of SLR resources of the same type, and \( Ql_i(t) \) indicates the resource allocation (queue level) waiting in the SLT queue at the time \( t \).

For SLRs labeled \( k \) in \( S_i \), their attributes are expressed as explained in equation 2:

\[ R_i^k \cong (Cost_i^k, SQ_i^k, Cap_i^k, Pq_i^k, A_i^k) \quad \forall s \in SLT \]  

(2)

\( A_i^k(t) \) is the set of SLTs active at time \( i \), which is defined by:

\[ A_i^k \cong \{(t, or)|Start_{i,or}^t \leq t < End_{i,or}^t, dec_{c,i} = 1\} \quad \forall s \in SLT \]  

(3)

\( dec_{c,i} \) is a decision variable \( \epsilon \{0,1\} \), equal to 1 if task \( T \) is assigned to resource \( R \), otherwise equal to zero.

In the e-commerce platform, “gold” orders placed by customers will be processed. However, the SLT allocation relationship of the process can be described as a subject for optimization. For each “gold”, the scheduling optimization aims to maximize the quality of service and minimize the costs, as well as the realization time. All these aspects are covered by a prediction system, in which we address the problem of resource allocation using a machine learning framework.

1) REAL-TIME TASK SCHEDULING

Scheduling refers to strategy control of the work to be executed. The objective of scheduling algorithms is to optimize workload, maximize resource use, and minimize execution time. For these reasons, task scheduling techniques remain important to enhance the potential of logistics systems. The main objective is to assign tasks to allocated resources based on time, which involves discovering an appropriate order in which tasks can be performed under transaction logic constraints. The main advantage of task scheduling is to achieve high system performance by regulating resource availability and ensuring maximum resource utilization. As part of real-time systems scheduling, the e-commerce process must be predictable for any kind of input. Also, resources must be multiplexed over numerous tasks, ensuring that the time limit for each task is respected.

2) MONOTONOUS SCHEDULING OF TASK RATES (RM)

In RM, which is part of static planning, the static priority of the task is mutually relative to the period of the task. Indeed, the priority of a task is determined at the time of its creation and remains invariable during execution, because MR considers the period of the task to be constant. At runtime, the ready-to-execute process with the highest priority is selected first. Then, during its execution, it can anticipate another low-priority process. For static job scheduling, MR remains a fairly practical and powerful tool. However, it is quite difficult to use static or monotonous scheduling, if the scheduling of a set of tasks cannot be done by RM. Let’s consider a set \( T = \{t_1, t_2, \ldots, t_n\} \) of \( n \) periodic tasks. \( T \) can be programmed if and only if the following conditions are met:

\[ \sum_{i=1}^{n} \frac{U_{s,ct_i}}{P_i} \leq n(2^{\frac{1}{n}} - 1) \text{and} \sum_{i=1}^{n} \frac{U_{s,ct_i}}{P_i} \leq 1 \]  

(4)

where \( U_{s,ct} \) is the most Unsuitable calculation time for \( Ti \), and \( P_i \) represents the period of \( Ti \); Ru (Resource use) represents the use of resources as \( n \) increases, and Ru use is close to \( \lim_{n \to \infty} (2^{\frac{1}{n}} - 1) = \ln 2 \).

The simplicity, stability, and performance of the MR algorithm make it easy to use for real-time systems. Its implementation for scheduling the online sales process with several fixed and high priorities is not easily missed in high load cases. In fact, in MR, maximum resource utilization is limited to 69.6%.

3) PRIORITY OF EARLIEST DEADLINE (EDF)

EDF is an algorithm of pre-empted planning (by priority). In EDF scheduling, the task selected for execution is the one with the closest due date and the deadlines of all tasks are respected for scheduling system tasks. EDF planning, a set of tasks \( T = \{T_1, T_2, \ldots, T_n\} \), can be achieved if and only if the condition \( \sum_{i=1}^{n} \frac{U_{s,ct_i}}{P_i} \leq 1 \) is satisfied. This condition is both necessary and sufficient. It also means that EDF cannot schedule this set of tasks, so the other algorithms will not satisfy the scheduling of this set of tasks. EDF planning requires less change of context compared to RM, but its implementation remains more complex, so, in heavily loaded situations, EDF performance drops rapidly. The advantage of EDF is that the use of resources can be close to 100%, efficient, and easy to calculate and derive.

4) ALTERNATE SCHEDULING (RR)

Scheduling on a rotational basis is based on a simple FIFO strategy that is not pre-empted. Embedded systems include a multitude of tasks, some with fairly strict time constraints, others that are ordinary in non-real-time, so consideration should be given to increasing scheduling, also in non-real-time, which can improve the functioning of the system as a whole. RR is simple, requires no change of context, and works well under heavy load conditions. If the overload is not taken into account, the RR algorithm can achieve 100% resource utilization.

5) IMPORTANCE OF INTEGRATED TASK SCHEDULING

Theoretically, EDF planning or scheduling will provide 100% resource utilization. However, this efficiency is difficult to implement for most processes with increased load. The EDF requires quite high operating costs due to the need to sort tasks according to the final time limit. RM planning provides...
lower operating costs, with average resource utilization well below EDF. In practical applications, these techniques (EDF and RM) alone cannot provide good performance. Therefore, this paper proposes a complete prediction system for task planning, guaranteeing the overall performance of the system.

D. PREDICTION SYSTEM DETAILS

The sales process is seen as a starting point for a workflow that ensures good use of resources with low costs and customer satisfaction. This environment involves various tasks that require accurate techniques. For that, the scheduling mechanism remains central to the sustainability and efficiency of smart logistics platforms. The proposed prediction system is shown in Figure 4. The main objective is to solve the problem of real-time scheduling and SLR resource allocation for each SLT.

Candidate SLR resources are determined based on type, capacity, and upper limit. Over time, and with the evolution of needs and the market, traditional algorithms are no longer efficient because of the need for time to verify the possible intervals for each candidate resource. This justifies our choice to adopt an LSTM model to accelerate prediction and allow a better estimation of the objective values of each candidate SLR. Specifically, let’s take the example of an SLT task. After filtering the candidate SLRs in accordance, the objective values projected onto the SLR can be estimated by the state-of-implementation forecasting model. This system provides a prediction of task completion time by studying several aspects. This is because an SLT task will only affect the execution time of its SLR if it is in a critical situation, otherwise it has a flexible interval for execution time. By using the level-order traversal algorithm, the upper and lower limits of SLT’s ideal realization time can be obtained, which should not affect the lifetime of its SLR.

- Constant Error Carousel (CEC): A central unit with a recurring connection to a weight unit. The recurrent connection represents a feedback loop with a time step equal to 1. CEC activation is the internal state that acts as the memory of past information;
- Gateway: A multiplicative unit that protects the information stored in the CEC from disturbances caused by irrelevant inputs;
- Exit door: One multiplicative unit w.

The entrance and exit door controls access to the CEC. During training, the front door learns when to let new information into the CEC. As long as the front door has a value of zero, no information is allowed inside. Similarly, the Exit Gate learns when to let information flow from the CEC. When both doors are closed (activation around zero), the information or activation is trapped in the memory cell. This allows the error signals to travel over many time steps (aided by the recurring front with the unit weight) without encountering the problem of disappearing gradients.

Our LSTM network as described above outperformed ANNs in learning the task dependencies of the long-term sales flow. Ideally, the MSTD learns to reset the contents of memory cells after completing the processing of a task.
sequence and before starting a new sequence. The components of the LSTM model are as follows:

- Input \( x_t \): The LSTM unit takes the current input, and the output of the previous time step (by the recurring edges) designated by \( o_t \).

- Entrance or input gate: The entrance door reads \( x_t \) and \( h_{t-1} \), calculates the weighted sum and applies the sigmoid activation:

  \[
  E_t = \sigma(w_{xi}x_t + w_{hi}h_{t-1} + w_{ci}c_{t-1} + b_i)
  \]  

- Forget gate: The forget gate is the mechanism by which an LSTM learns to reset the memory content when it becomes old and no longer relevant. This can happen, for example, when the network starts processing a new sequence:

  \[
  F_t = \sigma(w_{xf}x_t + w_{hf}h_{t-1} + w_{cf}c_{t-1} + b_f)
  \]  

- Memory cell: It is composed of the CEC, having a recurring edge with the unit weight. The current state of the \( C_t \) cell is calculated by omitting irrelevant information (if any) from the previous time step and accepting relevant information (if any) from the current entry:

  \[
  C_t = f_t c_{t-1} + i_t \tanh(w_{xc}x_t + w_{hc}h_{t-1} + b_c)
  \]  

- Exit door: The exit door takes the weighted sum of \( x_t \) and \( h_{t-1} \) and applies a sigmoid activation to control what information will come out of the LSTM unit.

- Output: The output of the LSTM unit is calculated by fleeting the state of the cell \( C_t \) through activation and multiplying it by the exit door \( o_t \):

  \[
  o_t = \sigma(w_{xo}x_t + w_{ho}h_{t-1} + w_{co}c_{t-1} + b_o)
  \]  

where \( \sigma(x) = \frac{1}{(1+e^{-x})} \) denotes the logistic sigmoid function, \( \tanh(x) = \frac{(e^x-e^{-x})}{(e^x+e^{-x})} \) is the hyperbolic tangent activation function, \( w \) are the weight matrices, and \( b_i, b_f, b_c, b_o \) denotes the corresponding bias vectors.

For task / resource allocation, the steps, explained in Figure 5, to implement our adopted LSTM model are summarized below:

- **Step 1: Data filtering and preparation**: This step plays an important role in system learning and data processing. Firstly, historical data are collected, preprocessed to generate time-series, and normalized in \([-1,1]\) to accelerate the training stage. Then, single-dimensional disordered data are converted into multidimensional data to prepare the model training inputs using the delay embedding theorem [54].

- **Step 2: Initialization**: All the parameters of the LSTM model are initialized randomly.

- **Step 3: Optimization**: For the training process, the parameters optimization is performed to obtain the weight matrix at each iteration. This step estimates the quality evaluation of learning indices, then the best weights are sorted and considered as the optimal for the prediction.

- **Step 4: Data computation**: For the proper functioning of the prediction system, the LSTM module must have monitoring and evaluation indicators that will provide well-targeted and relevant information about the tasks and resources of the SL system. This explains

---

**FIGURE 5.** Flowchart for the proposed LSTM model implementation.
FIGURE 6. The adopted process for the processing of tasks: This process begins with data pre-processing, which is the phase of data extraction, transformation, and loading, taking into account the list of selected characteristics. Then a phase of data pre-processing is performed for the proper modeling of resource planning in time sequences and resource labeling.

| New project | Delivery pool | Delivery services | Delivery resources | Processing |
|-------------|--------------|-------------------|--------------------|------------|
| Start Project | Decompose the project into elementary tasks | Discover and match the suitable services | Request resources’ allocation | Arrange delivery process |
| Place project into delivery pool | Place the tasks into the tasks’ pool | Filter candidate resources | Confirm allocation | End Project |
| | | Intelligent scheduling | | |

The integration of an analytical module in the proposed framework. The purpose of this module is to support the predictions made by the LSTM module to analyze the planned and actual resource allocation of tasks in a multidimensional way. This will make it possible to determine and extract a very large amount of information in terms of resources used, responses to customer needs, as well as delivery time. Also based on the analysis of resource allocation, the analysis module will above all make it possible to determine when a resource will cease to satisfy the tasks requested, which, to our knowledge, has not been proposed in previous work.

• **Step 5: Evaluation**: The obtained weights are used to evaluate and test the proposed model. All the evaluation metrics are similar for both the model training and the testing.

The integration of the LSTM module offers more visibility on the commitment, organization, achievement, and performance of resources. The data generated are integrated into a database and are used to classify resources into three classes (those valid for entering the queue, those on track, and those at risk of interruption) and will also be used for prediction analysis to provide reports that facilitate intervention and decision making. More precisely, the LSTM module receives a sequence of tasks (the tasks generated from an SLO order in the different weeks of the system), task by task, so that towards the end, it can give as a result the task class concerned, for a better prediction of future tasks. As far as the adopted LSTM architecture is concerned, the module is based on a “several to one” architecture. This will allow us to review tasks individually and take the sequence of their resources to predict the scheduling of future weeks and optimize the operation of the system. We model the resources as a sequence in time with a weekly frequency.

V. SIMULATION STUDY AND EXPERIMENTS

In this section, we describe our experimentation context and present the obtained results of the proposed prediction model.

A. EVALUATION PROCESS

To approve the validity and effectiveness of the proposed framework, experiments were conducted respecting the process described in Figure 6.

This process of extracting relevant, informative, and distinctive features from the data can then be used as an input to our machine learning algorithm. The quality of the characteristics largely influences the quality of the results. Thus, feature engineering is one of the key factors in the success of a learning system. Although this process is generally creative and labor-intensive, good progress can be made if one understands the methodology, the secrets of the trade, and the common pitfalls. Feature engineering, therefore, remains an essential step because it directly influences the performance of any algorithm and therefore the final decision making. This extraction of characteristics often calls, in the first instance, on the common sense and expertise of people who have proven themselves in the field under study.
B. EVALUATION BASELINES

Many standard data mining and machine learning algorithms are often provided by numerous libraries, toolboxes, and platforms. However, simply running the adopted algorithm on a data set does not guarantee good results. The data must be provided to the algorithm properly. More precisely, the most relevant, informative, and distinctive information is extracted from the raw data (which is not directly usable in all cases) by a feature engineering phase.

To assess the proposed system performance, we adopted tests. First, the system is tested using simulation datasets to verify the results and the changes that can occur when tasks are decreased or increased. Then, a comparative analysis was conducted to evaluate the proposed system compared to the most relevant anterior literature methods, presented in Table 1 (a- Zhang et al. [41], b- Xiaofeng et al. [42], c- Zhang et al. [43], d- Khan et al. [25], e- Ali et al. [27], f- Chen et al. [44], g- Our proposed method) For the comparative analysis, the training parameters of all the tested methods were initialized. For the machine learning-based ones, they were set using the initialization of He et al. [55] and optimized by stochastic gradient descent (a momentum of 0.9, and an initial learning rate of 10e-6). Accordingly, training, validation, and testing of each experiment were performed on a machine equipped with NVidia TitanX GPU with 12 GB of memory. The methods were implemented using the Python 3.6 library, 1.1.0 [56].

C. DATASET DESCRIPTION

To verify the proposed approach, we are conducting various experiments to test the quality of scheduling based on an LSTM network for real-time tasks. The experiments were directed using publicly available data: MPSPLib. This database offers a multitude of project instances, an example of which is shown in Table 3. An order is considered an MPSPLib project. So, we use the data of this project (tasks, resources, capacities, ...) for the tests of our system.

Projects in the MPSPLib can be grouped as follows:

- SLT-30: Each project in this group contains 30 SLTs;
- SLT-60: Each project in this group contains 60 SLTs;
- SLT-90: Each project in this group contains 90 SLTs;
- SLT-120: Each project in this group contains 120 SLTs;
- SLT-mix: each project of this group can have an arbitrary value of the set 30, 60, 90, 120.

To mimic the project flow of an e-commerce platform, each tested MPSPLib project is activated with the following random characteristics:

At this stage, and taking into account several criteria, such as a large number of features, their variety, and the involvement of human expertise, we called upon experts in the field to establish a better basis for our system. Once the data is prepared, the learning stage can be established.

D. EVALUATION MEASURES

All the numeric results shown in this paper refer to the average of the whole obtained results according to the performed tests. To inspect our proposed system, we have adopted two categories of evaluation measures:

- In the SL frame, the variance of the error magnitudes is quite interesting. Contrary to a classical classification problem, the objective of a regression problem, in the logistic framework, is not to make predictions on a discrete variable but continuous values. However, in a regression problem, different comparisons can be made and measurements can be considered to evaluate an intelligent system in an e-commerce platform. In this work, we test the learning of the proposed LSTM-based system through 4 measures i.e.: MAE, MSE, RMSE, and MAPE;
- To compare the actual findings and the efficiency of the proposed system with previous literature methods, we adopted several metrics i.e.: accuracy, precision, recall, AUC, ROC, and F1 score.

E. PERFORMANCE OF THE PROPOSED SYSTEM

Once our ML-based model is designed, a training phase is necessary for its operation. In the case of neural network-based models, several optimization algorithms are used to make these models learn, including the Stochastic Gradient Descent Method (SGD). The utility of such an algorithm is to find a set of internal model parameters that work well against certain performance measures such as accuracy or RMSE error. The SGD algorithm refers to the calculation of an error gradient or error slope and the movement along this slope towards a minimum error level. The algorithm is iterative, which means that the search process takes place in several
discrete steps, each of which slightly improves the parameters of the model. Each step consists of using the model with the current set of internal parameters to make predictions on certain samples, comparing the predictions with the expected actual results, calculating the error, and using it to update the internal parameters of the model.

The SGD is a learning algorithm with several parameters, including the number of epochs and the batch size. The batch size is a parameter that defines the number of samples to be processed before the internal model parameters are updated. A batch can be considered as a loop of iterations on one or more samples to make predictions. At the end of the batch, the predictions are compared to the expected output variables and an error is calculated. From this error, the update algorithm is used to improve the model. In this case, the training data set is divided into one or more batches. When all training samples are used to create a batch, the training algorithm is called Batch Gradient Descent. When the lot has the size of a sample, the learning algorithm is called “Stochastic Gradient Descent”. When the batch size is smaller than the size of the training data set, the training algorithm is called “Mini-Batch Gradient Descent”.

As for the epoch number, it is a parameter that defines the number of times the training algorithm will run on the totality of the training data. An epoch means that each sample in the training data set had the opportunity to update the internal parameters of the model. An epoch is composed of one or more lots. A good configuration

![FIGURE 7. Accuracy values according to epochs.](image7)

![FIGURE 8. Error rate according to epochs.](image8)
of the predictive model surely leads to better accuracy rates. This configuration includes the choice of activation and loss functions and the optimization procedure. It is defined by experiments until the network starts to generalize efficiently.

In this work, we devoted 70% of all data to the learning phase, 20% to testing, and 10% to validate the entire process. To activate the neurons of the LSTM generator model, the ReLu (Rectified Linear Unit) function was introduced as a non-linear activation function known by the mathematical formula:

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases}$$

It is inserted after each layer for its efficiency because it allows the network to converge very quickly and its non-linearity allows backpropagation. The training was performed on 200 Epoch with a size of 64 Mini-Batch Gradient Descent.

To test and validate the performance of the predictor model, the curves presented in Figures 7 and 8 show respectively the accuracy and error values calculated each time for the training and test data over the different epochs from the initial value of 20.

After recording transaction values from 20 to 100 epochs, accuracy peaks at 115 - epoch with a value of 91.5% for training data and 86.1% for test data. Then, this value stabilizes at 93.85% and 88.37% for training and test data.
respectively after the 130th epoch. A similar scenario corresponds to the error values that mark extreme rates of 7% and 13% respectively for training and test data from epoch 115 onwards.

In the following, we present the results obtained during the learning and testing phases of the proposed model using the error measures listed in the previous section. Table 5 shows the planning errors per week, and thus per output sequence. After the training phase, we used all the test data to validate the proposed model. Table 6 shows the error values obtained after the test.

For the LSTM module, we tested its performance in the learning and testing phases. Table 7 shows the measurement values obtained in both phases. We can notice that the performance rates as well as the error calculations prove the validity and performance of our model. This ensures that our approach reinforces the SL environment and allows correct scheduling, prediction of task accomplishments for flow optimization, and good decision making.

### FIGURE 11. Performance variations of the SLT-60 dataset.

![Graph showing performance variations of the SLT-60 dataset.]

### FIGURE 12. Performance variations of the SLT-90 dataset.

![Graph showing performance variations of the SLT-90 dataset.]

### TABLE 5. Drive error values for 120 epochs.

| Week | MAE  | MSE  | RMSE | MAPE |
|------|------|------|------|------|
| 2    | 0.0369 | 0.0136 | 0.116 | 3.58 |
| 3    | 0.0322 | 0.0103 | 0.101 | 3.17 |
| 4    | 0.0452 | 0.0214 | 0.146 | 3.64 |
| 5    | 0.0355 | 0.0150 | 0.122 | 3.69 |
| 6    | 0.0449 | 0.0129 | 0.113 | 4.28 |
| 7    | 0.0342 | 0.0138 | 0.117 | 3.86 |
| 8    | 0.0435 | 0.0192 | 0.138 | 3.98 |
| 9    | 0.0378 | 0.0156 | 0.124 | 3.65 |
The ROC curve, displayed in figure 8, illustrates the trade-off among specificity and sensitivity rates to assign the best cut-offs points. Since the overall Area Under the Curve (AUC) is usually well-defined as the area under the ROC curve, it is obvious that the test performance of the proposed system proved promising results. On the ROC curve, each point displays a specificity/sensitivity pair corresponding to a specific decision threshold. The area under the curve proves that the presented system offers good prediction quality.

**F. COMPARATIVE ANALYSIS**

The main objective of this benchmarking study is to demonstrate the potential advantages of our logistics resource
allocation methodology, used in a practical case, compared to existing strategies. The case study examines a data set of project tasks. To facilitate the conduct of this study, we adopted the same data sets for a fair comparison. Consequently, the performance of the proposed method was evaluated by comparison with the relevant existing resource allocation strategies already defined in the evaluation baselines (a- Zhang et al. [41], b- Xiaofeng et al. [42], c- Zhang et al. [43], d- Khan et al. [25], e- Ali et al. [27], f- Chen et al. [44], and g- Our proposed method).

The Zhang et al. [43] approach is considered as a reference for the comparative analysis. More precisely, this NSGA-II based method allows resolution via approximation. The NSGA-II segmentation factor $\sigma \in [0, 1]$ indicates the degree of division of a given problem. In our study, we consider $\sigma = 0$ to mimic the effect of real-time planning. The first test of comparative experiments is based on comparing the performance of these methods in terms of the total cost, service satisfaction, and profitability. Figures 9, 10, 11, 12, and 13 illustrate the results of this test for each of the project datasets.

In each data set (SLT), the proposed approach demonstrated quite remarkable results in terms of cost compared to the comparative methods (a, b, c, d, e, and f). Although the benefits of our method in terms of service satisfaction and cost-effectiveness are not significant enough, our method, too, has been able to achieve higher levels for these two criteria. Concerning service satisfaction testing, all the methods compared showed almost similar performance, which is mainly due to the limited number of candidates SLRs in each state in the dataset. It is obvious that the number of SLT tasks proportionally influences service performance and satisfaction (the higher the number of SLT tasks, the better the satisfaction and performance).

Another important scheduling test is illustrated in Figure 14. This test describes the variations in task accomplishment times for each method in the different SLT datasets. Task accomplishment time is exponential to SLT sets criteria. A distribution with one of the minimum tasks gains acceptable time means that it does not influence overall system decision making. However, for SLTs with more tasks, accomplishment time increases considerably.

Unlike the previous test, which aimed to assess the service performance and satisfaction of the proposed approach, the second comparative test aims to check the accuracy rates of the obtained results. We can infer from Table 8 that the performance of our prediction system is higher, mainly because the system takes into account the different details of the SL environment. Thus, the LSTM model processes the project characteristics to predict a stable result that is efficient. In each group of SLT sets, our LSTM-based method has an absolute performance rate advantage over other methods. However, the anterior approaches provided very competitive results. It can be seen that machine learning approaches provide better system cost-effectiveness.

In addition to performance testing, another important indicator to be studied is the reactivity of each method. The average decision time for resource allocation is approximately 58.84 sec, indicating that the LSTM-based approach is better suited for real-time scheduling because, compared to other
TABLE 8. Evaluation results of the different methods.

| Dataset | Method       | Accuracy (%) | Precision (%) | Recall (%) | Score F1 (%) | AUC (%) |
|---------|--------------|--------------|---------------|------------|--------------|---------|
| SLT-30  | a- Zhang et al. [41] | 85.01%       | 84.80%        | 91.03%     | 89.30%       | 76.70%  |
|         | b- Xiaofeng et al. [42] | 89.67%       | 91.05%        | 92.55%     | 91.72%       | 83.35%  |
|         | c- Zhang et al. [43] | 85.73%       | 91.11%        | 92.61%     | 91.78%       | 83.41%  |
|         | d- Khan et al. [25] | 88.60%       | 89.98%        | 91.48%     | 90.65%       | 82.28%  |
|         | e- Ali et al. [27] | 84.51%       | 83.69%        | 89.92%     | 88.19%       | 75.59%  |
|         | f- Chen et al. [44] | 88.17%       | 87.52%        | 91.11%     | 89.69%       | 80.00%  |
|         | g- Our proposed method | 90.93%       | 92.31%        | 93.81%     | 92.98%       | 84.61%  |
| SLT-60  | a- Zhang et al. [41] | 85.71%       | 84.89%        | 91.12%     | 89.39%       | 76.79%  |
|         | b- Xiaofeng et al. [42] | 90.49%       | 91.87%        | 93.37%     | 92.54%       | 84.17%  |
|         | c- Zhang et al. [43] | 90.55%       | 91.93%        | 93.43%     | 92.60%       | 84.23%  |
|         | d- Khan et al. [25] | 89.42%       | 90.80%        | 92.30%     | 91.47%       | 83.10%  |
|         | e- Ali et al. [27] | 85.33%       | 84.51%        | 90.74%     | 89.01%       | 76.41%  |
|         | f- Chen et al. [44] | 88.26%       | 87.61%        | 91.20%     | 89.87%       | 80.09%  |
|         | g- Our proposed method | 91.02%       | 92.40%        | 93.90%     | 93.07%       | 84.71%  |
| SLT-90  | a- Zhang et al. [41] | 86.31%       | 86.10%        | 92.33%     | 90.6%        | 78.00%  |
|         | b- Xiaofeng et al. [42] | 91.60%       | 92.98%        | 94.48%     | 93.65%       | 85.28%  |
|         | c- Zhang et al. [43] | 91.66%       | 93.04%        | 94.54%     | 93.71%       | 85.34%  |
|         | d- Khan et al. [25] | 90.53%       | 91.91%        | 93.41%     | 92.58%       | 84.21%  |
|         | e- Ali et al. [27] | 86.44%       | 85.62%        | 91.85%     | 90.12%       | 77.52%  |
|         | f- Chen et al. [44] | 89.32%       | 88.67%        | 92.26%     | 90.84%       | 81.15%  |
|         | g- Our proposed method | 92.61%       | 93.99%        | 95.46%     | 94.66%       | 86.29%  |
| SLT-120 | a- Zhang et al. [41] | 87.08%       | 86.87%        | 93.11%     | 91.37%       | 78.77%  |
|         | b- Xiaofeng et al. [42] | 92.91%       | 94.29%        | 95.59%     | 94.96%       | 86.59%  |
|         | c- Zhang et al. [43] | 92.97%       | 94.15%        | 95.65%     | 94.91%       | 86.65%  |
|         | d- Khan et al. [25] | 91.84%       | 93.22%        | 94.52%     | 93.89%       | 85.52%  |
|         | e- Ali et al. [27] | 87.75%       | 86.93%        | 92.96%     | 91.43%       | 78.83%  |
|         | f- Chen et al. [44] | 90.24%       | 89.59%        | 93.18%     | 91.76%       | 82.07%  |
|         | g- Our proposed method | 93.09%       | 94.38%        | 95.88%     | 95.05%       | 86.65%  |
| SLT-MIX | a- Zhang et al. [41] | 88.67%       | 88.46%        | 94.69%     | 92.96%       | 80.36%  |
|         | b- Xiaofeng et al. [42] | 93.08%       | 94.56%        | 95.59%     | 94.96%       | 86.39%  |
|         | c- Zhang et al. [43] | 93.24%       | 94.42%        | 95.92%     | 95.18%       | 86.65%  |
|         | d- Khan et al. [25] | 92.11%       | 93.49%        | 94.69%     | 94.16%       | 85.52%  |
|         | e- Ali et al. [27] | 88.02%       | 87.20%        | 93.13%     | 91.70%       | 78.83%  |
|         | f- Chen et al. [44] | 91.83%       | 91.18%        | 93.35%     | 93.35%       | 83.66%  |
|         | g- Our proposed method | 94.59%       | 95.97%        | 96.87%     | 96.64%       | 88.27%  |

TABLE 9. Average time for decision making.

| Methods     | Average decision time |
|-------------|-----------------------|
| a- Zhang et al. [41] | 96.43 sec |
| b- Xiaofeng et al. [42] | 135.5 sec |
| c- Zhang et al. [43] | 157.24 sec |
| d- Khan et al. [25] | 162.41 sec |
| e- Ali et al. [27] | 169.82 sec |
| f- Chen et al. [44] | 94.26 sec |
| g- Our proposed method | 58.84 sec |

methods, the decision time for determining proper scheduling in such a discrete SL environment is negligible compared to the duration of resource configuration, order fulfillment, transportation, etc., which are typically measured in hours or days.

VI. DISCUSSION
To test the performance of the proposed model, based on the time series of the LSTM network, a set of simple and effective measures has been adopted. These are the most commonly used loss functions in regression problems. Their simplicity lies in the fact that they simply calculate the distance between the data and the prediction values and by minimizing them, the model will be able to make near-perfect predictions.

First, we used Mean Absolute Error (MAE), the most intuitive and easiest to interpret measure because it simply examines the absolute difference between the data and the model’s predictions. Given this fact, the MAE does not indicate an under- or over-adjustment of the model (whether or not the model exceeds the actual data). Each difference contributes proportionally to the total amount of error, which means that larger errors will contribute linearly to the overall error. As presented in Tables 5 and 6, for both the learning phase and the test phase, the results obtained indicate a low MAE that tends towards 0 (approximate values of 0.035 on average for the learning phase and 0.05 for the test phase). This suggests that the model is a very good predictor for all weeks.

Second, two other fairly similar measures were used, namely MSE (root mean square error) and RMSE (root mean square error). As its name implies, RMSE is the square root of...
MSE. However, the former is more common than the latter for several reasons. First of all, RMSE allows the error measurement to be reconverted into similar units, which facilitates interpretation. Moreover, being symmetrical and quadratic, RMSE allows us to deal with Gaussian noise problems. Like MAE, MSE and RMSE can be set between 0 and positive infinity. From the results in Tables 5 and 6, we see that the values of the MSE (mean values of 0.015 in the learning phase and 0.025 in the testing phase) and the RMSE (learning and testing values around 0.1) are small enough to conclude that the proposed model gives very satisfactory and smaller forecasts for all weeks.

However, if these two measures increase, it becomes more difficult to interpret the performance of the model. To address this problem, we introduced a fourth measure using percentages, so that each prediction is scaled to the value it is supposed to estimate. For this, MAPE (Mean Absolute Percentage Error Mean Error) was used to normalize or weight the errors by the inverse of the actual value of the observation. This function is relevant because it makes it easier to visualize the margins of error. These do not exceed 4.3% during the learning phase (maximum of 4.28%) and less than 6.2% during the test phase (6.17% is the maximum value obtained), showing that the proposed LSTM model is a good tool for predicting the scheduling of future tasks within the SL environment. On the task scheduling side, the proposed system ensures a very satisfactory accuracy in all evaluation measures. At this level, our SL approach ensures, in the test phase (validation), the prediction accuracy of 92.43%. However, the main limitation is the small number of datasets. This study focuses on how the proposed model can learn from limited training cases. Accordingly, for our approach, we believe the number of training dataset is acceptable. More importantly, this study is also applicable to general task scheduling problems. In addition, the number of experiments cases is comparable with recent works.

VII. CONCLUSION

In this paper, we studied the problem of real-time scheduling in the SL environment. After presenting the premises and background, several research contents and results can be highlighted:

- We have built a mathematical model for real-time scheduling in the SL environment with optimization objectives such as minimizing cost, minimizing delivery time, and maximizing service satisfaction, which remains proportional to the priority of tasks, the occupation of resources, and the duration of logistics;
- An LSTM network-based approach to real-time scheduling has been designed that uses tasks and resource queues as inputs to predict task completion status, resource allocation, and provides a prediction of task scheduling for future weeks;
- We conducted comparative experiments; the results show that:
  - The approach based on the LSTM model is more efficient than other methods;
  - The response time of the proposed prediction system is optimal;
  - Using the module, the average decision time for resource allocation is around 50 s, indicating that the proposed approach is suitable for real-time scheduling.

The proposed method in this study can be easily integrated into any e-commerce platform and will be a high value-added tool to help managers of SL systems to have an idea about task scheduling, resource allocation, prediction of future tasks, the efficiency of deployed resources, upgrading, monitoring and evaluation of the different components of the SL environment.

What makes the proposed framework very competitive is its LSTM module that allows generating task schedules for the rest of the weeks from the tasks and resources scheduled in the first week of the month. In other words, by the end of the first week, the e-commerce manager will have visibility based on analyzable indicators of system performance.

For future work, we can consider order fulfillment, especially delivery activities, as a detailed process of the logistics environment. That requires effective optimization approaches to meet the high demand, real-world variations, with a greater focus to enhance the functioning of e-commerce systems.

REFERENCES

[1] O. Szymańska, M. Adamczak, and P. Cyplik, “Logistics 4.0—A new paradigm or set of known solutions?” Res. Logistics Prod., vol. 7, no. 4, pp. 299–310, 2017, doi: 10.21008/f.2083-4950.2017.7.4.2.
[2] A. Bask, H. Merisalo-Rantanen, and T. Tuunanen, “Developing a modular service architecture for E-store supply chains: The small- and medium-sized enterprise perspective,” Service Sci., vol. 6, no. 4, pp. 251–273, Dec. 2014, doi: 10.1287/serv.2014.0082.
[3] S. A. Torabi, E. Hassini, and M. Jelchoonian, “Fulfillment source allocation, inventory transshipment, and customer order transfer in e-tailing,” Transp. Res. E, Logistics Transp. Rev., vol. 79, pp. 128–144, Jul. 2015.
[4] Z. Hong, H. Wang, and Y. Yu, “Green product pricing with non-green product reference,” Transp. Res. Pt. e-Logist. Transp. Rev., vol. 115, pp. 1–15, Jul. 2018.
[5] S. Hochreiter, “The vanishing gradient problem during learning recurrent neural nets and problem solutions,” Int. J. Uncertainty, Fuzziness Knowl.-Based Syst., vol. 6, no. 2, pp. 107–116, Apr. 1998.
[6] G. Liu and J. Guo, “Bidirectional LSTM with attention mechanism and convolutional layer for text classification,” Neurocomputing, vol. 337, pp. 325–338, Apr. 2019.
[7] Y. Issaoui, A. Khiat, A. Bahnasse, and H. Ouajji, “Toward smart logistics: Engineering insights and emerging trends,” Arch. Comput. Methods Eng., vol. 28, no. 4, pp. 3183–3210, Jun. 2021, doi: 10.1007/s11831-020-09494-2.
[8] A. Muñoz-Villamizar, C. L. Quintero-Araújo, J. R. Montoya-Torres, and J. Faulin, “Short- and mid-term evaluation of the use of electric vehicles in urban freight transport collaborative networks: A case study,” Int. J. Logistics Res. Appl., vol. 22, no. 3, pp. 229–252, May 2019.
[9] S. Pan, D. Trelentes, E. Ballot, and G. Q. Huang, “Horizontal collaborative transport: Survey of solutions and practical implementation issues,” Int. J. Prod. Res., vol. 57, nos. 15–16, pp. 5340–5361, Aug. 2019.
[10] E. Ballot and F. Fontane, “Reducing transportation CO2 emissions through pooling of supply networks: Perspectives from a case study in French retail chains,” Prod. Planning Control, vol. 21, no. 6, pp. 640–650, Sep. 2010.
[11] S. Pan, E. Ballot, and F. Fontane, “The reduction of greenhouse gas emissions from freight transport by pooling supply chains,” Int. J. Prod. Econ., vol. 143, no. 1, pp. 86–94, 2013.
[12] L. Verdonck, P. Beullens, A. Caris, K. Ramaekers, and G. K. Janssens, “Analysis of collaborative savings and cost allocation techniques for the cooperative carrier facility location problem,” J. Oper. Res. Soc., vol. 67, no. 6, pp. 853–871, Jun. 2016.

[13] S. Nataraj and D. Ferone, “Consolidation centers in city logistics: A cooperative approach based on the location routing problem,” Int. J. Ind. Eng. Comput., vol. 10, no. 3, pp. 393–404, 2019.

[14] C. Defryn, K. Sörensen, and W. Dullaert, “Integrating partner objectives in horizontal logistics optimisation models,” Omega, vol. 82, pp. 1–12, Jan. 2019.

[15] Y. Wang, S. Zhang, X. Guan, S. Peng, H. Wang, Y. Liu, and M. Xu, “Collaborative multi-depot logistics network design with time window assignment,” Expert Syst. Appl., vol. 140, Feb 2020, Art. no. 112910.

[16] T. Kučera and J. Chocholáč, “Design of the city logistics simulation model using a 3D VISSIM software,” Transp. Res. Proc., vol. 53, pp. 258–265, Jan. 2021.

[17] M. Mahmoudi and I. Parviziomran, “Reusable packaging in supply chains under chaos environment with disruptions: Based on coupled map lattices,” Int. Food Econ. Alternatives, vol. 1, pp. 239–262, 2020.

[18] T. R. Amaral and E.-H. Aghezzaf, “City logistics and traffic management: Modelling the inner and outer urban transport flows in a two-tiered system,” Transp. Res., vol. 6, pp. 97–312, Jan. 2015.

[19] T. Kučera and J. Chocholáč, “Design of the city logistics simulation model using 3D VISSIM software,” Transp. Res. Proc., vol. 53, pp. 258–265, Jan. 2021.

[20] M. Mahmoudi and I. Parviziomran, “Reusable packaging in supply chains under chaos environment with disruptions: Based on coupled map lattices,” Int. J. Ind. Eng. Comput., vol. 13, no. 3, p. 1356, Jan. 2021.

[21] M. B. Qureshi, A. A. Khan, M. Zakarya, and M. Fayaz, “An energy and performance aware scheduler for real-time tasks in cloud datacenters,” IEEE Access, vol. 8, pp. 161288–161303, 2020, doi: 10.1109/ACCESS.2020.3020843.

[22] A. A. Khan, M. Zakarya, I. U. Rahman, R. Khan, and R. Buyya, “Hep-orCloud: An energy and performance efficient resource orchestrator for hybrid heterogeneous cloud computing environments,” J. Netw. Comput. Appl., vol. 173, Jan. 2021, Art. no. 102869.

[23] M. Zakarya and L. Gillam, “Managing energy, performance and cost in large scale heterogeneous datacenters using migrations,” Future Gener. Comput. Syst., vol. 93, pp. 529–547, Apr. 2019, doi: 10.1016/j.future.2018.10.047.

[24] H. Ali, M. S. Qureshi, M. B. Qureshi, A. A. Khan, M. Zakarya, and M. Fayaz, “An energy and performance aware scheduler for real-time tasks in cloud datacenters,” IEEE Access, vol. 8, pp. 161288–161303, 2020, doi: 10.1109/ACCESS.2020.3020843.

[25] A. A. Khan, M. Zakarya, I. U. Rahman, R. Khan, and R. Buyya, “Hep-orCloud: An energy and performance efficient resource orchestrator for hybrid heterogeneous cloud computing environments,” J. Netw. Comput. Appl., vol. 173, Jan. 2021, Art. no. 102869.

[26] Z. Song, Y. Sun, J. Wan, L. Huang, and J. Zhu, “Smart E-commerce systems: Current status and research challenges,” Electron. Markets, vol. 29, pp. 221–238, Jun. 2019.

[27] Y. Yakuulenko, “Service innovation in E-commerce last mile delivery: Mapping the e-customer journey,” Comput. Ind. Eng., vol. 101, pp. 898–912, Aug. 2019.

[28] H. Ali, M. S. Qureshi, M. B. Qureshi, A. A. Khan, M. Zakarya, and M. Fayaz, “An energy and performance aware scheduler for real-time tasks in cloud datacentres,” IEEE Access, vol. 8, pp. 161288–161303, 2020, doi: 10.1109/ACCESS.2020.3020843.

[29] A. A. Khan, M. Zakarya, I. U. Rahman, R. Khan, and R. Buyya, “Hep-orCloud: An energy and performance efficient resource orchestrator for hybrid heterogeneous cloud computing environments,” J. Netw. Comput. Appl., vol. 173, Jan. 2021, Art. no. 102869.

[30] M. Zakarya and L. Gillam, “Managing energy, performance and cost in large scale heterogeneous datacenters using migrations,” Future Gener. Comput. Syst., vol. 93, pp. 529–547, Apr. 2019, doi: 10.1016/j.future.2018.10.047.

[31] H. Ali, M. Zakarya, I. U. Rahman, A. A. Khan, and R. Buyya, “FollowMe@LS: Electricity price and source awareness resource management in geographically distributed heterogeneous datacenters,” J. Syst. Softw., vol. 175, May 2021, Art. no. 110907.

[32] K. H. Leung, K. L. Choy, P. K. Y. Siu, G. T. S. Ho, H. Y. Lam, and C. K. M. Lee, “A B2C E-commerce intelligent system for re-engineering the e-order fulfilment process,” Expert Syst. Appl., vol. 91, pp. 386–404, Jan. 2018.

[33] X. T. R. Kong, M. Li, Y. Yu, Z. Zhao, and G. Q. Huang, “Physical internet-enabled E-commerce logistics park platform,” in Proc. 13th IEEE Conf. Autom. Sci. Eng. (CASE), Aug. 2017, pp. 406–411, doi: 10.1109/COASE.2017.8266138.

[34] T. Qu, Y. Pan, X. Liu, K. Kang, C. Li, M. Thurer, and G. Q. Huang, “Internet of Things-based real-time production logistics synchronization mechanism and method toward customer order dynamics,” Trans. Inst. Meas. Control, vol. 39, no. 4, pp. 429–445, Apr. 2017.

[35] X. Liu, K. Wen, Y. Xu, and G. Yu, “Non-linear behavior of supply chains under chaos environment with disruptions: Based on coupled map lattices,” Intell. Decis. Technol., vol. 11, no. 3, pp. 399–414, Aug. 2017.

[36] A. Serrano-Hernandez, J. Fauluin, P. Hirsch, and C. Fikar, “Agent-based simulation for horizontal cooperation in logistics: Transportation: From the individual to the granular level,” Simul. Model. Pract. Theory, vol. 85, pp. 47–59, Jun. 2018.

[37] M. M. Nejad, L. Mashayekhy, and D. Grosu, “Truthful greedy mechanisms for dynamic virtual machine provisioning and allocation in clouds,” IEEE Trans. Parallel Distrib. Syst., vol. 26, no. 2, pp. 594–603, Feb. 2015.

[38] L. Mashayekhy, M. M. Nejad, D. Grosu, and A. V. Vasilakos, “An online mechanism for resource allocation and pricing in clouds,” IEEE Trans. Comput. Syst., vol. 45, no. 4, pp. 1172–1184, Apr. 2016.

[39] H. Ali, M. Zakarya, I. U. Rahman, A. A. Khan, and R. Buyya, “FollowMe@LS: Electricity price and source awareness resource management in geographically distributed heterogeneous datacenters,” J. Syst. Softw., vol. 175, May 2021, Art. no. 110907.

[40] R. Hegger, H. Kantz, and T. Schreiber, “Practical implementation of nonlinear time series methods: The TISEAN package,” J. Stat. Softw., vol. 4, pp. 60–71, Dec. 2015.

[41] N. Dragomirov, (in Bulgarian), Information Systems in Logistics, Mladost, Sofia: Publishing Complex UNWE, 2014.

[42] J. Zarek, “E-commerce influence on changes in logistics processes,” Logforum, vol. 11, no. 2, pp. 129–138, 2015, doi: 10.17270/JLOG.2015.2.1.

[43] W. He and L. Xu, “A state-of-the-art survey of cloud manufacturing,” Int. J. Comput. Integr. Manuf., vol. 28, no. 3, pp. 239–250, Mar. 2015.

[44] X. Xu, “From cloud computing to cloud manufacturing,” Robot. Comput.-Integr. Manuf., vol. 28, no. 1, pp. 75–86, Feb. 2012.

[45] B. Hegger, H. Kants, and T. Schreiber, “Practical implementation of nonlinear time series methods: The TISEAN package,” Chaos, Interdiscipl. J. Nonlinear Sci., vol. 9, no. 2, pp. 413–435, 1999.

[46] M. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 1026–1034.

[47] A. Paszke and S. Gross, “Automatic differentiation in PyTorch,” in Proc. NIPS Autodiff Workshop, 2017, pp. 1–4.
YASSINE ISSAOUI received the M.S. degree from Hassan II University of Casablanca, in 2014, where he is currently pursuing the Ph.D. degree with the Signals, Distributed Systems and Artificial Intelligence Laboratory. His research interests include artificial intelligence, machine learning, deep learning, blockchain, project management, smart logistic, the Internet of Things, industry 4.0, smart supply chain management, smart manufacturing, smart grid, and smart city. He is an outstanding reviewer of various indexed journals.

AZEDDINE KHAT received the Ph.D. degree in computer science, networks, and telecommunications from Hassan II University of Casablanca, Morocco. He is currently a Professor and a Researcher with the Department Mathematics and Computer Science, ENSET Mohammedia, Hassan II University of Casablanca. He is also a Research Member with SSDIA Laboratory, Network and Telecommunications Team, ENSET Mohammedia, Hassan II University of Casablanca. He has published tens of refereed journal and conference papers. His research interest include wireless networks, QoS on mobile networks, handover on wireless networks, networks and telecommunications, SDN, new generation networks, security, mobile learning, the IoT, smart city, smart grid, and industry 4.0. He is an organizing committee member and a technical program committee member of tens of international conferences. He is an outstanding reviewer of various indexed journals. He certified on several products and solutions.

AYOUB BAHNASSE received the Ph.D. degree in networks and telecommunication from Chouaib Doukkali University, Morocco. He is currently a Professor and a Researcher with ENSAM Casablanca, Hassan II University of Casablanca, Morocco. He is also a Research Associate with the Information Treatment Laboratory, Faculty of Sciences Ben M’sik, Hassan II University of Casablanca. He is a member of “Big data and Internet of Things project for urban service” Project, Hassan II University of Casablanca. He has published tens of refereed journal and conference papers. His research interests include new generation networks, security, mobile learning, wireless networks, quality of service, multimedia systems, the IoT, smart city, and MPLS. He is an organizing committee member and a technical program committee member of tens of international conferences around 20 countries. He is an Outstanding Reviewer of various indexed journals, such as Computer Networks journal (Elsevier), Computers & Electrical Engineering (Elsevier), Computers & Security (Elsevier), IEEE TRANSACTIONS ON COMMUNICATIONS, Telecommunication Systems (Springer), Wireless Personal Communications (Springer), Mobile Networks and Applications (Springer), and tens of indexed journals. He has certified on several products and solutions among them: Fortinet NSE4, Microsoft Azure Administrator Associate, Juniper DevOps Associate, Juniper Cloud Associate, Juniper Design Associate, and Huawei RS Associate.

HASSAN LAJMI received the degree (higher education) from the Department of Electronic and Electrical Engineering, ENSET Mohammadia. He is currently a Full Professor with the Department of Electronic and Electrical Engineering, ENSET Mohammadia. He is also a Research Member with SSDIA Laboratory and the Head of Network and Telecommunications Team, Hassan II University of Casablanca, Morocco. He has published tens of refereed journal and conference papers. He is a member of several scientific project with Hassan II University of Casablanca. His research interests include acoustic electrical engineering, signal processing, electronic, networks, and telecommunications. He is an outstanding reviewer of various indexed journals. He is an organizing committee member and a technical program committee member of tens of international conferences.