Automate Employees Effiiency Gauging and Task Scheduling

Purusothman P[1], Piriyangan K[2] and Sasini Nuwanthika[3]

[1]Faculty of Computing (CSNE), Sri Lankan Institute of Information Technology, SRI LANKA

[2]Faculty of Computing (CSNE), Sri Lankan Institute of Information Technology, SRI LANKA

[3]Department of Computer System Engineering, Sri Lankan Institute of Information Technology, SRI LANKA

[1]Corresponding Author: purushothpurus1998@gmail.com

ABSTRACT

There is a common fact in the world that all employees and employers are not satisfied with themselves. This is due to a lack of awareness about the work done by employees. But most of the time employees put their maximum effort into their work and didn't get recognized by the employer. As a result, the challenge is identifying the efficiency of the employees and managing their respective schedules of tasks. This paper proposes an approach to help employers identify the employees' efficiency in an urgent situation. Also, this approach can help to manage the work allocation by their efficiency to the respective tasks. This paper will discuss how it could be used in real-time scenarios autonomously. This paper presents a practical method for modeling and solving a dynamic resource allocation of automatic scheduling problem using a forward gauging heuristic approach in the case of an Organization employee's Project and task respective scheduling.

Keywords— Gauge, Scheduling, System, Efficiency, Employee, Employer, Autonomous, Adaboost, GradientBoostingRegressor, RandomForestRegressor, Support Vector Regression, DecisionTreeRegressor

I. INTRODUCTION

In this Corona pandemic situation, most work-from-home employees face unrecognized issues for their work, and employers are deceived by some employees [1]. Especially in Sri Lanka most of the employees prefer to work from home because of the economic critical issue. Working from home is not like the office environment. Employers can't identify how employees work in their homes [1]. Employers can't identify the efficiency and skills of the employees [2] [3]. It is difficult to manage the schedule of the employees for their respective skills [2] [3].

The employee management system is very user-friendly and it is very helpful to employees and employers. This system can identify the employee's efficiency in the tasks. Then we can be able to assign the tasks at which time the employees can work efficiently. This organization or the company can earn many profits. The employee can also be able to work easily. Our system has two sub-systems; One is the Gauge system and the other is the Scheduler system. Our system asks the employees how much they work on every task.

Every employee will have specific skills in their respective modules. Every organization wants to know that their employees have work skills. So, our system proposed a schedule management system with the help of gauging efficiency.

Workforce planning is one of the most important activities in many organizations [2] [4]. It is related to the allocation of personnel to tasks and has a direct effect on operational efficiencies such as productivity of day-to-day operations, labor cost, and quality of service provided. Labor costs often account for a large portion of total operating costs, especially in companies in service industries. Reducing this cost by just a few percent can have huge implications for businesses [3] [2]. Effective workforce planning is a central aspect of running an organization smoothly and cost-effectively.

Workforce scheduling has drawn a lot of interest from academics and businesses over the years due to its complexity and high practical importance [4] [5]. The majority of the literature now available on workforce scheduling takes into account comparatively stable work contexts, including allocating personnel to a set of shifts at a manufacturing facility or hospital [6]. However, there are several situations where employees must perform duties in dispersed locations. We need software that benefits both employees and employers in these situations since we are unable to determine the job performed by the employee [5] [6] [4]

Among other things, these include workers with varying work schedules, places where a workday begins and ends, and individuals with various talents and credentials. Since they are pertinent in the majority of firms, several of these elements, including varying working hours and employee expertise, are well included in the standard workforce planning literature.

The study's hypothetical situation involves an on-site service provider with a freelancing workforce, where workers set their hours. The business operates in a very dynamic environment, and every day, services are provided at various customer locations. Random customer requests come in throughout time, and judgments about scheduling are made online without the knowledge of upcoming work.

Customers can change the way that resources are scheduled by choosing the best time window and resource for their needs, during which the resource will be accessible. As a result, the organization's planning

system must take into account significant dynamic fluctuations in both client expectations and resource availability. However, there hasn't been much study done in this planning setting.

This study aims to examine the formulation and resolution of a time-sensitive online workforce scheduling and routing problem. Using data sets given by the case firm and contrasted with the company's present planning system, the proposed solution method will be assessed. Maintaining realistic situations that may be employed in the actual world will receive a lot of attention.

II.   LITERATURE REVIEW

Most employers didn't know how much work was done by the employees [3]. Employers didn't know the skills of the employees and it was hard to identify the skills for the respective tasks. Employers can't identify the time in which employees work with maximum efficiency. These are the problems that continue in the corona pandemic situation on work from home. So, now our system provides solutions for identifying the works and the respective skills.

From some research papers, we can get to know that the workforce management system currently in use is not fully automated and does not prescribe the tenure of employees and an employer in the scheduling process [3] [2]. When an employer selects the desired service and gives the required information about the job, the system scans the active database of staff and filters out those who are not available in the region or who do not have the necessary qualifications to perform the service. The system evaluates the available employee's schedule and suggests the first available time that fits the new job in the schedule.

At present, all services have the same service timings and service timings are hereby corrected. Firms without mandatory working hours create employees, which implies Employees have time preferences regarding the workday, which results in diverse working hours. The customer has the booking time and preferences because they have the choice of employees, suggested hours, and, or select workers.

Our product will analyze employees' efficiency and assign suitable employees for the respectful task. It is also required to input some data manually. For example, the work done by employees per date; work done by an employee in hour wise. From these records and historical data, our software can analyze the suitable employee for the work and the employer can also get to know the details of each employee. It was the biggest issue in previous times, but we'll try to full fill the requirements.

III.   METHODOLOGY

A. Employee Gauge Prediction

The employee Gauge prediction component is responsible for predicting the factors that help measure the employee's efficiency. Historical working records of employees and their profile data such as skills, word k experience is considered the most efficient working hour for the employee and mother t efficient type of task attributes such as suitable task complexity, task skills, etc.

Working time and employee prediction models were created using the "Developers and programming languages" dataset, which was taken from the "Top 980 Starred Open Source Projects on GitHub." 17.000 github.com developers and 1.400 programming languages and technologies are included in the dataset [7]. Every user is a row, and every technology is a column. Each number falls inside the [0, 1] range and represents the relative importance of this ability to the user. The frequency the label occurs disrespects the other labels, for this user, writing his user and forked repositories, is what matters, not the expertise level of other users [8].

The model is trained using the multiclass classification algorithm to predict the most appropriate working time for each employee in the organization. The model utilizes past data which were records of previously completed projects belonging to each employee for training the process. Employee ID, Code quality of the projects each employee works on, level of each project, project duration, and completion time by the employee to complete each task is fed as input features that are used to train the machine learning model for predicting the best working time for each employee. The model determines that each employee's best working period is the class of period in which they it the highest score in the model The Adaboost classification is used to train the model and achieved 95% of accuracy. After the training is done the trained model is deployed in a central server where the classification of efficiency of employees is determined for time durations.

B. Employee prediction to assign the task

This is a categorical regression analysis to determine the most appropriate employees for upcoming projects in the organization. In this case, we have utilized the top ten skills employees have, the types of projects they perform previously, and project duration data for training the machine learning algorithm. The model can predict the most appropriate employee id(employee) to work on each project when it comes to the technologies required and the project duration.

The "Developers and programming languages" dataset, which is derived from the "Top 980 Starred Open Source Projects on GitHub" was used to generate both working time prediction and employee prediction, models [9]. A dataset containing Each user is a row, and each column is a technology. Each value is in the range [0,1], meaning this skill's relative weight for this user. It is not the level of experience between users, but the frequency the label appears to respect all the other labels, for this

user, when parsing his user and forked repositories. So, we can simply measure the valuable employee who can most suitable for the task.

In the employee prediction model, we have applied four regression algorithms: GradientBoostingRegressor, RandomForestRegressor, Support Vector Regression, and DecisionTreeRegressor to predict the employee by using user-input multiple feature sets [10]. Here, we have taken three evaluation metrics: model accuracy (R2 score), mean absolute error and mean squared error for evaluating each model and selecting a more appropriate algorithm to train our data. According to those evaluation metrics of each algorithm, we selected the GradientBoostingRegressor to train our data to get a prediction outcome. Gradient boosting Regression calculates the difference between the current prediction and the known correct target value.

IV. RESULTS AND DISCUSSION

A. Employee Gauge Prediction

To forecast the output class of labels, the Adaboost classification model is employed. The Adaboost approach is used for model training because it outperformed other fundamental machine learning multiclass algorithms such as Decision tree classification, K-Nearest Neighbor (KNN) algorithm, and SVM classification algorithms in terms of accuracy [11] [12] [13].

The Adaboost algorithm is used to gain the advantage of increasing the model's accuracy. Furthermore, the Adaboost model has a variety of options for fine-tuning the characteristics of our training model [13] [14]. Since the input parameters are not collectively optimized, the chances of Adaboost overshooting are minimal [13]. Adaboost has increased the accuracy of weak classifiers as well compared to other approaches. Table 1 represents the evaluation metrics of the Adaboost model. Figure 1 represents the confusion matrix of the trained model.

Class	Metrics		
	Precision	*Recall*	*F1Score*
00.00-06.00	0.97	0.96	0.96
00.00-12.00	0.93	0.97	0.95
06.00-12.00	0.94	0.96	0.95
06.00-18.00	0.98	0.86	0.92
12.00-18.00	0.92	0.91	0.92
18.00-00.00	0.97	0.97	0.97
Model accuracy			0.95

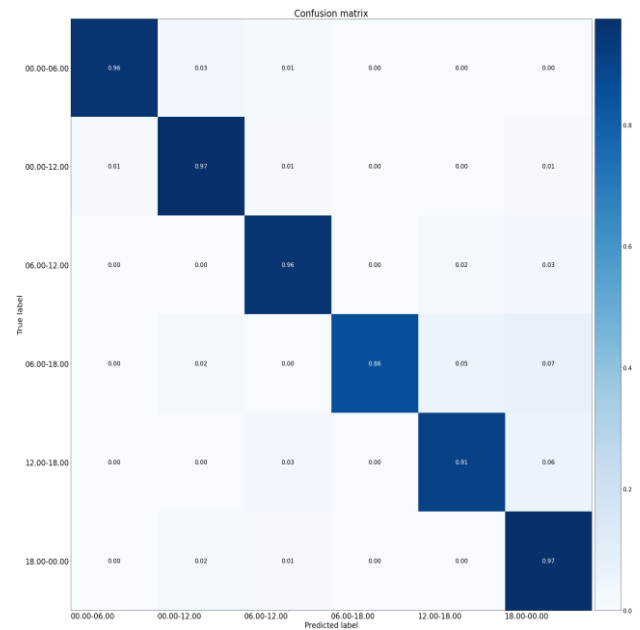**Table 1:** Accuracy of every models

Figure 1: True label and confusion matrix

B. Employee prediction to assign the task

For the tasks, we need to allocate the employees for the working time. So, we tested four regression algorithms: GradientBoostingRegressor, RandomForestRegressor, Support Vector Regression, and DecisionTreeRegressor to predict the employee by using user-input multiple feature sets. GradientBoostingRegressor has increased the accuracy of regression as well compared to other algorithms.

The algorithm determines that each employee's best match to the task that they can work on according to the projects. The GradientBoostingRegressor algorithm is used to train the method and achieved 90% of accuracy.

Method	r2_score	Mean Absolute Error (MAE)
GradientBoostingRegressor	0.90	2.12
RandomForestRegressor	0.78	6.31
SVR	0.82	5.50
DecisionTreeRegressor	0.70	10.28

Table 2: r2_scoure and MAE of used methods

V. CONCLUSION

This Management system is software to identify the respective value of employees, rating the skills, source value for each skill, categorize employee performance, module, skills rating selection, analysis what suitable time, most performance time from

historical data, list of the period, project task, task attribute, task complexity, required skills, level of skills, period of task finish, rating, and arrange the comparing skills set, experience, task complicity, performance prediction, the project contains, time management. This organization can put the employee to the respective project and task by this system. They can also identify whether the employee is working hard or not, and they can assign the employees by the schedule from time management. Our system can identify the skills of the employees and gauge the employee's skills and performance by the respective tasks. So, employers can assign the best-predicted employees for the task of real-time scheduling.

Our system can improve in the future to identify the task in monitoring them by monitoring the screen. Tasks are monitored by the AI system to identify the respective tasks of the employees.

REFERENCES

[1] pwc, [Online]. Available: https://www.pwc.com/us/en/library/covid-19/coronavirus-technology-impact.html.

[2] viewsonic, [Online]. Available: https://www.viewsonic.com/library/business/10-challenges-of-working-from-home-for-employers-and-how-to-solve-them/.

[3] pcmag, [Online]. Available: https://www.pcmag.com/how-to/get-organized-20-tips-for-working-from-home.

[4] sinnaps, "sinapps," [Online]. Available: https://www.sinnaps.com/en/project-management-blog/real-time-scheduling.

[5] J. Y.-T. Leung, "link.springer," [Online]. Available: https://link.springer.com/chapter/10.1007/978-1-4615-3956-8_2.

[6] K. Ramamritham and J. Stankovic. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/259426 .

[7] "kaggle," [Online]. Available: https://www.kaggle.com/datasets/jaimevalero/developers-and-programming-languages?select=user-languages.csv.

[8] "coralpublications," [Online]. Available: https://coralpublications.org/index.php/jemi/article/view/56.

[9] "kaggle," [Online]. Available: https://www.kaggle.com/datasets/chasewillden/topstarredopensourceprojects.

[10] "RandomForestRegressor," [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html.

[11] "k nearest neighbor," [Online]. Available: https://realpython.com/knn-python/.

[12] "SVM," [Online]. Available: https://scikit-learn.org/stable/modules/svm.html.

[13] RuihuWang, "sciencedirect," [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1875389212005767.

[14] P. W. &. H. Zhao, "link.springer," [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-642-18129-0_1.