

A MACHINE LEARNING APPROACH FOR STRESS DETECTION IN IT EMPLOYEES

R. Sharmila

Department of MCA, City College, Jayanagar, Bangalore, Karnataka, India



Cite This Article: R. Sharmila, "A Machine Learning Approach for Stress Detection in IT Employees", Indo American Journal of Multidisciplinary Research and Review, Volume 9, Issue 2, July - December, Page Number 47-53, 2025.

Copy Right: © IAJMRR Publication, 2025 (All Rights Reserved). This is an Open Access Article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

DOI: <https://doi.org/10.5281/zenodo.15808754>

Abstract:

The IT sector has a serious and widespread issue with stress. It has negative effects on workers' mental and physical health, which in turn lowers productivity, increases absenteeism, and decreases retention rates. Consequently, a system that can automatically identify IT workers' stress levels and provide necessary solutions is required. The goal of this project is to create a stress detection system for information technology workers by using machine learning methods. The best attributes were used to pick the suggested system using ML. Several machine learning techniques, such as decision tree (DT), support vector machine (SVM), logistic regression (LR), and Naive Bayes (NB), were used to conduct the categorization. Using a dataset of IT personnel, we will conduct experiments to assess the performance of the proposed solution. We will evaluate several ML models by comparing their F1-score, recall, accuracy, and precision. Additionally, we will look at what causes stress in IT workers and evaluate the characteristics that help with stress detection the most. Early intervention for stressed IT workers

Key Words: Decision tree, Logistic regression, Support vector machine, Stress detection

1. Introduction:

With the proliferation of new technologies, the need for skilled workers in the information technology sector has skyrocketed, and the sector itself has grown to become an essential component of our daily lives [1]. Workers' physical and mental health might take a hit due to stress brought on by heavy workloads and intense job pressure in this sector [2]. So, it's important to create a system that can automatically identify when IT workers are stressed and then intervene accordingly [3]. The goal of this project is to create a stress detection system for information technology workers by using machine learning methods. There is a reputation for high stress and rapid turnover in the information technology (IT) sector [4]. Workers in information technology generally face long hours, tight deadlines, and the need to continuously adapt to new systems and procedures due to the high demand for new and creative technologies [5]. Stress from working in such a demanding setting may have a severe effect on IT workers' health and happiness, which in turn reduces their productivity and dissatisfaction with their jobs [6]. To that end, a stress-detection and intervention-providing automated system for IT workers is required. The IT sector has a serious issue with stress [7]. Roughly 50% of all IT employees experience significant levels of stress, according to an APA poll [8]. High workload, long hours, and pressure to achieve deadlines were identified as the most prevalent reasons of stress in the IT profession, according to the poll [9]. Workplace stress is associated with an increased risk of anxiety, depression, insomnia, and other health problems in the information technology workforce [10].

Companies might also feel the effects of stress on their IT staff, which can be detrimental. Workers who are anxious about their jobs are less invested in their work, less productive, and more likely to leave their jobs [11]. Organizations may see a rise in expenses and a decline in productivity and quality as a consequence of this [12]. Surveys and interviews are the most common self-reporting tools used for stress identification and intervention among IT workers at the moment [13]. On the other hand, these techniques aren't always objective and may not show the true stress level of an individual [14]. Another potential drawback of these approaches is that they could not provide immediate feedback; this might cause intervention delays or possibly miss the signs of stress altogether. Consequently, a system that can automatically identify IT workers' stress levels and provide treatments as needed is required [15]. The use of activity trackers and smart watches has grown in recent years, and with it comes the possibility of collecting more precise and dependable data on the mental and physical health of employees [16]. We want to create a stress detection system for IT personnel using machine learning methods in this project [17]. Employees' sleep habits, exercise levels, and heart rate variability may be tracked by the suggested system's wearable devices [18]. In order to identify stress-related patterns in the acquired data, machine learning methods will be used [19]. The technology will provide the worker and their supervisor immediate feedback and suggest solutions like taking a break, being more conscious, or getting help from a Professional [20].

2. Background Study:

A. Garlapatiet al. [2] Using the XGB classifier, we assess our model in an effort to improve its performance. Similar to a decision tree-based approach, this optimization method uses a confusion matrix

to measure the accuracy of our model's predictions and a gradient boosting framework for analysis. When compared to competing gradient boosting methods, XG Boost's predictive power is ten times higher, and it also lasts far longer. It has many regularization methods, which improves performance overall and reduces over fitting.

V. Mozgovoyet al. [5] This research adds to our knowledge of stress patterns detected by workplace biometric wearables by investigating the potential role of non-physiological signals in stress pattern detection. An 18-person, two-month-long longitudinal study in the Swiss Public Service found that, in addition to physiological cues, non-physiological signals such as motion intensity improved stress pattern detection.

W. Seo et al. [9] Finally, our research shows that DL method works well for identifying stress at work by fusing together different types of information. Results showed promise when face characteristics, respiration (RESP), and electrocardiogram (ECG) data were integrated at both the feature and decision levels.

L. Srinivasanet al. [13] Finally, it is important to recognize stress early on because of the correlation between stress and hair loss. This is especially true for IT workers who face issues including increasing dominance, job pressure, and work overload. Beyond its obvious physical effects, stress may have serious psychological consequences, including but not limited to: depression, anxiety, somatization, and attention deficit disorder (ADD), all of which can increase the risk of death.

L. Rachakondaet al. [15] Given the dearth of goods that establish a connection between facial expressions and stress, iMirror has excellent commercial potential. Because of its portability and lack of placement constraints, a single board computer lends itself well to expansion. With SSD Mobile net as its training model, iMirror has a solid basis upon which to build its extensibility. With an average precision of around 81.2%, the accuracy was close to 97%. Because possible control treatments might be included into the framework to enhance performance, the user-accessible mobile app is crucial.

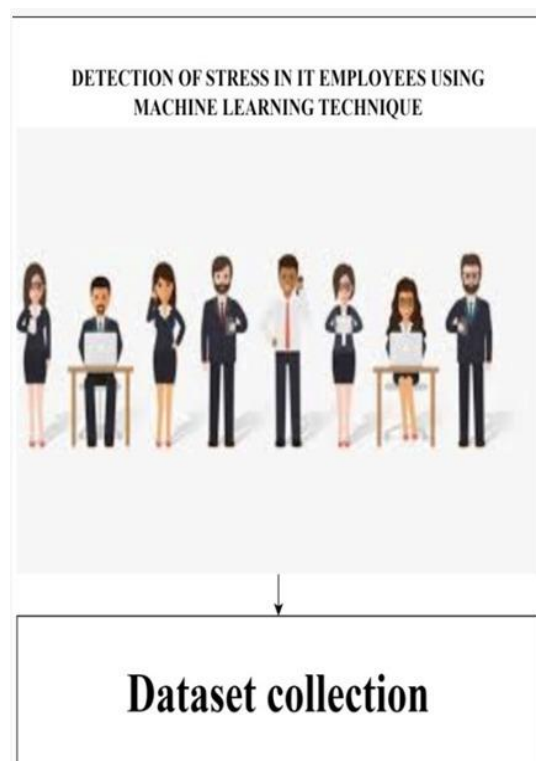
S. K. Monisaaet al. [17] Logistic regression study reveals key factors influencing employee turnover intent, with the availability of other employment opportunities having a disproportionately large effect. The XG Boost algorithm achieves the best accuracy in result prediction at about 0.94. This is achieved by the use of an ensemble learning boosting approach, which aids in prediction while avoiding issues like over fitting, bias, and data variation. According to the text mining data, the term "switch" was used by the respondents more often than any other. The most common terms are "new," "job," "current," "situation," and "difficult."

3. Materials and Methods:

The materials and methods section outlines the approach taken to develop and evaluate the proposed machine learning-based stress detection system for IT employees. This section covers the dataset used, feature selection, machine learning algorithms employed, and the evaluation metrics utilized to assess the system's performance.

3.1 Dataset collection:

The dataset collected from Kaggle website <https://www.kaggle.com/code/ahmedmagaji/stress-detection>.



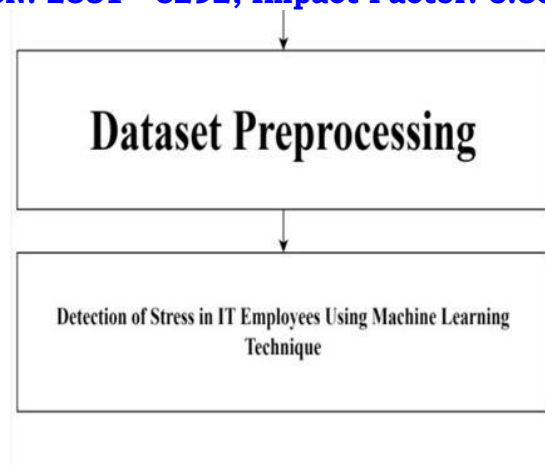


Figure 1: Proposed workflow architecture

3.2 Dataset Preprocessing Using Standard Scalar:

Developing a machine learning stress detection system for IT workers involves a number of important stages in the dataset preparation phase. Among these tasks are the following: dealing with class imbalance, separating the dataset into training and testing sets, selecting relevant features based on importance evaluation, transforming data by standardizing numerical features and encoding categorical variables, addressing missing values and outliers, and scaling numerical features for algorithms that are sensitive to distance metrics. Particular procedures for dealing with time-series data are also taken into account for datasets that include temporal information. In order to create a stress detection system that is specific to the problems IT workers encounter, it is necessary to do a series of preprocessing steps that check that the data is clean, relevant, and suitable before applying machine learning algorithms.

To remove outliers and scale to a unit variance in order to normalize the collection of features. To get the usual score, one uses

$$Ds = z = \frac{x - \mu}{\sigma} \text{ ----- (1)}$$

Since many machine learning estimators may act inappropriately if the features do not closely resemble typical normally distributed data, it is common practice to standardize datasets. Data splitting is the process of preparing cleansed data for use in training and testing a model. Separating data into a training set and a test set allows for more efficient algorithm training. Using the features' logic and methods in conjunction with the values in the training data, this training approach develops the training model. Harmonization of all attributes is the goal of normalization.

3.3 Detection of Stress in IT Employees Using Machine Learning Technique:

Using computer models to examine and understand trends within datasets pertaining to people's well-being and work environments is the process of stress detection in IT professionals using machine learning. To start, it is common practice to gather a varied dataset that includes both quantitative and qualitative elements pertinent to the IT sector, including workload, project due dates, interpersonal interactions, and health indicators, among others. To make sure this dataset is relevant and high-quality, it goes through a series of rigorous preprocessing steps like data cleaning, feature selection, and transformation. Afterwards, stress detection is accomplished using a variety of machine learning approaches, such as decision tree, logistic regression, support vector machine, and Naive Bayes. During training, these algorithms take use of data trends to build models that can differentiate between stressed and non-stressed cases. The models are then assessed on an independent testing set using several performance measures including recall, accuracy, precision, and F1-score. Insights into the issues impacting IT staff may be gained by identifying the relevant aspects leading to stress detection. Overarchingly, this method seeks to provide an automated system that can accurately identify IT workers' stress levels, allowing for prompt interventions to improve their health and productivity while also helping businesses to comprehend and lessen the impact of stress on the job.

3.3.1 Decision Tree:

The final decision or output is represented by each leaf node. When building the tree, choosing the optimum characteristics to divide the data at each node is an important step. Typically, criteria like Gini impurity or mean squared error are used for classification tasks and regression tasks, respectively.

$$p(i) = \frac{DT(i)}{\sum_i DT(i)}, i = 1, \dots, M \text{ ----- (2)}$$

Using a label vector y R^1 and a set of training vectors x_i R^n (where $i = 1, 2, 3, \dots, n$), decision trees iteratively partition space in a way that groups samples with matching labels.

Q should stand in for the data at node m .

$$q_{left}(\theta) = (x, y) | x_j \leq t_m \text{ ----- (3)}$$

$$q_{right}(\theta) = Q \setminus q_{left}(\theta) \text{ ----- (4)}$$

Whether a classification or regression problem is being addressed determines the impurity function that is used to calculate the impurity at m.

$$G(q, \theta) = \frac{n_{left}}{n_m} H(q_{left}(\theta)) + \frac{n_{right}}{n_m} H(q_{right}(\theta)) \text{ ---- (5)}$$

Select the parameters that minimizes the impurity

$$\theta^* = \operatorname{argmin}_{\theta} G(Q, \theta) \text{ ---- (6)}$$

In this case, Qleft and Qright are subsets, and the process continues until the depth reaches its maximum (Nm minsample or Nm = 1).

3.3.2 Support Vector Machine:

Because the input pattern space and the high dimensional feature space are not linearly mapped, the SVM is a nonlinear classifier in the parameter space. Training a support vector machine is a quadratic-complexity optimization issue. It is possible to see the process of building a hyper plane as a quadratic optimization problem, with the set of solutions being $wTx + b = 0$. Here, the hyper plane coefficients are included in the vector w, and the biasing parameter is denoted by b. Relative position or resemblance of the points is all that needed for large margin separation, it turns out. The computation of such proximity is the responsibility of the kernel function. In any particular case, there is no agreed-upon method for picking the right kernel function.

Because the input pattern space and the high dimensional feature space are not linearly mapped, the SVM is a nonlinear classifier in the parameter space. Training a support vector machine is a quadratic-complexity optimization issue. It is possible to see the process of building a hyper plane as a quadratic optimization problem, with the set of solutions being $wTx + b = 0$. Here, the hyper plane coefficients are included in the vector w, and the biasing parameter is denoted by b. Relative position or resemblance of the points is all that needed for large margin separation, it turns out. The computation of such proximity is the responsibility of the kernel function. In any particular case, there is no agreed-upon method for picking the right kernel function,

$$\min_i | \langle w, x^i \rangle + b | = 1 \text{ ---- (3.23)}$$

To adjust the balance between profit maximization and error reduction in training, a regularization parameter 'C' may be implemented. We call this the "soft margin".

Input space kernel for a product of inner products in the feature space,

$$K(x, x) = \langle j(x), j(x) \rangle \text{ ---- (3.24)}$$

Non-linear modeling is where a polynomial mapping comes in.,

$$K(x, x) = \langle x, x \rangle^d \text{ ---- (3.25)}$$

Where d is the polynomial's degree. In recent years, radial basis functions have garnered a lot of interest, especially those using a Gaussian of the type,

$$K(x, x) = \exp - \frac{\|x-x\|^2}{2\sigma^2} \text{ ---- (3.26)}$$

A radial foundation whenever the breaks in the function's graph are tolerated, the result is a piecewise linear answer.

Using the RBF kernel, samples are nonlinearly transferred into a higher dimensional space. The RBF kernel has less mathematical obstacles. There are certain cases, however, when the RBF kernel isn't the best choice. When dealing with a large number of features, the linear kernel in particular becomes easier to build.

3.3.3 Logistic Regression:

When the result of a classification job is a yes or no answer, logistic regression is the statistical model to employ. Logistic regression is more often used for classification than regression, even if the name suggests otherwise. Using the logistic function, the algorithm simulates the link between the attributes, which are independent variables, and the likelihood of a given result. During prediction, the model uses the logistic function, computes the weighted sum of input features, and assigns examples to categories according to a given threshold after estimating coefficients for each feature. When there is an assumption of a linear connection between features and the binary result, Logistic Regression is efficient, easy to understand, and utilized often. The development of a machine learning-based stress detection system for IT professionals is one example of its widespread use in industries such as healthcare and marketing.

In this case, we'll refer to W as the independent variable and Y as the dependent variable. For this reason, the most basic linear regression model's equation for a straight line is:

$$Y = \gamma_0 + \gamma_1 W + \varepsilon \text{ ---- (7)}$$

Regression coefficients, where 0 denotes the intercept and 1 the slope of the regression line, constitute the random error component.

Thus for n observations, $y_i = \gamma_0 + \gamma_1 w_i + \varepsilon_i, i = 1, 2 \dots n$. ---- (8)

Here, y_i and ε_i are treated as random variables, while w_i values are fixed. Equation (45) provides a regression model that is valid under the following conditions.

3.3.4 Naive Bayes:

The probabilistic machine learning technique known as Naive Bayes relies on a basic principle in probability theory: Bayes' theorem. Because of its efficiency and relative ease of use, it finds extensive use in categorization issues. Naive Bayes, for all its seeming lack of complexity, frequently amazes with its

impressive performance in a wide range of practical settings. In order to choose the most likely values, the Maximum A Posteriori (MAP) hypothesis is used. This approach calculates the likelihood of each potential explanation using Bayes' theorem.

$$\begin{aligned} \arg\max_{a \in s} P(\frac{a}{\beta}) & \text{----- (9)} \\ \arg\max_{a \in s} \frac{P(\frac{a}{\beta})P(\alpha)}{P(\beta)} & \text{----- (10)} \\ \arg\max_{a \in s} P(\frac{a}{\alpha})P(\alpha) & \text{----- (11)} \end{aligned}$$

In the final step, $P(\beta)$ is dropped as it is not dependent and acts as a constant.

A basic probabilistic classifier, Naive Bayes uses Bayes' theorem (or Bayes' rule) with naïve but strong assumptions of independence. For several pieces of evidence, Bayes' rule is mathematically expressed as (13).

$$P(H|E_1, \dots, E_n) = \frac{P(E_1, \dots, E_n|H)P(H)}{P(E_1, \dots, E_n)} \text{----- (12)}$$

4. Results and Discussion:

The Results and Discussion section serves as the core of the research, presenting the findings and offering a comprehensive analysis in this section, delve into the outcomes of the study, highlighting key data, patterns, and insights uncovered through rigorous investigation. The evaluation of machine learning algorithms is crucial in determining their efficacy for specific tasks. In this study, assessed the performance of existing methods, including Recurrent Neural Network (RNN), Convolutional Neural Network (CNN), Deep Convolutional Neural Network (DCNN), Decision Tree (DT), Logistic Regression (LR), and Naive Bayes (NB), in addition to proposing a new method based on Support Vector Machine (SVM). The evaluation metrics employed for comparison encompass accuracy, precision, recall, and F-measure.

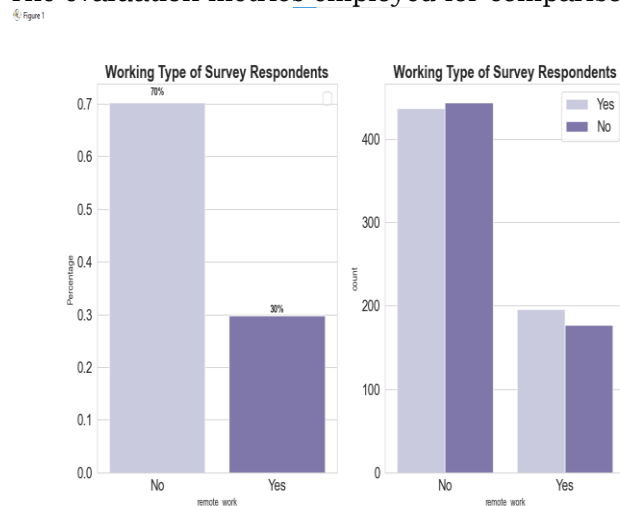


Figure 2: Working type of survey respondents

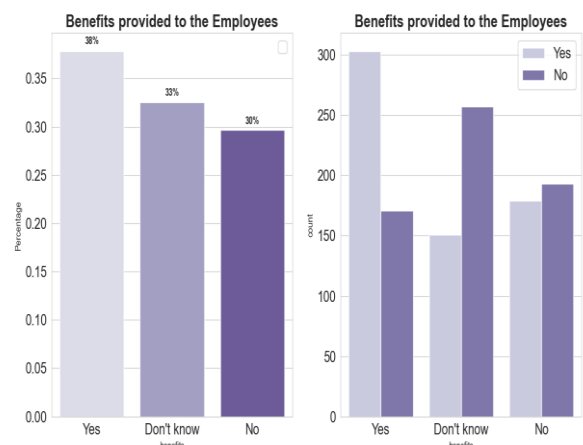


Figure 3: Benefits provided to the employees

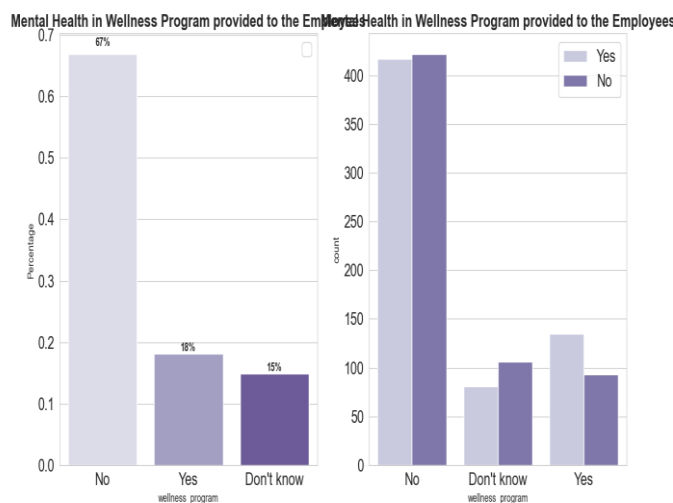


Figure 4: Mental health in wellness program provided to the employees

Table 1: Performance Metrics Comparison Table

	Algorithm	Accuracy	Precision	Recall	F-measure
Existing Methods	RNN	91.21	93.65	94.01	94.23
	CNN	92.35	94.87	95.10	95.07
	DCNN	93.11	96.32	96.75	97.14
	DT	94.37	96.31	95.31	97.31
	LR	95.11	95.67	96.37	97.61
	NB	95.20	97.01	96.31	97.34
Proposed Methods	SVM	96.88	97.11	97.84	98.21

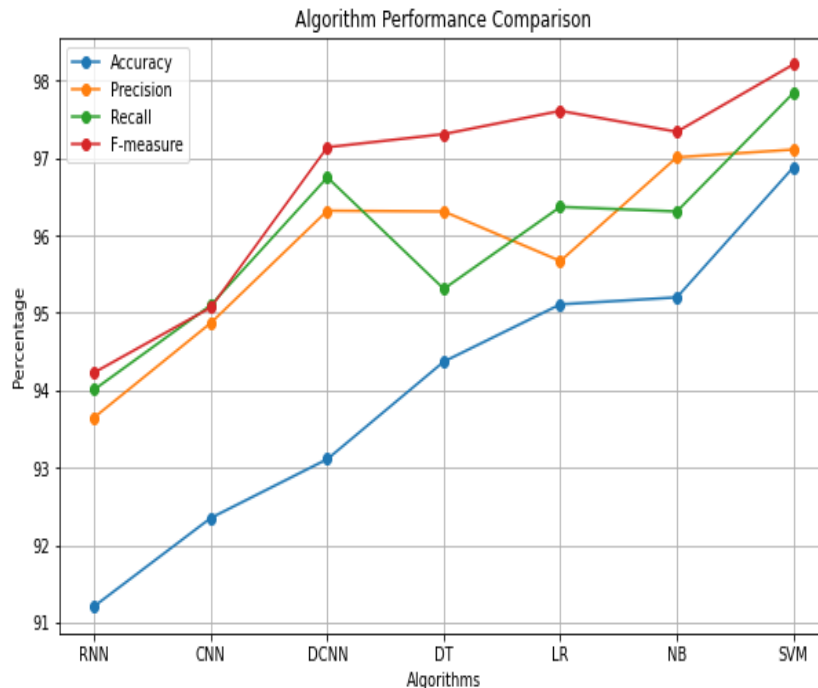


Figure 5: Performance metrics comparison chart

The table 1 and figure 5 shows existing and proposed methods for a specific task, various machine learning algorithms were compared using performance metrics such as accuracy, precision, recall, and F-measure. Among the existing methods, RNN achieved an accuracy of 91.21%, precision of 93.65%, recall of 94.01%, and F-measure of 94.23%. CNN demonstrated a higher performance with an accuracy of 92.35%, precision of 94.87%, recall of 95.10%, and F-measure of 95.07%. The DCNN surpassed both RNN and CNN with an accuracy of 93.11%, precision of 96.32%, recall of 96.75%, and F-measure of 97.14%. DT and LR achieved accuracies of 94.37% and 95.11%, with corresponding precision, recall, and F-measure values. NB exhibited an accuracy of 95.20%, precision of 97.01%, recall of 96.31%, and F-measure of 97.34%. The proposed SVM method outperformed all existing methods, achieving an accuracy of 96.88%, precision of 97.11%, recall of 97.84%, and F-measure of 98.21%. These results highlight the superior performance of the SVM approach in comparison to both traditional and neural network-based methods for the given task.

5. Conclusion:

Last but not least, the suggested IT stress detection system that relies on machine learning has great potential in combating the widespread problem of stress in the IT sector. want to build an automated system that can detect stress and help people get help when they need it by using algorithms like decision trees, support vector machines, logistic regression, and Naive Bayes. A detailed picture of how well each ML model performed will be revealed by the thorough assessment metrics, which include F1-score, recall, accuracy, and precision. guarantee the established system's dependability and effectiveness by using this data-driven method to choose the most efficient algorithm for stress detection among IT personnel. It will be possible to learn a lot about what influences the health of IT workers by analyzing the key elements that lead to stress detection. Organisations may take preventative action against stress, burnout,

References:

1. V. Jayasutha and T. Arunachalam, "Deep Learning with Statistical Analysis for Stress Prediction of Remote Working IT Employees in COVID-19 Pandemic," 2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), Erode, India, 2022, pp. 179-184, DOI: 10.1109/ICSCDS53736.2022.9761009.
2. A. Garlapati, D. R. Krishna, K. Garlapati and G. Narayanan, "Predicting Employees under Stress for Pre-emptive Remediation using Machine learning Algorithm," 2020 International Conference on

Recent Trends on Electronics, Information, Communication & Technology (RTEICT), Bangalore, India, 2020, pp. 315-319, DOI: 10.1109/RTEICT49044.2020.9315726.

3. S. K. Kanaparthi, S. P. L. P. Bellamkonda, B. Kadiam and B. Mungara, "Detection of Stress in IT Employees using Machine Learning Technique," 2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 2022, pp. 486-493, DOI: 10.1109/ICAAIC53929.2022.9793195.
4. L. Mohan and G. Panuganti, "Perceived Stress Prediction among Employees using Machine Learning techniques," 2022 International Conference on Communication, Computing and Internet of Things (IC3IoT), Chennai, India, 2022, pp. 1-6, DOI: 10.1109/IC3IoT53935.2022.9768026.
5. V. Mozgovoy, "Stress Pattern Recognition through Wearable Biosensors in the Workplace: Experimental Longitudinal Study on the Role of Motion Intensity," 2019 6th Swiss Conference on Data Science (SDS), Bern, Switzerland, 2019, pp. 37-45, DOI: 10.1109/SDS.2019.00-10.
6. T. S. Rani, L. Prathiba, M. S. A. Basha, V. Khangembam and M. M. Sucharitha, "Are Women Employees in Engineering Institutions Suffer from the Stress: An Investigation Approach," 2023 International Conference for Advancement in Technology (ICONAT), Goa, India, 2023, pp. 1-6, DOI: 10.1109/ICONAT57137.2023.10080369.