Abstract

Machine Learning is a branch of artificial intelligence that leverages data to imitate the pattern by which a human brain learns, in the process improving accuracy. Statistical principles are used to process data and learn from it efficiently. Machine Learning algorithms are extensively used to predict future outcomes based on past experiences that were recorded in a meticulous manner. These past experiences or data can help us to deduce preliminary insights about the data and what it represents. The current paper discusses certain Machine Learning algorithms where the prime objective is to classify the inputs into one of the two categories. The dataset in focus is the employee attrition dataset that gives various insights regarding the presumable reasons behind an employee leaving the job. The factors such as accuracy, precision score, recall score and f1\_score for Random Forrest, XgBoost, Adaboost, Gradient boosting and Decision Tree Classifier have been ascertained and compared. Furthermore, Hyperparameter tuning, using the ‘RandomSearchCV’ python library is also implemented on the better performing algorithms, with the goal of achieving better performance.

**Keywords**: Machine Learning, Random Forrest, XgBoost, Adaboost, Decision Tree, Gradient Boost, Hyperparameters, RandomSearchCV.

Introduction

Employee attrition is when an employee of an organisation, leaves the company due to various reasons. It may be because his or her performance was not up to the mark. Or due to incompatibility with their colleagues and more so on. If the attrition happens with the will of the organisation than it is supposed to be fine. But if an organisation loses its valuable employee because the employee wanted it so, then it is a problem. The organisation might have gone through a long process of hiring and training of that employee. Both time and money of the organisation is invested on a particular employee, and when this employee leaves than it is a matter of great concern for the employer. The organisation will have to do all that investments of time and resource again to hire another employee to replace the one that left. This problem can be analysed and to a great extent be solved by using machine learning algorithms. To do such a task, the most preliminary tool required to get started is the past data of the employees who left the company along with the ones who stayed. Machine learning models learn from the past data, or we might say, past experiences to predict the future, so if we feed this data consisting of employees that stayed and the employees that left, we might get a model that takes the employee data as input and tells if the employee is going to leave or not. Further analysis of this model will give us insights of what are the plausible factors leading to attrition of employees. This will help the management to take steps in the right direction.

Previous Work and their short comings

Machine Learning

The procedure of using computational methods to learn information directly from the data. This learning should be done without relying too much on a predetermined equation. This whole process is called Machine learning. The data samples used to train the model is very important. When the data models increase, the accuracy of this model might also increase. Since the process of Machine learning largely involves a lot of statistical computations, the data used to train such a model needs to be clean and relevant to the intended goals of the project. The data used should be consistent, that is, it should not have too many null values and understandingly the tabulated data should be correct. The relevance of the data used is important because if the data is not relevant than the model will predict wrongly for real world cases. For example, if we need a model to predict a disease in an individual, we need the medical data of that person rather than the data about his TV watching habits. The TV habits data can be useful for the model where subscription to a TV plan is predicted. There are many machine learning algorithms that use different statistical computation techniques to train a model. We have used some of these models during this research.

Hyperparameter Tuning

Machine Learning models are mathematical models associated with several parameters. These parameters are to be learned by the same machine learning model. These parameters are learnt by the model by training the model with the existing data. This process is called fitting the data to the model. There is another kind of parameters that cannot be directly learned by a model through the regular training process, known as Hyperparameters. These parameters are fixed even before the actual training process begins. Significant properties of a model is expressed by such parameters. The complexity or how fast a model must learn is also ascertained by Hyperparameters. In practice, there are two main strategies that are used to implement Hyperparameter tuning, Randomized Search and Grid search. For grid search technique, all the possible hyperparameters including the intermediate combinations of the hyperparameters are tried, that is, for each hyperparameter a new model is created. This is the reason why grid search is more computationally expensive. The Randomized search Tuning strategy is better than the “GridSearch” strategy because it tries only a fixed number of hyperparameters for the model. For this research, the “RandomizedSearchCV” python module is used.

Decision Tree

In each decision tree, the internal nodes stand for the features of the data set, branches for decision rules and leaf nodes for outcomes. All in all, Decision tree classifier is a machine learning algorithm that gives a model that can classify using a tree data structure. It is easy to implement and comprehend because tree structures are easy to understand. A tree structure also mirrors a human being’s decision-making process, that is, the process of choosing one of the options at each step until the goal is reached or final decision is made. A decision tree starts with a question and based on the possible outcomes; it splits the tree at every level. The main issue in the decision tree algorithm is to choose the best attributes for the root node and the sub nodes. To do this task, we use the technique called Attribute selection measure or ASM. With ASM we can easily select the best possible attribute for the nodes of the tree.

Random Forest

Random Forest contains several decision trees associated with various subsets of the dataset. To improve the accuracy, average is taken at the end. Random Forest is used for classification and regression problems and is based on the concept of ensemble learning. Ensemble learning is the process of combining multiple classifiers to solve a difficult or complex problem and to improve the all over accuracy of the solution. This is obvious that a greater number of trees in the forest will lead to better accuracy but also take a toll on performance. Random Forest is useful because it takes less time to train, predicts the output with high accuracy and can maintain accuracy when a large proportion of data is missing.

Diagram

Description automatically generated

XG-Boost

XgBoost or Extreme Gradient Boosting was proposed by researchers at University of Washington. Its library is written in C++ which optimizes the Gradient boost training process. As the name suggests, XgBoost attempts to boost the Gradient Boosting model. Boosting is an ensemble modelling technique where several weak classification models are combined to form a stronger classification model. This processes simply adds weak models in series. At the beginning a model is created on the data and then a second model is added in series. The second model attempts to correct the errors made by the first model. This process is repeated until the complete data set is predicted or the maximum number of models is added. For XgBoost, decision trees are created in sequential form. Weights are important part of this process. The independent variables that are fed to the decision tree which predicts results are assigned with weights first. If an independent variable is predicted wrongly, the weight of that variable is increased and then fed to the next decision tree. These individual trees or classifiers are then ensembled together to obtain a more precise and stronger model.

Proposed approach and analysis

Results and discussions

Conclusion

references