PREDICTING STUDENT PERFORMANCE USING PERSONALIZED ANALYTICS

[1]B.SONA,[2] S.T.DEEPA
[1]MPHIL RESEARCH SCHOLAR DEPARTMENT OF COMPUTER SCIENCE
SHRI SHANKARLAL SUNDARBAI SHASUN JAIN COLLEGE FOR WOMEN, CHENNAI, TAMILNADU
[2]ASSOCIATE PROFESSOR
DEPARTMENT OF COMPUTER SCIENCE,
SHRI SHANKARLAL SUNDARBAI SHASUN JAIN COLLEGE FOR WOMEN, CHENNAI, TAMILNADU.

ABSTRACT

In recent years the number of knowledge keep in educational information is growing apace. The keep information contains hidden data that if used aids improvement of student's performance and behavior. During this paper prophetical modeling approach is employed for extracting this hidden information. Student behavior record analyzation by using traditional approach is not so convenient hence for personal analyzation of student going with hadoop with different techniques as per convenience. The prophetical models can facilitate the instructor to know however well or however poorly the scholars in his/her category can perform, and hence the teacher will select proper educational and tutorial interventions to reinforce student learning outcomes.

Keyword: smart education, algorithm

INTRODUCTION

An educational institution needs to have an approximate prior knowledge of enrolled students to predict their performance in future academics. This helps them to identify promising students and also provides them an opportunity to pay attention to and improve those who would probably get lower grades. There is no absolute scale for measuring knowledge but examination score is one scale which shows the performance indicator of students. Quality education is one of the most promising responsibilities of any country to his countrymen. Quality education does not mean high level of knowledge produced. But it means that education is produced to students in efficient manner so that they learn without any problem. For this purpose quality education includes features like: methodology of teaching, continuous evaluation, categorization of student into similar type, so that students have similar objectives, educational background etc.

LITERATURE SURVEY:

Massive Open Online Courses (MOOCs) are an increasingly pervasive newcomer to the virtual landscape of higher-education, delivering a wide variety of topics in science, engineering, and the humanities. However, while technological innovation is enabling unprecedented open access to high quality educational material, these systems generally inherit similar homework, exams, and instructional resources to that of their classroom counterparts and currently lack an underlying model with which to talk about learning. In this paper we will show how existing learner modeling techniques based on Bayesian Knowledge Tracing can be adapted to the inaugural course, 6.002x: circuit design, on the edX MOOC platform. We identify three distinct challenges to modeling MOOC data and provide predictive evaluations of the respective modeling approach to each challenge. The challenges identified are; lack of an explicit knowledge component model, allowance for un penalized multiple problem attempts, and multiple pathways through the system that allow for learning influences outside of the current assessment. **:** The accurate estimation of students' grades in future courses is important as it can inform the selection of next term's courses and create personalized degree pathways to facilitate successful and timely graduation. This paper presents future course grade predictions methods based on sparse linear and low-rank matrix factorization models that are specific to each course or student–course tuple. These methods identify the predictive subsets of prior courses on a course-by-course basis and better address problems associated with the *not-missing-at-random* nature of the student–course historical grade data. The methods were evaluated on a dataset obtained from the University of Minnesota, for two different departments with different characteristics. This evaluation showed that focusing on course-specific data improves the accuracy of grade prediction.

Educational data mining is an emerging discipline, concerned with developing methods for exploring the unique types of data that come from the educational context. This work is a survey of the specific application of data mining in learning management systems and a case study tutorial with the Moodle system. Our objective is to introduce it both theoretically and practically to all users interested in this new research area, and in particular to online instructors and e-learning administrators. We describe the full process for mining e-learning data step by step as well as how to apply the main data mining techniques used, such as statistics, visualization, classification, clustering and association rule mining of Moodle data. We have used free data mining tools so that any user can immediately begin to apply data mining without having to purchase a commercial tool or program a specific personalized tool. The factorization machine (FM), a general-purpose matrix factorization (MF) algorithm suitable for this task, is leveraged as the state-of-the-art method and compared to a variety of other methods. Our experiments show that FMs achieve the lowest prediction error. Results for both cold-start and non-cold-start prediction demonstrate that FMs can be used to accurately predict in both settings. Finally, we identify limitations observed in FMs and the other models tested and discuss directions for future work. To our knowledge, this is the first study that applies state-of-the-art collaborative filtering algorithms to solve the next-term student grade prediction problem. Based on a comparative analysis of the combination of interaction features, our best CRF model can achieve a precision of 0.581, recall of 0.660 and a weighted F-score of 0.560, outweighing several baseline discriminative classifiers applied at each sequence position. These findings have implications for initiating early instructor intervention, so as to engage students along less active interaction dimensions that could be associated with low grades.

EXISTING SYSTEM:

Existing concept deals with providing backend by using mysql which contains lot of drawbacks i.e data limitation is that processing time is high when the data is huge and once data is lost we cannot recover so thus we proposing concept by using Hadoop tool.

DRAWBACKS

We can process limitation of data.

We get results with take more time and maintenance cost is very high.

PROPOSED SYSTEM:

Propsed concept deals with providing database by using hadoop tool we can analyze no limitation of data and simple add number of machines to the cluster and we get results with less time, high throughput and maintence cost is very less and we are using joins, partition’s and bucketing techniques in hadoop.

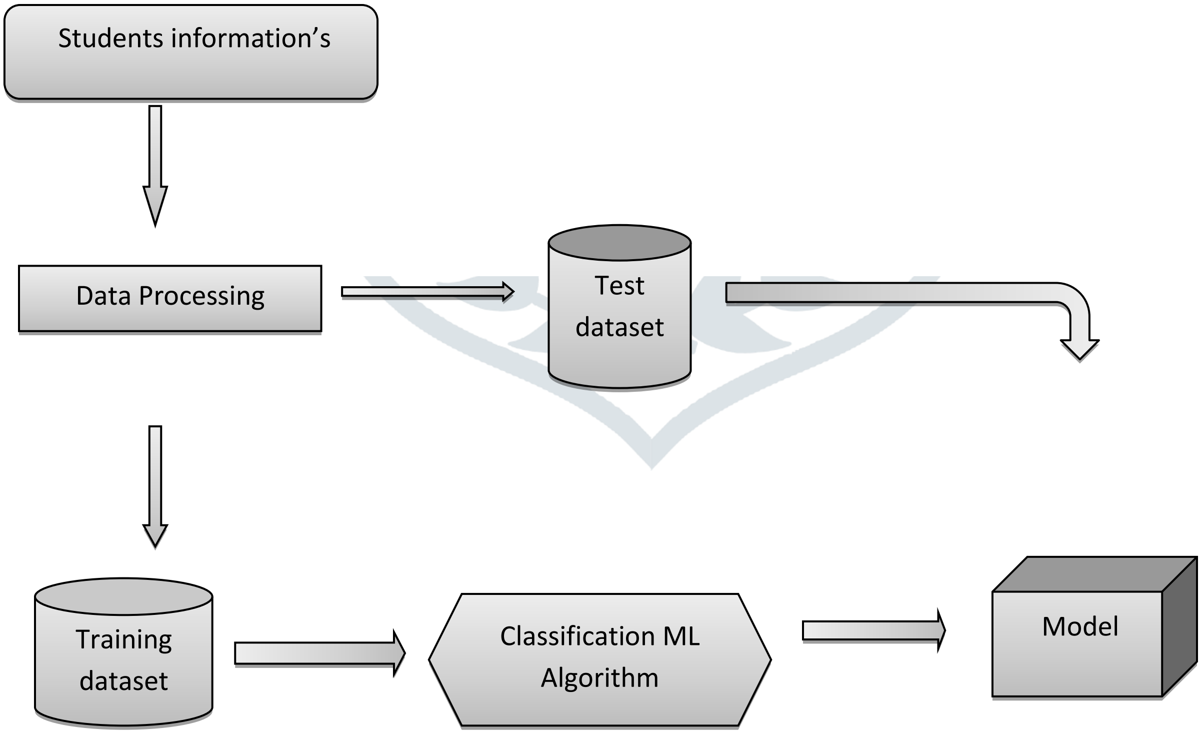
ADVANTAGES

No data loss problem

Efficient data processing.

PROPOSED SYSTEM

Following are the steps performed to determine the breast cancer stage of a patient



Architecture of proposed system

SOFTWARE DESCRIPTION

Anaconda is a free and open-source distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management and deployment. Package versions are managed by the package management system "Conda". The Anaconda distribution is used by over 12 million users and includes more than 1400 popular data-science packages suitable for Windows, Linux, and MacOS. So, Anaconda distribution comes with more than 1,400 packages as well as the Conda package and virtual environment manager called Anaconda Navigator and it eliminates the need to learn to install each library independently. The open source packages can be individually installed from the Anaconda repository with the `conda install` command or using the `pip install` command that is installed with Anaconda. Pip packages provide many of the features of conda packages and in most cases they can work together. Custom packages can be made using the `conda build` command, and can be shared with others by uploading them to Anaconda Cloud, PyPI or other repositories. The default installation of Anaconda2 includes Python 2.7 and Anaconda3 includes Python 3.7. However, you can create new environments that include any version of Python packaged with conda.

Anaconda Navigator

Anaconda Navigator is a desktop graphical user interface (GUI) included in Anaconda distribution that allows users to launch applications and manage conda packages, environments and channels without using command-line commands. Navigator can search for packages on Anaconda Cloud or in a local Anaconda Repository, install them in an environment, run the packages and update them. It is available for Windows, macOS and Linux.

The following applications are available by default in Navigator:

- JupyterLab
- Jupyter Notebook
- QtConsole
- Spyder
- Glueviz
- Orange
- Rstudio
- Visual Studio Code

METHODOLIGY

Dataset

The data set collected for predicting passengers is split into Training set and Test set. Generally, 7:3 ratios are applied to split the Training set and Test set. The Data Model which was created using Random Forest, logistic, Decision tree algorithms, K-Nearest Neighbor (KNN) and Support vector classifier (SVC) are applied on the Training set and based on the test result accuracy, Test set prediction is done.

Preprocessing

The data which was collected might contain missing values that may lead to inconsistency. To gain better results data need to be preprocessed so as to improve the efficiency of the algorithm. The outliers have to be removed and also variable conversion need to be done. Based on the correlation among attributes it was observed that attributes that are significant individually include property area, education, loan amount, and lastly credit history, which is the strongest among all. Some variables such as applicant income and co-applicant income are not significant alone, which is strange since by intuition it is considered as important. The correlation among attributes can be identified using plot diagram in data visualization process. Data preprocessing is the most time consuming phase of a data mining process. Data cleaning of loan data removed several attributes that has no significance about the behavior of a customer. Data integration, data reduction and data transformation are also to be applicable for loan data. For easy analysis, the data is reduced to some minimum amount of records. Initially the Attributes which are critical to make a loan credibility prediction is identified with information gain as the attribute-evaluator and Ranker as the search-method.

CLASSIFICATION METHODS

Logistic regression algorithm also uses a linear equation with independent predictors to predict a value. The predicted value can be anywhere between negative infinity to positive infinity. We need the output of the algorithm to be classified variable data. Higher accuracy predicting result is logistic regression model by comparing the best accuracy.

Over-fitting is a common problem in machine learning which can occur in most models. K-fold cross-validation can be conducted to verify that the model is not over-fitted. In this method, the data-set is randomly partitioned into *kmutually exclusive* subsets, each approximately equal size and one is kept for testing while others are used for training. This process is iterated throughout the whole k folds.

True Positive Rate(TPR) = TP / (TP + FN)

False Positive rate(FPR) = FP / (FP + TN)

Accuracy: The Proportion of the total number of predictions that is correct otherwise overall how often the model predicts correctly defaulters and non-defaulters.

Algorithm Explanation

In machine learning and statistics, classification is a supervised learning approach in which the computer program learns from the data input given to it and then uses this learning to classify new observation. This data set may simply be bi-class (like identifying whether the person is male or female or that the mail is spam or non-spam) or it may be multi-class too. Some examples of classification problems are: speech recognition, handwriting recognition, bio metric identification, document classification etc. In Supervised Learning, algorithms learn from labeled data. After understanding the data, the algorithm determines which label should be given to new data based on pattern and associating the patterns to the unlabeled new data.

Used Python Packages:

sklearn:

- In python, sklearn is a machine learning package which include a lot of ML algorithms.

- Here, we are using some of its modules like train_test_split, DecisionTreeClassifier or Logistic Regression and accuracy_score.

NumPy:

- It is a numeric python module which provides fast maths functions for calculations.
- It is used to read data in numpy arrays and for manipulation purpose.

Pandas:

- Used to read and write different files.
- Data manipulation can be done easily with data frames.

Matplotlib:

- Data visualization is a useful way to help with identify the patterns from given dataset.
- Data manipulation can be done easily with data frames.

Logistic Regression

It is a statistical method for analysing a data set in which there are one or more independent variables that determine an outcome. The outcome is measured with a dichotomous variable (in which there are only two possible outcomes). The goal of logistic regression is to find the best fitting model to describe the relationship between the dichotomous characteristic of interest (dependent variable = response or outcome variable) and a set of independent (predictor or explanatory) variables. Logistic regression is a Machine Learning classification algorithm that is used to predict the probability of a categorical dependent variable. In logistic regression, the dependent variable is a binary variable that contains data coded as 1 (yes, success, etc.) or 0 (no, failure, etc.).

In other words, the logistic regression model predicts P(Y=1) as a function of X. Logistic regression Assumptions:

- Binary logistic regression requires the dependent variable to be binary.
- For a binary regression, the factor level 1 of the dependent variable should represent the desired outcome.
- Only the meaningful variables should be included.
- The independent variables should be independent of each other. That is, the model should have little.
- The independent variables are linearly related to the log odds.
- Logistic regression requires quite large sample sizes.

Decision Tree

It is one of the most powerful and popular algorithm. Decision-tree algorithm falls under the category of supervised learning algorithms. It works for both continuous as well as categorical output variables. Assumptions of Decision tree:

- At the beginning, we consider the whole training set as the root.
- Attributes are assumed to be categorical for information gain, attributes are assumed to be continuous.
- On the basis of attribute values records are distributed recursively.
- We use statistical methods for ordering attributes as root or internal node.

Decision tree builds classification or regression models in the form of a tree structure. It breaks down a data set into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. A decision node has two or more branches and a leaf node represents a classification or decision. The topmost decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data. Decision tree builds classification or regression models in the form of a tree structure. It utilizes an if-then rule set which is mutually exclusive and exhaustive for classification. The rules are learned sequentially using the training data one at a time. Each time a rule is learned, the tuples covered by the rules are removed.

This process is continued on the training set until meeting a termination condition. It is constructed in a top-down recursive divide-and-conquer manner. All the attributes should be categorical. Otherwise, they should be discretized in advance. Attributes in the top of the tree have more impact towards in the classification and they are identified using the information gain concept. A decision tree can be easily over-fitted generating too many branches and may reflect anomalies due to noise or outliers.

K-Nearest Neighbor (KNN/KNC)

K-Nearest Neighbor is a supervised machine learning algorithm which stores all instances correspond to training data points in n-dimensional space. When an unknown discrete data is received, it analyzes the closest k number of instances saved (nearest neighbors) and returns the most common class as the prediction and for real-valued data it returns the mean of k nearest neighbors. In the distance-weighted nearest neighbor algorithm, it weights the contribution of each of the k neighbors according to their distance using the following query giving greater weight to the closest neighbors.

Usually KNN is robust to noisy data since it is averaging the k-nearest neighbors. The k-nearest-neighbors algorithm is a classification algorithm, and it is supervised: it takes a bunch of labeled points and uses them to learn how to label other points. To label a new point, it looks at the labeled points closest to that new point (those are its nearest neighbors), and has those neighbors vote, so whichever label the most of the neighbors have is the label for the new point (the "k" is the number of neighbors it checks). Makes predictions about the validation set using the entire training set. KNN makes a prediction about a new instance by searching through the entire set to find the k "closest" instances. "Closeness" is determined using a proximity measurement (Euclidean) across all features.

Random Forest

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of over fitting to their training set. Random forest is a type of supervised machine learning algorithm based on ensemble learning. Ensemble learning is a type of learning where you join different types of algorithms or same algorithm multiple times to form a more powerful prediction model. The random forest algorithm combines multiple algorithm of the same type i.e. multiple decision trees, resulting in *a* forest of trees, hence the name "Random Forest". The random forest algorithm can be used for both regression and classification tasks.
The following are the basic steps involved in performing the random forest algorithm:

- Pick N random records from the dataset.
- Build a decision tree based on these N records.
- Choose the number of trees you want in your algorithm and repeat steps 1 and 2.
- In case of a regression problem, for a new record, each tree in the forest predicts a value for Y (output). The final value can be calculated by taking the average of all the values predicted by all the trees in

forest. Or, in case of a classification problem, each tree in the forest predicts the category to which the new record belongs. Finally, the new record is assigned to the category that wins the majority vote.

Support Vector Machines

A classifier that categorizes the data set by setting an optimal hyper plane between data. I chose this classifier as it is incredibly versatile in the number of different kernelling functions that can be applied and this model can yield a high predictability rate. Support Vector Machines are perhaps one of the most popular and talked about machine learning algorithms. They were extremely popular around the time they were developed in the 1990s and continue to be the go-to method for a high-performing algorithm with little tuning.

- How to disentangle the many names used to refer to support vector machines.
- The representation used by SVM when the model is actually stored on disk.
- How a learned SVM model representation can be used to make predictions for new data.
- How to learn an SVM model from training data.
- How to best prepare your data for the SVM algorithm.
- Where you might look to get more information on SVM.

Conclusion

The analytical process started from data cleaning and processing, missing value, exploratory analysis and finally model building and evaluation. The comparison results of public test set are showing various prediction results.

BIBLIOGRAPHY:

[1] N. Thai-Nghe, T. Horváth, and L. Schmidt-Thieme, "Factorization Models for Forecasting Student Performance," Proc. 4th Int'l Conf. Educational Data Mining (EDM 11), 2011, pp. 11–20.

[2] C. Romero, S. Ventura, and E. Garca, "Data Mining in Course Management Systems: Moodle Case Study and Tutorial," Computers & Education, vol. 51, no. 1, 2008, pp. 368–384.

[3] Z. Pardos et al., "Adapting Bayesian Knowledge Tracing to a Massive Open Online Course in edX," Proc. 6th Int'l Conf. Educational Data Mining (EDM 13), 2013; www.educational datamining.org/EDM2013/papers /rn_paper_21.pdf

[4] Ventura, M.J., Franchescetti, D.R., Pennumatsa, P., Graesser, A.C., Jackson, G.T., Hu, X., Cai, Z., & the Tutoring Research Group. (2004). Combining computational models of short essay grading for conceptual physics problems. In J.C. Lester, R.M. Vicari, & F. Paraguacu (Eds.), Intelligent Tutoring Systems 2004 (pp. 423-431). Berlin, Germany: Springer.

[5] .Hofmann, T. (1999). Probabilistic Latent Semantic Indexing. In SIGIR '99: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 50-57). Berkeley, CA, USA.

[6] Ellwein, M.C., & Graue, M.E. (1996). Assessment as a way of knowing children. In C.A. Grant & M.L. Gomez (Eds.), Making schooling multicultural: Campus and classroom. Englewood Cliffs, NJ: Merrill.

[7] Kluger, A.N., & DeNisi, A. (1996). Effects of feedback intervention on performance. Psychological Bulletin, 119(2), 254-284.

[8] Campbell, J.P., DeBlois, P.B., & Oblinger, D.G. (2007). Academic analytics: A new tool for a new era. EDUCAUSE Review, 42(4), 41-57.

[9] .Blei, D., & Lafferty, J. (2009) Visualizing topics with multiword expressions. arXiv:0907.1013

[10] VanLehn, K. (2011). The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. Educational Psychologist, 46(4), 197-221.