Applications of Machine Learning in Education

Gourav Kumar
201507666
gourav.kumar@research.iiit.ac.in

Enna Sachdeva
201532655
sachdeva.enna@research.iiit.ac.in

Abstract

Various research communities are addressing the limitations in the traditional education system by designing a learning system that is capable of adapting to students' learning style rather than making them adapt to the existing education system. The refinement of the education system from the traditional one-size-fits-all to the one that is based on the learning style of each student, determines the performance of the learning process. A personalized learning system must strike a balance between testing the efficacy of every learning action and maximizing the students learning outcomes using observations on the actions. This paper discusses various machine learning techniques to develop models for different education applications, with the aim to improve learning experience of individual student.

Introduction:

Machine learning is being used extensively in various areas for helping organizations enhance decision making and analyzing new patterns and relationships among a large amount of data. One of those key areas is education with the aim of making personalized learning process for the individual student. The applications of machine learning in the field of education focusses on areas that directly impact students, such as the development of a recommender system for recommending courses to students based on their previous experiences and grades; analysis of educational processes like admissions, alumni relations, and course selections; prediction and forecasting of learning and institutional improvement needs,etc, few of which are discussed in the subsequent sections.

1. What are the different issues being addressed by machine learning, in education system ?

The advancement in the field of machine learning has the potential to create a unique learning path for an individual student. Depending on the diversity of backgrounds, life experiences, learning pace, and familiarity with the topic, machine learning based personalized education systems create personalized learning actions, schedules, and tutoring facilities for individual that maximizes his/her learning. It enables to gain insight into student's knowledge and assist them with self-direction, self-assessment, teamwork, etc. A review of the related literature follows that it is used for improving student success and processes directly related to student learning, examines different ways that may assist faculty and staff with improving learning and supporting educational processes. Various applications identified by the research communities that addresses few of the common issues in education system are as follows-

1. Improving student retention and attrition

   Machine learning and data mining techniques have been used as a way to improve student retention efforts. For instance, Lin *et al*. [8] was able to generate predictive models, based on incoming students data, which were able to provide short-term accuracy for predicting which type of students would benefit from student retention programs study. Similarly, Yu *et al*. [12] applied classification trees, multivariate adaptive regression splines (MARS), and neural networks to educational data which resulted in finding transferred hours, residency, and ethnicity as critical elements in retention efforts

2. Track students' knowledge
   Learning analytics build statistical models of students knowledge to provide computerized and personalized feedback to the students.

3. Optimize learning content
   Content analytics organize and optimize content items like assessments, textbook sections, video lecture, etc.

4. Adaptive teaching policy
   Scheduling algorithms that search for an optimal and adaptive teaching policy helps students learn more efficiently. Active learning and experimental design, which adaptively select assessments and other learning resources for each student individually, enhances learning efficiency.

5. Dynamic scheduling
   Matching students needs with the teachers availability.

6. Plagiarism detection
   Grading systems that assess and score student responses to computer assignments, at large scale, either automatically or via peer grading.

2. How a fully personalized adaptive learning environment is formulated for each student?

Adaptive learning provides students with a customized learning experience with the materials, practice activities, and assessments, based on their progress and previous accomplishments. It refers broadly to a learning process where the content taught or the way such content is presented changes, or adapts, based on the responses of the individual student. Three major components of adaptive learning involves-
Content model: model of contents(problem, question, quiz etc.) to be learned.
Learner model:model to estimate student proficiency.
Instruction model: model to present content to student in a personalized fashion based on proficiency, i.e selecting items of optimal difficulty for the student.

2.1. Learners models

Few of the commonly used learners models are as following-

2.1.1 Bayesian Knowledge Tracing(BKT) models

The standard Bayesian Knowledge Tracing (BKT) model, shown in Figure 1, can be described as a Hidden Markov Model (HMM). Bayesian Network models student learning by capturing the dynamic of knowledge, probabilistically. It combines a priori information with evidence from data. For instance, let X denote the attribute set and Y denote the class variable. The classifier learns the posterior probablity $P(Y|X)$ for every combination of X and Y, so a new record can be classified such that the posterior probability is maximal [4]. BKT has been used to determine when a student has mastered a particular KC and is thus no longer asked to answer steps consisting only of mastered KCs.This framework allows manipulation of probability distributions over multi-dimensional spaces in which there are hundreds of variables [9].

2.1.2 Logistic regression based models of growth

This model estimates the mastery level of a particular skill, given a set of responses. This is obtained by maximizing the conditional probability.

2.1.3 Neural Network based model

The above described models are unable to handle multi-concept items. Chaplot *et al.* [3] proposed an adaptive learning system architecure, based on Artificial Neural Network, to handle multi-concept items. The proposed architecture handles multiconcept problems, i.e they can identify complex nonlinear relationship between the concepts, and systematically selects problems of appropriate difficulty that maximizes learning, as shown in figure 2.
This approach formulated a 'learning gain' of an item as the geometric mean of both the quantities so that it is maximized when chances of correctness and difficulty are balanced.

$$LG(M_j) = \rho\sqrt{sigmoid(b_j) * P(X = 1)} \qquad (1)$$

where,
ρ : a constant
b_j: the difficulty of item $M_j, b_j(,)$
$P(X = 1)$: probability of solving item correctly

Therefore, a student with higher learning rate is given more challenging items and should have higher learning gain than another student having the same skill level but a lower learning rate.

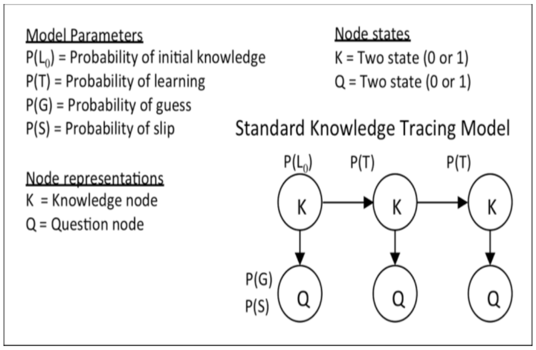

Model Parameters
P(L0) = Probability of initial knowledge
P(T) = Probability of learning
P(G) = Probability of guess
P(S) = Probability of slip
Node states
K = Two state (0 or 1)
Q = Two state (0 or 1)
Standard Knowledge Tracing Model
Node representations
K = Knowledge node
Q = Question node
P(L0)
P(T)
P(T)
K
K
K
P(G)
P(S)
Q
Q
Q

Figure 1. Standard BKT model with parameter and node descriptions

3. What are the different modeling techniques to model educational datasets for different applications?

Student-response modeling involves developing principled statistical models that could (i) accurately predict unobserved student responses to questions and (ii) identify those concepts that govern correct or incorrect responses. In educational systems, data changes quite often, therefore, the main requirements are that the model is robust, not sensitive to small variations in data and should be able to adapt to new students, new teachers, new exercise tasks or updated learning material, which all affect the data distribu-

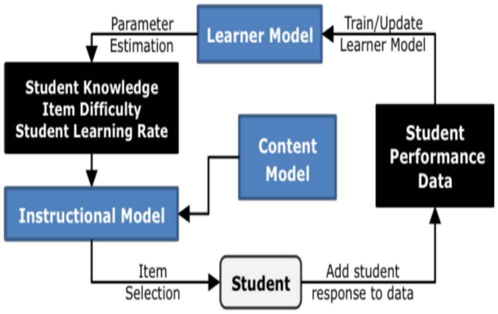


Figure 2. Adaptive Learning System Architecture

tion. In educational system, it is a special requirement that the model is transparent , i.e. the student should understand, how the system models her/him.

Along with it, the main problem in modeling educational data is that the data sets are typically small, sparse, consists of mixed data types, and contain relatively many outliers (exceptionally brilliant and poor students).

Model selection consists of three steps: defining the model family, selecting the model class and selecting the model parameters. The current education systems use mostly predefined ad hoc models, which means that the learners are fit to the model, instead of fitting the model to the learners, which is a serious restriction to adaptivity. Therefore, a dual principle of descriptive modelling $-$ discovering new information in educational data, and predictive modelling $-$ predicting learning outcomes, has been proposed by [5], in which descriptive data produces accurate domain knowledge, which is used as a semantic bias in predictive modelling process and models evolve in an iterative manner, as shown in figure 3. Few of these models reported in literature for the educational dataset are as following-

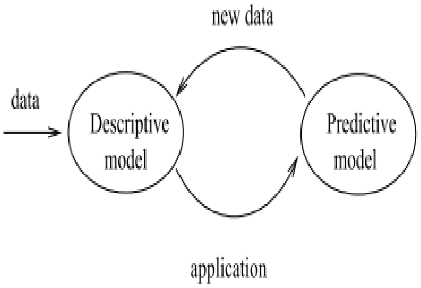


Figure 3. Iterative process of descriptive and predictive modelling. Descriptive modelling reveals the underlying patterns in the data and guides the selection of the most appropriate modelling paradigm and model family for the predictive modelling. When the predictive model is applied in practice, new data is gathered for new descriptive models.

3.1. Probabilistic clustering method

This method produces a probabilistic clustering and learns the model, which best describes the data. This enables to predict clusters for new data points and update the model. Since, the data comes from a mixture of probabilistic models (multivariate distributions), each cluster has a prior probability and its own probability distribution. The whole model is typically represented as a multivariate finite mixture model.

$$f(p) = \sum_{j=1}^{k} \pi_j f_j(p, \theta_j) \tag{2}$$

where, $f_j(p, \theta_j)$ defines the probability that data point p belongs to cluster c_j, with parameters θ_j and π_j.
Low value of $f(p)$ indicates that the point does not fit the model, and can be interpreted it as an outlier.

The advantages of probabilitic clustering method are that each data point p can be represented by its own same or different probability distribution $P(c_1|p), ..., P(c_k|p)$, which allows it to represent the overlapping clusters, where the same data point belongs to several clusters with different probabilities and model outliers with their own distribution. However, probabilistic clustering method is very sensitive to initial parameter settings and often the data is first clustered with another method followed by selecting parameters with Maximum likelihood (ML) principle which maximizes the data (log)likelihood given the model.

Major Application: For courses, study materials recommendations to students.

3.2. Hidden Markov Model

Based on the assumption that contexts cannot be observed directly but are inferred from observations, this technique is quite feasible in learning environments as the contexts often describe unobservable variables like the students skills or motivation. The data can be represented by the Markov Chain, where it is assumed that the current state(context) depends on the previous k sates, i.e.

$$P(C(t)|C(0), ..., C(t1)) = P(C(t)|C(tk), ..., C(t1)) \tag{3}$$

It can be further generalized by allowing dependencies between observable variables, e.g. $O(t)$ depends on k previous observations, $O(t1), ..., O(tk)$, in addition to the current context $C(t)$.

Major Applications: For mastering a skill.

3.3. Bayesian network model

Bayesian network represents the statistical dependencies(as edges) between variables(as vertices) as a directed acyclic graph structure. However, learning a globally optimal Bayesian network structure is generally an NP-hard problem and in practice, the learning algorithms find only a local optimum. A brief description has been discussed in section 3.1.1.

Major Applications: To predict dropout, mastering a skill, dropout, .

4. How the grades are predicted in the courses, students will enrol into? And which models have been used that produces less prediction errors?

The ability to predict student grades in future enrollment terms provides valuable information to aid students, advisors, and educators in achieving the mutually beneficial goal of increased student retention. This information can be used to help students choose the most suitable majors, properly blend courses of varying difficulty in a semesters schedule, and indicate to advisors and educators when students need additional assistance.

To predict the grades of the students for the next term enrollment, Sweeney *et al.* [11] formulates the problem by using a database of (student, course) dyads (i.e. combination of two vectors) with associated content features for the course, student, and course instructor.

Given a database for n students and m courses, the problem is formulated as an $n \times m$ sparse grade matrix G, where $\{G_{ij} \in R | 0 \leq G_{ij} \leq 4\}$ is the grade, i student earned in course j. The A-F letter grades have nominal equivalents in the range 0-4, so the target space is actually discrete. The problem is casted as a regression problem rather than the classification, as classification methods fail to capture the ordinal nature of the data.

The students are characterized by the demographics data, such as age, race, gender, GPA, SAT scores, the declared major and the grade earned in previous term and cumulative GPA. Each cell is considered to be a (student, course) dyad and is represented as a feature vector $X_{ij} \in R^{1 \times p}$. The model is trained on all feature vectors X_{ij} and predict grades $\hat{G_{ij}}$ for all features. For each dyad, the courses are characterized by its aggregate GPA over the terms the course has been offered in the past and the number of students enrolled in the course.

The following regression models, incorporating content features have been explored-

4.1. Random Forest (RF)

This algorithm combines a group of random decision trees, each of which is constructed by discovering the most informative questions that split all samples into groups with similar target attribute values. For regression, the most informative questions are those that produce leaf nodes whose mean squared error (MSE) are minimal among all possible splits and tree construction stops once an additional split would not reduce MSE. Since this usually overfits, an early termination criterion is often specified. Once built, the tree can be used for regression of new data samples. A sample is run through the decision making sequence defined by the structure of the tree until reaching a leaf. Then the prediction is the mean of the grades of the samples at that node.

4.2. Stochastic Gradient Descent (SGD) Regression

The SGD regression method learns a least squares linear regression fit under an L1 regularization. The least squares fit minimizes the squared difference between actual and predicted grades, while the L1 regularization penalty encourages feature sparsity. In particular, unimportant parameters have a tendency to be pushed towards 0, so the L1 penalty operates as a kind of online feature selection. SGD is a gradient-based optimization technique that updates the model parameters incrementally, rather than on the entire training set at once. This reduces overfitting and improves training time, significantly.

4.3. k-Nearest Neighbors

The k-Nearest Neighbors (kNN) algorithm, can be used for regression by finding the k most similar neighbors among all dyads in the training set, by a pairwise distance metric. For instance, to predict a grade for a new dyad (i, j), the Euclidean distance from (i, j) to every dyad in the training set is computed. The k dyads (i', j') with the smallest distance are selected and placed into a set of neighbors $N_{i,j}$. The grade for student i in course j is then predicted as the uniformly weighted average of these neighbors grades.

4.4. Personalized Multi Linear Regression

The original model predicts a missing grade $\hat{G_{ij}}$ for the student i in course j using:

$\hat{G_{i,j}} = s_i + c_j + \sum_{l=1}^{k} P_{il} \sum_{f=1}^{p} W_{lf} X_{ijf}$
$= s_i + c_j + P_i W X_{ij}$

where

s_i is a bias term for student i,
c_j is a bias term for course j
P_i is the $1 \times k$ vector of model weights for student i
W is the $k \times p$ matrix of regression coefficients
X_{ij} is the feature vector generated by student i taking course j.

The model is learned using the following objective function with the Root Mean Squared Error (RMSE) as the loss function $L()$, P, and W are regularized using the squared

Frobenius norm, and all parameters are constrained to be non-negative.

$\hat{G_{i,j}} = w_0 + s_i + c_j + P_i W X_{ij}$
minimize $L(P, W, s, c) + (\lambda_W(||P||_F^2 + ||W||_F^2) + (\lambda_B(||s||_F^2 + ||c||_F^2)$

where,
w_0 is a global intercept term
λ_B is a regularization on the bias.
However, factorization machine(FM), random forests (RF) and the personalized multi linear regression model and a hybrid(FM-RF) method have been reported to achieve the lowest prediction errors. there are 30,754 students declared in one of 144 majors, each of which belongs to one of 13 colleges.

5. How ML techniques are being applied in predicting the possibility of a student to dropout from the univerity?

Students drop out continues to be a major concern to the education community as the attriting students lose time and effort and institutions have no alternative to recover the resources, they devoted to those students. *Data:* Lovenoor Aulck *et al*. [2] models the student dropout using the largest known dataset on higher education attrition, which tracks over 32,500 students' demographics and transcripts records. Some missing SAT and ACT scores in the data were modelled using a linear regression model with other demographic data and pre-college entry data.

*Feature Mapping:*Each department in which students took classes was mapped across four features for each student: a binary variable indicating whether the student took a class in that department, a count of the number of credits taken in that department by the student, a count of the number of classes taken in that department by the student, and the grade point average (GPA) of the student for all graded classes taken in that department.

*Experiments:*To understand the elements which are the best predictor of dropouts, k separate logistic regressions of dropout were made to run on the kth feature, followed by using a regularized linear regression to predict the number of terms in which each non completing student enrolled in before dropping out. To tune the regularization parameters(e.g. the regularization strength for logistic regression, the number of neighbors in kNN, and the depth of the tree in random forests), 10-fold cross-validation was used on 70% of the randomly sampled data and the performance has been reported on the remaining 30% of the data. 3 machine learning models- regularized logistic regression, k-nearest neighbors, and random forests, were used to predict the binary dropout variable and the obtained results were compared using receiver operating characteristic (ROC), as shown in the figure 4 and table 1.

Model	**Accuracy**
Logistic Regression	66.59
Random Forest	62.24
K-nearest neighbors	64.60

Table 1. Accuracies of models

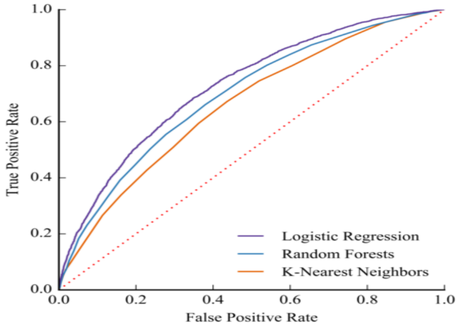


Figure 4. ROC curves model

This methodology with regularized logistic regression was able to give promising results with the prediction of not only dropouts, but also the timings of dropouts as well as the correlation among those predicted students.

6. How the standard computerized adaptive testing(CATs) sorts the difficulty of the questions in the exam and how do we quantify these algorithms?

6.1. Introduction

As the amount of knowledge is increasing rapidly, an ability which can effectively understand and well-organize the knowledge, namely creativity, has been widely required. In order to support such creative human in learning, it is very important to develop effective testing methods which correctly estimate examinees proficiency and then provide detailed learning information based on it. However, the classical testing methods such as paper-based testing are inadequate in the modern testing environment because the classical estimation methods not only need time-consuming testing task but also are often troubled with inaccuracy problem due to guess or error effects and inconsistent responses. To overcome this weakness Computerized Adaptive Testing (CAT) was proposed.

In order to predict the difficulty of a question, there are various methods available, some are analytical and others are learning based. Here in this paper, we will discuss the learning method. We will also discuss a method(Rasch Model) to maximize the information gain and propose a method to include both techniques to improve the information gain about the student.

6.2. Difficulty in Prediction using Learned model

Multiple-Choice Questions(MCQs) are commonly used in many standardized tests, due to its ease of collecting and marking the answers.
CAT(Computer Adaptive Technique) is a technology that is motivated to offer a better assessment by being adaptive to the testee, where the computer administers subsequent questions depending on the precedent performance of the testee. IRT(Item Response Theory) provides the theoretical background on most of the current CAT systems, in which a testees ability (or latent trait) and the difficulty of a question are described in values in a common unit called logit. In IRT, the difference in the ability and difficulty is projected to the probability of the users getting the right answer, using a sigmoid function.
IRT is a well-researched area, where many positive results have been reported.IRT-based CAT system achieves sufficient precision with only 20 questions, whereas pen-and-paper testing requires 100 questions.But for adapting the model a cost of conducting pre-test on a comparable group of testees. In cases where pre-testing is not possible, all questions are assumed to be of equal difficulty at the onset, then the difficulty of the questions is updated as the users responses accumulates.

6.2.1 Difficulty in Prediction

The technique of supervised machine learning was used for the task of difficulty prediction.First, the learning algorithms were explored, and then the best performing classifier was trained using the question data that are annotated with the correct response rate. Then, with a simple binary search-like method based on the predicted difficulty values, a CAT system was built and was tried out by human subjects. As this is one of the earliest attempts in applying machine learning methods to such a task,a simple binary classification was set out.Regression was not employed because it is expected to be unworthy to predict the correct response rate that is observed from subject groups since such observation usually contains what is called measurement error in the literature of psychometrics. A simple binary classifier was trained with the labels easy or difficult, letting the computer to try to grasp a rough notion of difficulty.

*Training Data:*The training data set is obtained from a series of TOEIC preparation books (a total of 702 questions). Each question is annotated with the correct response rates, ranging from 0.0 to 98.5. The figures are based on the tests in4 a TOEIC preparation school in Japan and reportedly based on the results of about 300 testees.All questions consist of a stem sentence of 20-30 words and four alternatives. Seemingly, all questions are intended to be of the same difficulty.The top 305 easiest questions were labeled as easy and the top 305 difficult questions as difficult, based on the correct response rates, leaving out 8% around the average value.

Feature Selection for classification: For classification, the data points(here questions) are needed to be projected in a feature space which is relevant to the application.For this task the following three features were used :

1) Sentence feature: The sentence feature represents the sentence length, which is a measure of number of words in the question sentence.
2) Answer feature: The answer features provide information on the right answer, consisting of blank length, which is the number of words in the correct answer, and an array of binary features on the POS (Part Of Speech) of the right answer.
3)Distractor similarity feature: The distractor similarity features are obtained from the analysis using the technique of modified edit distance, which have been used to extract a lowest-cost conversion path from one string to another. This indicates how effective are the choices in creating confusion in the minds of testees.

Learning Algorithm: After conducting the 10-fold cross validation to compare the performance of different learning algorithms SVM turned out to be best outperforming others.

6.3. Rasch Model

This is an analytical method that is used to rank questions based on their ability to provide information about student's capability. It makes two major assumptions regarding the prior information available.
Firstly: it assumes given a set of questions we know the difficulty parameter $\mu_1, ..., \mu_Q$ associated with it.
Secondly: It assumes the ability parameter $a_1, ..., a_N$ of each student appearing in the test is known beforehand.

Using the item difficulty parameter μ_i and the student ability parameter a_j , the response $Y_{i,j}$ is characterized as a Bernoulli random variable satisfying:

$$P(Y_{i,j} = 1|\mu_i, a_j) = 1/1 + \exp(-(a_j - \mu_j)) \quad (4)$$

According to Rasch model the question which is more uncertain(i.e. having higher entropy) is also more informative.Hence to calculate the entropy of an item i can be calculate by:

$$S_i = \sum_{j-1}^{N} H(Y_{i,j}; \mu_i, a_j) \quad (5)$$

$$= \sum_{j-1}^{N} [\log(1 + \exp(a_j - \mu_i) - \frac{a_j - \mu_i}{1 + \exp-(a_j - \mu_i)})] \quad (6)$$

Hence we rank the items from their highest entropy to lowest entropy.

6.4. Our Proposal

We propose a CAT with recently emerging AQG technology (Automatic Question Generation, explained in the following section) can be used with more efficiency and there can be more information gain about the testees which will make the selection process more precise.
We propose to use the learned SVM classifier to classify the question set generated into 'easy' and 'difficult' and then apply the above mentioned Rasch Model in each class separately to give the ranking to each question in their respective classes.
This will render possible a novel assessment system that adaptively administers questions from the automatically generated question set.This method will not only help in adaptive generation of question according to testees' ability but it will also give an opportunity to the tester to choose questions from a generated set according to their level of informativeness.

7. Using machine learning for detecting plagiarism in a source code

Source code plagiarism is a severe problem in academia. In academia programming assignments are used to evaluate students in programming courses. Therefore checking programming assignments for plagiarism is essential. If a course consists of a large number of students, it is impractical to check each assignment by a human inspector. Therefore it is essential to have automated tools in order to assist detection of plagiarism in programming assignments.

7.1. Implementation and Training

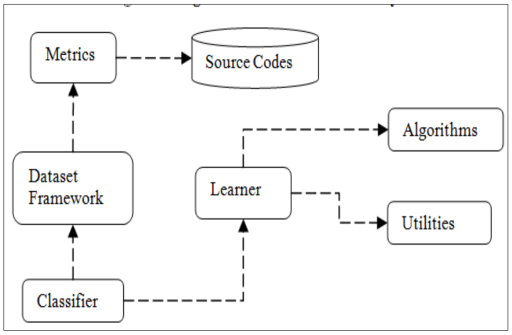


Figure 5. High level Architecture of the system

It is seen that not all source code metrics contribute equally for source code author identification.Therefore the following nine metrics were identified which perform well in source code identification:
1)Line Length Calculator(LLC)
2)Line Words Calculator(LWC)
3)Access Calculator(AC)
4)Comments Frequency Calculator(CFC)
5)Indentifiers Length Calculator(ILC)
6)InLine Space InlineTab Calculator(INT)
7)Trail Tab Space Calculator(TTS)
8)Underscores Calculator(USC)
9)Indent Space Tab Calculator(IST)

In order to extract some of the metrics, it was essential to parse source codes files according to the syntactic rules of the programming language, which was used to write that source code hence we can represent each source code file as a set of tokens together with token frequencies. Those tokens and token frequencies are used as inputs for our learning algorithms. This process is almost identical to the use of word and word frequencies in document classification problems.

*Training Dataset:*same dataset as used by Lange and Mancoridis [7].This dataset consists of Java Source code files belonging to 10 developers. Each subsystem in figure 5is mapped into a Java package except, source code sub-system. It represents the Java source codes, which are used for training and testing the system.Metrics sub-system generates source code metrics from Java source code files.The above model was trained using multinomial naive bayes classifier, AdaBoost learning as well as using kNN learning algorithm.It was seen that each of these methods complemented each other.
One limitation that was found out was

8. Probabilistic author-topic models for information discovery helps in conducting literature survey for a research.

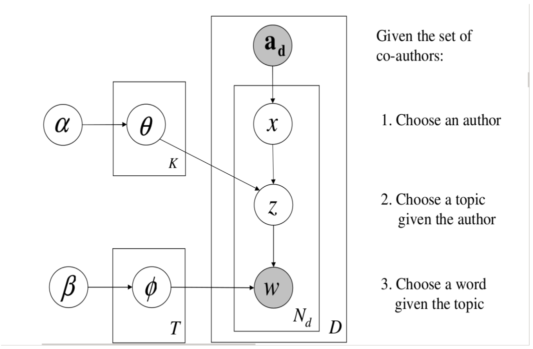
With the advent of the Web and various specialized digital libraries, the automatic extraction of useful information from text has become an increasingly important research area in data mining.

8.1. Introduction

Developing a technique which can help in finding out literature according to the author or the topic of choice can save a lot of time and energy.The author-topic models can be used to support a variety of interactive and exploratory queries on the set of documents and authors, including analysis of topic trends over time, finding the authors who are most likely to write on a given topic, and finding the most unusual paper written by a given author.This can help the research community in an unprecedented way. Bayesian unsupervised learning is used to fit the model to a document collection.

We model documents as if they were generated by a two-stage stochastic process. Each author is represented by a probability distribution over topics, and each topic is represented as a probability distribution over words for that topic. The words in a multi-author paper are assumed to be the result of a mixture of each authors topic mixture. The topic-word and author-topic distributions are learned from data in an unsupervised manner using a Markov chain Monte Carlo algorithm.

8.2. Probabilistic Generative Model



$\mathbf{a}_d$
Given the set of co-authors:
α
θ
K
x
1. Choose an author
z
2. Choose a topic given the author
β
ϕ
w
3. Choose a word given the topic
T
N_d
D

Figure 6. The graphical model for author topic model

To simplify the representation of documents, we use a bag of words assumption that reduces each document to a vector of counts, where each vector element corresponds to the number of times a term appears in the document. As shown in figure 6 Each author (from a set of K authors) is associated with a multinomial distribution over topics, represented by θ. Each topic is associated with a multinomial distribution over words, represented by ϕ. The multinomial distributions θ and ϕ have a symmetric Dirichlet prior with hyperparameters α and β

For each word in the document, we sample an author x uniformly from the set of co-authors, then sample a topic z from the multinomial distribution θ associated with author x and sample a word w from a multinomial topic distribution ϕ associated with topic z. This sampling process is repeated N times to form document d.

8.3. Estimating the Model Parameters through Bayesian Estimation

The author-topic model includes two sets of unknown parametersthe K author-topic distributions θ, and the T topic distributions ϕ as well as the latent variables corresponding to the assignments of individual words to topics z and authors x. The Expectation-Maximization (EM) algorithm is a standard technique for estimating parameters in models with latent variables, finding a mode of the posterior distribution over parameters. However, when applied to probabilistic topic models this approach is susceptible to local maxima and computationally inefficient.We pursue an alternative parameter estimation strategy Instead of estimating the model parameters directly, we evaluate the posterior distribution on just x and z and then use the results to infer θ and ϕ.For each word, the topic and author assignment are sampled from:

$$\begin{aligned} &P(z_i = j, x_i = k | w_i = m, z_{-i}, x_{-i}) \\ &\approx \frac{C_{mj}^{WT} + \beta}{\sum\limits_{\hat{m}} C_{\hat{m}j}^{WT} + V\beta} \frac{C_{kj}^{AT} + \alpha}{\sum\limits_{\hat{j}} C_{k\hat{j}}^{AT} + T\alpha} \end{aligned} \tag{7}$$

where z_i = j and x_i = k represent the assignments of the ith word in a document to topic j and author k respec- tively, w_i = m represents the observation that the ith word is the mth word in the lexicon, and z_{-i} , x_{-i} represent all topic and author assignments not including the ith word.Furthermore, C_{mj}^{WT} is the number of times word m is assigned to topic j, not including the current instance, andC_{kj}^{AT} is the number of times author k is assigned to topic j, not including the current instance, and V is the size of the lexicon.

During parameter estimation, the algorithm only needs to keep track of a V T (word by topic) count matrix, and a K T (author by topic) count matrix, both of which can be represented efficiently in sparse format. From these count matrices, we can easily estimate the topic-word distributions and author-topic distributions by:

$$\phi_{mj} = \frac{C_{mj}^{WT} + \beta}{\sum\limits_{\hat{m}} C_{\hat{m}j}^{WT} + V\beta} \tag{8}$$

$$\theta_{kj} = \frac{C^{AT}_{kj} + \alpha}{\sum\limits_{\hat{j}} C^{AT}_{k\hat{j}} + T\alpha} \tag{9}$$

8.4. Applications:

1)*Topic Trends over Time:*These fractions provide interesting and useful indicators of relative topic popularity in the research literature in recent years.The results are quite informative and indicate substantial shifts in research topics within the field of topic.
2)*Topic and Authors for New Documents:*In many applications, we would like to quickly assess the topic and author assignments for new documents not contained in our subset.

9. How machine learning could aid blind students in education.

Machine learning techniques have been applied to automate the process of translating figures from mathematics, science and engineering textbooks to a tactile form suitable for blind students which improve the character recognition accuracy without any need to fine tune parameters to the character finding algorithm.

The Tactile Graphics Assistant (TGA) is a program created at the University of Washington to aid in the tactile image translation process. It separates text from an image so that the text can later be replaced by Braille (a tactile writing system used by blind or visually impaired people) and inserted back onto the image. In order to streamline the text selection process, the TGA employs machine learning to recognize text so that large groups (possibly hundreds) of images can be translated at a time.

9.1. Basic Work Flow

The basic workflow of the tactile graphics translation process is summarized in the subsequent paragraph. The Tactile Graphics Project is aimed at streamlining the tactile image translation process to produce graphics in the most efficient way. This means, with the right tools, images are produced that are inexpensive, quick, and easily customizable. Jayant *et al.* [6] explains this process of automating the translation of figures from mathematics, science, and engineering textbooks to a tactile form suitable for blind students,as below-

The images are scanned and put into a proper file format using Photoshop. These images are further pre-processed with Photoshop and cropped and thresholded. TGA extracts and separates the text from the images that contain only the text labels. This separates text-free images from the images that contain only the text labels. OCR is further applied on the text labeled images. The text less images are resized and Braille text is placed onto these images. The complete work flow is summarized in figure 7.

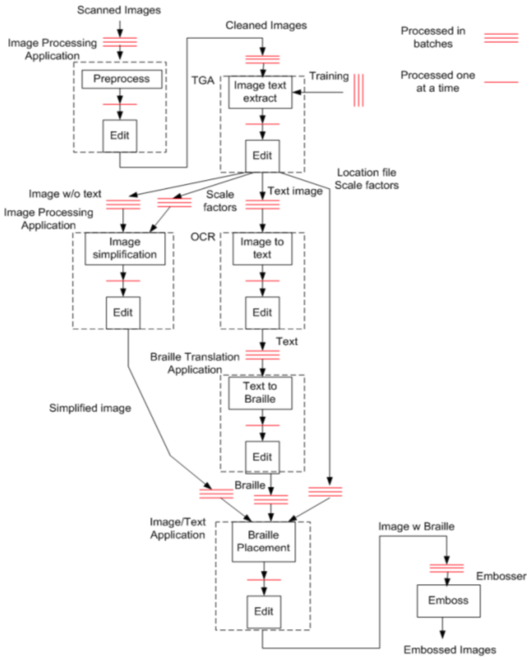


Figure 7. Work Flow

9.2. Classification using SVM

The classification of characters from the original image is done using the support vector machine (SVM). While training, user manually selects characters in the image and training is done on positive and negative examples. Those selected characters are positive examples and all other connected components are negative examples of characters. Features like height, width, area, pixel color are used to classify characters. Labels are manually chosen on the training set by the user, and using a minimum spanning tree algorithm, centroids of all characters in the image are connected. In the TGA, machine learning has been applied to recognize text characters in images, and deal with angled text.

10. Machine learning application in determining the liveliness of an educational video.

Online educational videos have emerged as one of the most popular modes of learning in the recent years. Studies have shown that liveliness is highly correlated to engagement in educational videos.When educational videos are not engaging, students tend to lose interest in the course content. This has led to recent research activity in speaking style analysis of educational videos.

10.1. Proposed Approach

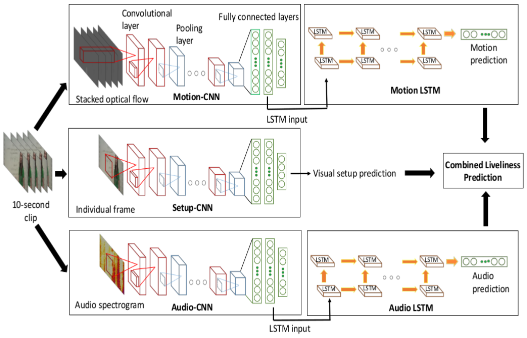


Figure 8. The overall pipeline of LIVELINET

The proposed approach starts begins with modelling the given video as a set of temporal events followed by the visual setup classification algorithm.The audio and visual feature is extracted with the help of CNN which is then used to train LSTM to classify the video.The pipeline of the proposed approach is shown in figure 8

10.2. Visual Setup Classification

Video setup label can be defined as one of the following five type:
1)*Content:*This category includes the scenarios where the video feed mainly displays the content such as a blackboard or a slide or a paper. Since the lecturer is not visible in this case, only the audio modality will be used for liveliness prediction.
2) *Walking/Standing Lecturer:*lecturer walks around or remain in a standing posture. In this the lecturers face and upper body parts (hand/shoulder) should be visible. Both audio and visual modality are used to predict liveliness in this case.
3) *Lecturer Sitting:*The content is not visible and the camera should focus only on the lecturer in a sitting posture. Both audio and visual modalities are considered for liveliness prediction.
4)*Content And Lecturer:*This can be combination of 1,2 and 3.
5)All other categories.
The StyleX dataset [1] was used for liveliness prediction task.

10.3. CNNs for Feature Extraction and Classification

*CNN for Label Classification:*CNN architecture was used for label classification task.The final three fully connected layers (fc6, fc7, fc8) of Alexnet was fine-tuned on the dataset containing 10-second clips from various categories. As Deep neural networks usually have millions of parameters. If the available training data for a particular classification task is not large enough, then training a deep neural network from scratch might lead to over fitting hence we used a pre-trained network.The last fully connected layer of 1000 nodes were replaced with 5 node layer as our objective is to classify each frame into one of the five setup categories.
CNN for Visual Feature Extraction: The visual modality is used to capture the movement of the lecturer hence we refer to this CNN model as Motion-CNN.For the Motion-CNN, the VGG-16 temporal-net trained on UCF-101 [10] was fine-tuned.
*Audio Feature Extraction:*The audio feature at was extracted using a convolutional neural network. For each t, we find a corresponding one second long audio signal from the 10-second clip. Short- Time Fourier Transformation was applied to convert each one second 1-d audio signal into a 2-D image (namely log-compressed mel-spectrograms with 128 components) with the horizontal axis and vertical axis being time-scale and frequency-scale respectively. The CNN features are extracted from these spectrogram images and used as inputs to the LSTM.
LSTM For Liveliness Prediction: The Motion-CNN and the audio-CNN model only the short-term local motion and audio patterns in the video respectively.LSTMs were further employed to capture long-term temporal patterns/dependencies in the video. LSTMs map the arbitrary length sequential information of input data to output labels with multiple hidden units. Each of the units has built-in memory cell which controls the in-flow, out-flow, and accumulation of in formation over time with the help of several non-linear gate units.

References

[1] H. Arsikere, S. Patil, R. Kumar, K. Shrivastava, and O. Deshmukh. Stylex: A corpus of educational videos for research on speaking styles and their impact on engagement and learning. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[2] L. Aulck, N. Velagapudi, J. Blumenstock, and J. West. Predicting student dropout in higher education. *arXiv preprint arXiv:1606.06364*, 2016.

[3] D. S. Chaplot, E. Rhim, and J. Kim. Personalized adaptive learning using neural networks. In *Proceedings of the Third (2016) ACM Conference on Learning@ Scale*, pages 165–168. ACM, 2016.

[4] G. Dekker, M. Pechenizkiy, and J. Vleeshouwers. Predicting students drop out: A case study. In *Educational Data Mining 2009*, 2009.

[5] W. Hämäläinen. Descriptive and predictive modelling techniques for educational technology. *Licentiate thesis, Department of Computer Science, University of Joensuu*, 2006.

[6] C. Jayant, M. Renzelmann, D. Wen, S. Krisnandi, R. Ladner, and D. Comden. Automated tactile graphics translation: in the field. In *Proceedings of the 9th international ACM SIGACCESS conference on Computers and accessibility*, pages 75–82. ACM, 2007.

[7] R. C. Lange and S. Mancoridis. Using code metric histograms and genetic algorithms to perform author identification for software forensics. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, pages 2082–2089. ACM, 2007.

[8] S.-H. Lin. Data mining for student retention management. *Journal of Computing Sciences in Colleges*, 27(4):92–99, 2012.

[9] M. LLC. Knewton Adaptive Learning building the worlds most powerful education recommendation engine, 2015.

[10] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*, pages 568–576, 2014.

[11] M. Sweeney, H. Rangwala, J. Lester, and A. Johri. Next-term student performance prediction: A recommender systems approach. *arXiv preprint arXiv:1604.01840*, 2016.

[12] C. H. Yu, S. DiGangi, A. Jannasch-Pennell, and C. Kaprolet. A data mining approach for identifying predictors of student retention from sophomore to junior year. *Journal of Data Science*, 8(2):307–325, 2010.