IST-Africa 2019 Conference Proceedings
Paul Cunningham and Miriam Cunningham (Eds)
IIMC International Information Management Corporation, 2019
ISBN: 978-1-905824-63-2

Behaviour Prediction in a Learning Management System

Charles LWANDE, Lawrence MUCHEMI , Robert OBOKO
University of Nairobi, School of Computing and Informatics, Box 30197, Nairobi, 00100, Kenya
Email: {lwande.omondi,lmuchemi,roboko}@uonbi.ac.ke

Abstract: Learning Management Systems (LMS) lack automated intelligent components that analyse data and classify learners in terms of their respective characteristics. Manual methods involving administering questionnaire related to a specific learning style and cognitive psychometric tests have been used to identify such behaviour. The problem such method is that a leaner can give inaccurate information, time consuming and prone to errors. Although literature reports complex models predicting leaning styles, only a few have used machine learning methods such as k-nearest neighbour (KNN). The primary objective of this study was to design, develop and evaluate a model based on machine learning model for predicting LS from LMS log records. Approximately 200,000 log records of 199 students who had accessed e-Learning course for a 15-week semester were extracted from LMS to create a dataset. Machine learning concepts were identified from the log records. The dataset was split into training and testing set. A model using K-NN algorithm designed and implemented on using r-studio programming language. The model was trained to predict LS and classify each student based on FSLSM. From this, a model predicting learning behaviour based on the theory was developed and evaluated. Preliminary results are promising demonstrating the model after full validation can be relied on to identify the LS.

Keywords: learning style, learning management system, learner modelling, learner behaviour, machine learning, k-nearest neighbour

1. Introduction

Learning Management System (LMS) is a web based platform that manages the delivery of courses [1]. Such systems create class attendance roster, register users, upload, manage and deliver content, create online tests. Examples include Blackboard Learn, Desire 2 Learn, Moodle, Activate Mind Solutions and Claroline [2]. The systems during student access generate huge volumes of data that can be analysed using machine learning methods to generate knowledge useful for online tutors and lecturers. These include access statistics, login details, and learning progress and server logs. Instructors in most cases rely on a quick view of such basic learning data to monitor progress

It has been reported that individuals have different learning preferences and cognitive behaviour regarding leaning materials they use. Learning style is defined as the most preferred mode of instruction or study [3]. Some examples of learning style theories include Felder Silver Model [4], Myers Briggs Type indicator [5], Kolb learning model [6] and VARK learning styles [7]. The problem with LMS is lack of features that automatically analyse records of students' access to generate knowledge on individual behaviour such as learning styles and cognitive traits but instead treat learners equally. Instructors have previously administered manual instruments such as Index of Learning Style (ILS) questionnaire and psychometric tests like operation task span [8] to gather information on learning styles and cognitive traits respectively. As a result of the above challenges, studies have been conducted to create automated computer applications that analyse log data gathered during students' interactions with LMS without additional effort to use psychometric tools or questionnaire. One recent study reported with significant evaluation results is a system taking users' answers as an input to the system and inferencing using fuzzy logic to predict the learning style [9]. Another related study using intelligent agents to customize and adjust the learning time according to the learner's concentration time during LMS access [10]. The noted issue with studies reviewed is lack of a well analysed list of concepts and a computing model that can be used to identify and group students based on learning styles

Thus, this study sought to develop a computing model that analyses automatically generated logs from LMS and classifies students' behaviour. The study used Felder-Silverman Learning Style Model (FSLSM) as a theoretical framework. FSLSM profiles a learner as active or reflective, sensing or intuitive, sequential or global and visual or global [4]. The model uses a 44 - questions Index of Learning Styles Questionnaire (ILS) with 11 for each dimension as a manual measurement tool. A student selects choice a or b. Questionnaire score sheet classifies a respondent as 1-3 mild, 5-7 moderate,9-11 strong preference for either dimension. A study conducted by [11] indicated the tool generated consistent, reliable and valid results.

The paper is organized as follows. Section 2 states objectives to be addressed by the paper. Section 3 describes methodology. Section 4 discusses machine learning approach used. Section 5 describes modelling steps, 6 results, 7 application area, section 8 industrial benefits and 9 suggests future work.

2. Objectives

The study sought to develop a computing model analysing logs from LMS and classify in terms of LS

The study intended to address the following objectives:

- To identify appropriate theory for identification of learner behaviour
- To identify appropriate concepts as machine learning attributes for identification of learner behaviour
- To develop appropriate computing model for mapping LMS log patterns to learning behaviour theory
- To evaluate the performance level of the model

3. Methodology

The study is based on Felder Silverman Learning Style Model (FSLSM). This theory classifies a leaner as active or reflective, sensing or intuitive, visual or verbal, sequential or global. The study assumes there exists a relationship between the learning traits and online browsing behaviour. The study analysed access records of e-books hosted in LMS, analysed and classified students by mapping to the dimensions proposed in the theory.

Figure below 1 illustrates the modelling steps. Firsts, records of students' access are extracted from LMS log file. Second, access patterns are mapped to relevant LS dimensions described in FSLSM and used as input for developing and training a machine learning model. Third, a machine learning algorithm is used to classify learners in terms of LS. Fourth, the accuracy of the algorithm is evaluated using machine learning method.

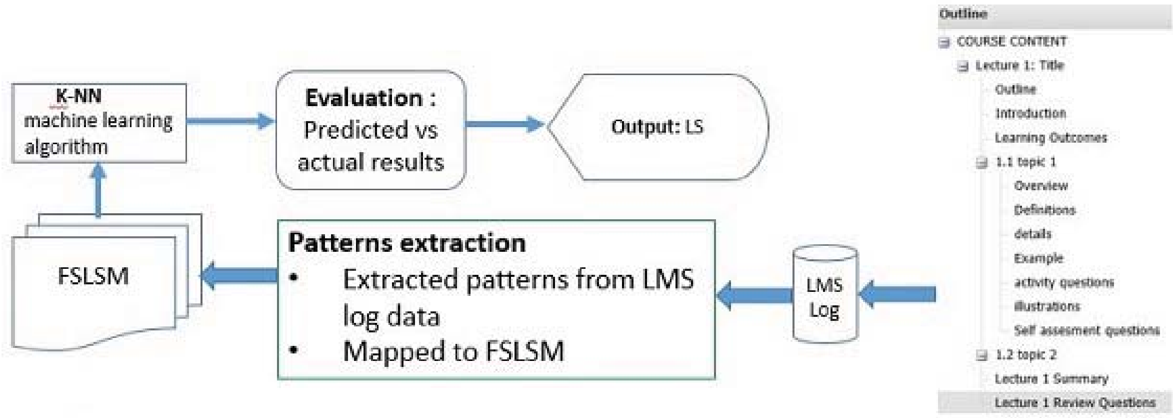

Figure 1: Methodology

4. Machine Learning

Machine learning is a branch in computer science that studies the design of algorithms that take data as input and predict output. It is artificial intelligence technique that help computers to program themselves based on the input data. Machine learning gives AI the ability to do data-based problem solving [12]. Machine learning algorithms are leaning functions f() that map input variable x to output y: y=f(x).

There are several machine learning algorithm such as linear regression, logistic regression, Naive Bayes and decision trees.

K-Nearest Neighbours (K-NN) is a supervised machine learning algorithm that classifies an object based on majority of similar objects closer to a query point by calculating square block or Euclidean distance [13]. It is an instance-based learning, where new data are classified based on stored, labels. Distance between the stored labels and a new instance is estimated by means of a distance measure such as the Euclidean distance, cosine similarity or the Manhattan distance. K-NN algorithm works as follows:

- Determine parameter K= number of nearest neighbours
- estimate the distance between the query instance and all training samples
- Sort the distance and determine the neighbour based on K-th minimum distance
- Gather category Y of nearest neighbours
- Use simple majority of the category of nearest neighbours as the prediction value of the query instance

5. Developments

The study identified concepts from FSLSM theory and mapped them to log patterns. The identified patterns were used as training data

5.1 Felder Silverman Learning Style Model (FSLSM)

The model profiles a learner as active or reflective, sensing or intuitive, sequential or global and visual or global (Felder, 1988). Active learner likes trying things out while reflective think about learned materials. Sensing learner Sensing learner is patient with details, carefully checks answers before delivering a test and visits examples and theories while intuitive prefer challenges. Visual learner like graphic content while verbal prefers discussions, textual and audio materials. Global learner skips, contents wants to get big picture and overview before reading content and also navigates by skipping pages. Sequential read step by step

5.2 *Concepts for Identification of Learner Behaviour*

Patterns matching descriptions of FSLSM dimensions were extracted from relevant learning objects of content pages such as examples, summaries, revisions, result, conclusion, exercises. Tables 1 below shows how patterns were mapped to LS dimensions.

Table 1: Learning Style, Objects and Investigated Patterns

FSLSM	Description	Relevant Learning Object (e-book page)	Pattern extracted
active	Likes trying things out	activities, tests, exercise	%visits and time
reflective	Think about learned materials	Examples, summaries, revisions, result pages, conclusion pages	%visits and time
sensing	dislike challenges, patient with details	Practical, user manuals, explanations, lectures, units, topics	%visits and time
intuitive	likes challenges, impatient with details	Definitions, procedures, meanings, process	%visits and time
verbal	written text, audio, discussing	Text	%visits and time
visual	Graphic, illustrations	Images, illustrations	%visits and time
global	Jumping; whole picture of the content	Area before content, content, area after content	navigation order, visits and time
Sequential	step by step navigation	Area before content, content, area after content	navigation order, visits and time

Table 1 above shows each dimension of FSLSM, description, leaning object investigated and pattern extracted. According to FSLSM, active learner likes trying out things while reflective think about learning materials. Time spent and number of visits on activities, test, exercises show hints for active preference. Time and visits on examples, summaries, revisions, result pages, conclusion pages show preference of reflective. Sensing learner dislikes challenges but patient with details while intuitive like challenges but impatient with details. Visits and time on practical, user manuals, explanations, lectures, units, topics with contents indicate preference for sensing learner. Visits and time spent on content with definitions, procedures and meaning indicate preference for intuitive learner. Verbal learner like text based contents, discussions and audio. Visual learner prefer graphical contents. Time spent and number of visits indicated preference for each dimension. Global learner like getting a full picture of the course contents and sometimes read by skipping pages. A sequential learner navigates content pages step by step. Time and visits on overview pages (area before content) followed by conclusions and summaries (area after content) indicate preference for a global learner. Time and visits on overview pages (area before content) followed by content pages (lectures, units, topics) then conclusions and summaries (area after content) indicate preference for a sequential learner. The patterns extracted were used as training data for predicting learner behaviour

5.3 *Machine Leaning Attributes for Predicting Learner Behaviour*

Inputs and desired outputs were extracted from the patterns by matching access statistics for each student to appropriate LS dimension.

Table 2: Machine Learning Attributes

FSLSM	Description	Relevant patterns from book pages	ML Attribute
active	Likes trying things out	% activities , activity time	A, AT
reflective	Think about learned materials	%area after content, time spent	AAC, AACT
sensing	dislike challenges, patient with details	%content with details, time spent	C, CT
intuitive	likes challenges, impatient with details	%definitions, time	D, DT
verbal	written text, audio, discussing	%visual, time	VS, VS_T
visual	Graphic, illustrations	%visual, time	VS, VS_T
global	Jumping; whole picture of the content	%area before content, time	ABC, ABC_T, AAC
Sequential	step by step navigation	%area before, content, area after, time	ABC,C,AAC

Table 3: Sample Machine Learning Training Data

Name	ABC_%	D_%	C_%	AAC_%	A_%	V_%	ABC_T	D_T	C_T	AAC_T	A_T	LS
ABWAO, J.,M.	85.714	100	40.478	60	0	100	20.714	26.4	99.6	9	0	rssvHL
KINYANJUI, R., W.,	28.571	20.208	20.361	20	0	0	22.571	41.4	431.2	0	0	risrHH
SHARIFF, Y., H.,	85.714	60.308	20.456	80	0	50	80.143	297.4	560.4	156	0	rssrLH
MOHAMMED, S.	85.714	100	40.489	60	0	100	27.143	86.2	231	13.667	0	rssvLL
GONDA, M.	71.429	100	80.178	100	100	100	11.429	47.6	144.4	15.333	36	assvLH
VASISHT, M	85.714	100	60.317	60	100	50	19.857	69.8	139.4	14.667	13	rssrHH

Simple rules were used to map LMS log patterns to LS dimensions.
In table 2 below, a_r stands for active-reflective, s_i sensing-intuitive, sq_g sequential-global and v_vb visual verbal.

IF A, AT > AAC, AACT then a ELSE r
IF C, CT > D, DT then s ELSE i
IF VS, VS_T = 100% then v ELSE r (for reader or verbal)
IF ABC, ABC_T, AAC, > ABC,C,AAC then g ELSE s

Table 4: Mapping LMS LOG Patterns to LS Dimensions

	Active(a) or reflective(r)	Sensing(s) or intutitive (i)	Sequential(s) or global (g)	Visual(v) or verbal(r)	
name	a_r	s_i	s_g	v_r	LS combinations
ABWAO, J.,M.	r	s	s	v	rssv
KINYANJUI, R., W.,	r	i	s	r	risr
SHARIFF, Y., H.,	r	s	s	r	rssr
MOHAMMED, S.	r	s	s	v	rssv
GONDA, M.	a	s	s	v	assv
VASISHT, M	r	s	s	r	rssr

The above rules assign a student four different learning style combinations from each pair of FSLSM dimensions. The learning combination is as follows:

Position 1 active or reflective: a or r,
Position 2: sensing or intuitive: s or i
Position 3 sequential or global: s or g,
Position 4: visual or verbal: v or r

LS: risr for instance means a leaner is reflective, intuitive, sequential and verbal
LS: assv shows a leaner is active, sensing, sequential and visual

5.4 Machine Leaning Dataset for Predicting Learner Behaviour

The model uses supervised learning method to feed access patterns as input and the LS desired outputs. The dataset has 199 instances of records collected from undergraduate medical students from university of Nairobi. There are 13 attributes including class LS and name which identifies a student. Each attribute is detailed as follows:

- ABC – area before content
- D – definitions

- C – detailed content
- A – activity questions
- AAC – area after content (summaries , revision exercises)
- V – content pages with visual illustrations
- ABC T – time on area before content
- D T –time spent on definitions
- C T – time spent on detailed content
- A T – time spent on activity questions
- AAC T –time spent on area after content (summaries , revision exercises)
- LS – learning style to be predicted

LS is the response variable, while the rest are inputs for predictions. The goal of this analysis is to predict LS based on input. There is a relation between thirteen input variables and LS response. The goal is to predict LS value based on the input of thirteen columns.

5.5 Predicting Learner Behaviour Using K-NN

The prediction model was implemented using r-studio programming language. Steps 1 to 6 describe the implementation process.

Step 1: Loading Dataset

Learning behaviour (LB) data set is used for classification with the last attribute of the LS, as the target variable. The dataset is stored as excel file 'LB.xlsx'. The dataset LB is first loaded as shown.

```
library(readxl)
LB<-read_excel("LB.xlsx")
names(LB) <- c("name","ABC", "D", "C", "AAC", "A","V_S", "ABC_T", "D_T", "C_T", "AAC_T","A_T", "LS")
```

The command above reads LB dataset and adds column names

Step 2: Data Exploration

The general distribution of each LS combination in the LB dataset is checked.

```
table(LB$LS)
```

The percentage distribution of each LS combination in the dataset is also checked.

```
round(prop.table(table(LB$LS)) * 100, digits = 1)
```

The minimum value, first quantile, median, mean, third quantile and maximum value of the dataset is also checked

```
summary(LB[c("ABC", "D", "C", "AAC", "A","V_S", "ABC_T", "D_T", "C_T", "AAC_T","A_T")])
```

Step 3. Data Normalization

The dataset is normalized using `normalize()` function to ensure consistency

```
normalize <- function(x) {
 num <- x - min(x)
 denom <- max(x) - min(x)
 return (num/denom)}
LB_norm <- as.data.frame(lapply(LB[2:12], normalize))
```

Step 4. Split Training and Test Sets

The dataset is divided training and test set. The first is used to train the system, while the test is used for evaluation. The dataset is split by taking 2/3 as training and 1/3 as testing set.

```
ind <- sample(2, nrow(LB), replace=TRUE, prob=c(0.67, 0.33))
LB.training <- LB[ind==1, 2:12]
LB.test <- LB[ind==2, 2:12]
```

The training and test set only select the second to twelfth attributes: "ABC", "D", "C", "AAC", "A","V_S", "ABC_T", "D_T", "C_T", "AAC_T","A_T".

The class labels are stored in factor vectors and divided as the training and test sets

```
LB.trainLabels <- LB[ind==1,13]
LB.testLabels <- LB[ind==2, 13]
```

The thirteenth attribute LS is the class to be predicted.

Step 5. Building K-NN Model

The next step is estimating the k nearest neighbours of the training set. This is implemented using using the knn() function of r-studio. The function uses the Euclidian distance method to find the k-nearest neighbours to unknown instance. The following commands show the training procedure

```
cl =  LB.trainLabels[,1, drop = TRUE]
cl_2 = LB.testLabels[,1, drop = TRUE]

LB_pred <- knn(train = LB.training, test = LB.test, cl, k=3)
```

The LB_pred is assigned the knn() function that takes training, the test set, the train labels and the number of neighbours to generate predicted classes for each row of the test data.

Step 6. Evaluation of the Model

This is done to check the accuracy by comparing predicted against test results:

```
LBTestLabels <- data.frame(LB.testLabels)
merge <- data.frame(LB_pred, LB.testLabels)
names(merge) <- c("Predicted LS", "Observed LS")
merge
```

Further evaluation is done using a cross tabulation table is often to understand how the classes of test data relate to predicted

```
CrossTable(x = LB.testLabels, y = LB_pred, prop.chisq=FALSE)
CrossTable(cl_2, y = LB_pred, prop.chisq=FALSE)
```

6. Results

This section discusses the preliminary results for evaluating the performance of the model in predicting LS.

Figure 2 below shows minimum, first quantile, median, mean, third quantile and maximum value of the LB dataset. Figure 3 shows percentage distribution of LS

```
> summary(LB[c("ABC", "D", "C", "AAC", "A","V_S", "ABC_T", "D_T", "C_T", "AAC_T","A_T")])
      ABC                D                  C                AAC
 Min.   :  0.00   Min.   :  0.175   Min.   :  0.25   Min.   :  0.00
 1st Qu.: 71.43   1st Qu.: 60.183   1st Qu.: 40.13   1st Qu.: 40.00
 Median : 85.71   Median :100.000   Median : 80.00   Median : 80.00
 Mean   : 80.22   Mean   : 80.195   Mean   : 64.69   Mean   : 72.56
 3rd Qu.:100.00   3rd Qu.:100.000   3rd Qu.:100.00   3rd Qu.:100.00
 Max.   :200.00   Max.   :100.000   Max.   :100.00   Max.   :100.00
       A               V_S             ABC_T             D_T
 Min.   :  0.00   Min.   :  0.00   Min.   :  1.00   Min.   :   6.0
 1st Qu.:  0.00   1st Qu.: 50.00   1st Qu.: 10.36   1st Qu.:  34.5
 Median :100.00   Median :100.00   Median : 19.29   Median :  69.6
 Mean   : 73.37   Mean   : 82.16   Mean   : 31.57   Mean   : 126.6
 3rd Qu.:100.00   3rd Qu.:100.00   3rd Qu.: 33.36   3rd Qu.: 128.1
 Max.   :100.00   Max.   :100.00   Max.   :390.43   Max.   :1036.4
      C_T             AAC_T              A_T
 Min.   :  20.4   Min.   :  0.000   Min.   :  0.00
 1st Qu.:  97.4   1st Qu.:  6.333   1st Qu.:  0.00
 Median : 176.0   Median : 14.333   Median : 15.00
 Mean   : 336.2   Mean   : 20.645   Mean   : 39.67
 3rd Qu.: 367.6   3rd Qu.: 23.167   3rd Qu.: 33.00
 Max.   :2152.2   Max.   :177.333   Max.   :807.00
```

Figure 2: Dataset Distribution

```
> round(prop.table(table(LB$LS)) * 100, digits = 1)

aigr aisr aisv asgr asgv assr assv rigr rigv risr risv rsgr rsgv rssr rssv
 1.0  7.5 17.6  3.5  4.0  5.0 15.6  0.5  0.5  5.5 14.1  1.0  3.0  8.0 13.1
>
```

Figure 3: LS % Distribution

Training was repeatedly done by adjusting the value of k-n and recording outcome. The model evaluation is done by adjusting k-value as shown in the table 5 below.

Table 5: Training Results

k-value	correct predictions	incorrect predictions
2	22	35
3	20	37
4	19	38
5	18	39
6	18	39
7	17	40
8	19	38

Preliminary results indicate the correct predictions increase with a smaller k-value
The figure 4 below shows results for first ten predictions.

	Predicted LS	Observed LS
1	risv	assv
2	assv	assv
3	rssv	rssv
4	assv	assv
5	asgv	assr
6	rssr	rsgr
7	aisv	assv
8	assv	assv
9	asgv	asgv
10	risr	rssv

Figure 4: Sample Predicted vs Observed Results

The model makes five wrong and right predictions for first 10 instances (k=3)

Figure 5 below shows cross tabulation results with correct and incorrect predictions. The learning style combination aigr has only one instance which is wrongly predicted as assv. The learning style combination aisr has a total of five instances with two correctly predicted, the rest classified as assr,risv and rssr respectively. The learning style combination aisv has a total of nine instances with four correctly classified, two as risv and the rest aisr, asgr and rssv each.

```
   Cell Contents
|-------------------------|
|                       N |
|           N / Row Total |
|           N / Col Total |
|         N / Table Total |
|-------------------------|

Total Observations in Table:  57

             | LB_pred
  cl_observe |      aisr |      aisv |      asgr |      asgv |      assr |      assv |      risv |      rsgv |      rssr |      rssv | Row Total |
-------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
        aigr |         0 |         0 |         0 |         0 |         0 |         1 |         0 |         0 |         0 |         0 |         1 |
             |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     1.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.018 |
             |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.100 |     0.000 |     0.000 |     0.000 |     0.000 |           |
             |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.018 |     0.000 |     0.000 |     0.000 |     0.000 |           |
-------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
        aisr |         2 |         0 |         0 |         0 |         1 |         0 |         1 |         0 |         1 |         0 |         5 |
             |     0.400 |     0.000 |     0.000 |     0.000 |     0.200 |     0.000 |     0.200 |     0.000 |     0.200 |     0.000 |     0.088 |
             |     0.400 |     0.000 |     0.000 |     0.000 |     0.167 |     0.000 |     0.091 |     0.000 |     0.250 |     0.000 |           |
             |     0.035 |     0.000 |     0.000 |     0.000 |     0.018 |     0.000 |     0.018 |     0.000 |     0.018 |     0.000 |           |
-------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
        aisv |         1 |         4 |         1 |         0 |         0 |         0 |         2 |         0 |         0 |         1 |         9 |
             |     0.111 |     0.444 |     0.111 |     0.000 |     0.000 |     0.000 |     0.222 |     0.000 |     0.000 |     0.111 |     0.158 |
             |     0.200 |     0.400 |     0.500 |     0.000 |     0.000 |     0.000 |     0.182 |     0.000 |     0.000 |     0.167 |           |
             |     0.018 |     0.070 |     0.018 |     0.000 |     0.000 |     0.000 |     0.035 |     0.000 |     0.000 |     0.018 |           |
-------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
        asgr |         0 |         0 |         0 |         0 |         0 |         0 |         0 |         0 |         0 |         1 |         1 |
             |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     1.000 |     0.018 |
             |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.167 |           |
             |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.000 |     0.018 |           |
-------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
```

Figure 5: Sample Cross Tabulation Results (k=3)

7. Business Benefits

This paper brings forth a generic modelling technique that developers can integrated with existing learning management system platform to improve learner characterization

Furthermore, a lecturer can use information generated by the model to provide learning materials matching identified characteristics. The results enable online tutors to understand students so as to apply appropriate teaching methods matching their learning characteristics

Despite the above benefits, interventions built around learning styles are costly since students are assessed and grouped by similar characteristics and then provided with customized instruction [3]. This requires additional teacher training as well as the customization of instructional activities for each learning style. Integrating the proposed prototype with an existing learning management system that automatically creates groups according to learning preference can improve teaching efficiency. Institutions can reduce the cost of individual instructions by assigning group tutors. Furthermore, instruction designers can prepare learning materials matching all preferences such as active- reflective, sensing-intuitive, sequential-global, visual-verbal. The system can be programmed to automatically match contents with relevant learning styles. This will reduce the cost of learning style intervention raised above

8. Industrial Significance and Benefits

Students differ from one another in terms of instructions they respond to best, ways they approach their studies and their attitude on the nature of knowledge**. S**ilent leaners are hard to track in an online course due to their near invisibility [14]**.** It is important to develop strategies to encourage such leaners become more active**s**. Integrating a model which automatically identifies learning styles enable institutions identify such students and offer closer attention thus higher retention and less dropout rates. Knowing student behaviours can help teachers to get information on students' preferences and ways in which they learn. Students become aware of their own learning styles, strengths and weaknesses in the learning process

9. Conclusions

This paper has demonstrated how a machine learning approach can be used to predict LS based on a learning theory. It is feasible on larger scale to predict learning styles from corpus of historical data to enable instructors provide personalized learning and education. The prototype plugin can be integrated with existing learning management system platform to automatically present learning materials matching individual behaviour. This is done by continuously analysing system access records to classify leaners and match contents to learning styles.

The model evaluation progress is giving promising results. This is work on progress and it is envisaged that the model will generate more accurate results after proper training and validation.

Additional tests are being carried out to improve the performance of the model

References

[1] J. Ondřej, M. Hála, and V. Přech, "Some Thoughts about e-Learning in Teaching International Law," in *Czech Yearbook of International Law*, no. November 2008, Czech Society of International Law, 2010, pp. 223–231.

[2] M. Don, "List Of Corporate Learning Management Systems," 2014. [Online]. Available: https://elearningindustry.com/list-corporate-learning-management-systems.

[3] H. Pashler, M. Mcdaniel, D. Rohrer, and R. Bjork, "Learning Styles: Concepts and Evidence What Evidence Is Necessary to Validate Interventions Based on Learning Styles? Evaluation of Learning-Styles Literature 111 Style-by-Treatment Interactions: The Core Evidence Is Missing Learning-Styles Studies With A," *Psychol. Sci. Public Interes.*, vol. 9, no. 3, pp. 105–119, 2008.

[4] R. M. Felder, "Learning and Teaching Styles in Engineering Education," *Eng. Educ.*, vol. 78, no. 7, pp. 674–681, 1988.

[5] I. B. Myers, *Gifts Differing: Understanding Personality Type*, Reprint. Davies-Black Publishing, 1995.

[6] D. Kolb, A., *Experiential Learning: Experience as the Source of Learning and Development*, 2nd ed. Pearson Education, 2015.

[7] N. D. Fleming, "The VARK Modalities," 2014. [Online]. Available: https://web.archive.org/web/20150314235648/http://vark-learn.com/introduction-to-vark/the-vark-modalities/.

[8] M. L. Turner and R. W. Engle, "Is working memory capacity task dependant ?," *Journal of Memory and Language*, vol. 28, no. 2. pp. 127–154, 1989.

[9] A. Ozdemir, A. Alaybeyoglu, N. Mulayim, and M. Uysal, "An Intelligent System for Determining Learning Style," *Int. J. Res. Educ. Sci.*, pp. 208–214, 2018.

[10] I. Kamsa, R. Elouahbi, and F. El Khoukhi, "Study Smart Not Hard," *Turkish Online J. Distance Educ.*, vol. 19, no. 1, pp. 62–74, 2018.

[11] T. A. Litzinger, S. Lee, J. Wise, and R. M. Felder, "A Psychometric Study of the Index of Learning Styles ©," *J. Eng. Educ.*, vol. 96, no. 4, pp. 309–319, 2007.

[12] G. Ciaburro and B. Venkateswaran, *Neural Networks with R: Smart models using CNN, RNN, deep learning, and artificial intelligence principles*. Packt Publishing, 2017.

[13] D. Barber, *Bayesian Reasoning and Machine Learning*. 2016.

[14] S. Honeychurch, A. Bozkurt, L. Singh, and A. Koutropoulos, "Learners on the Periphery: Lurkers as Invisible Learners," *Eur. J. Open*, vol. 20, no. 1, pp. 191–1027, 2017.