Pervasive Student Profiling Method for Team Matching Recommendation

By

Lean Wei Fong

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

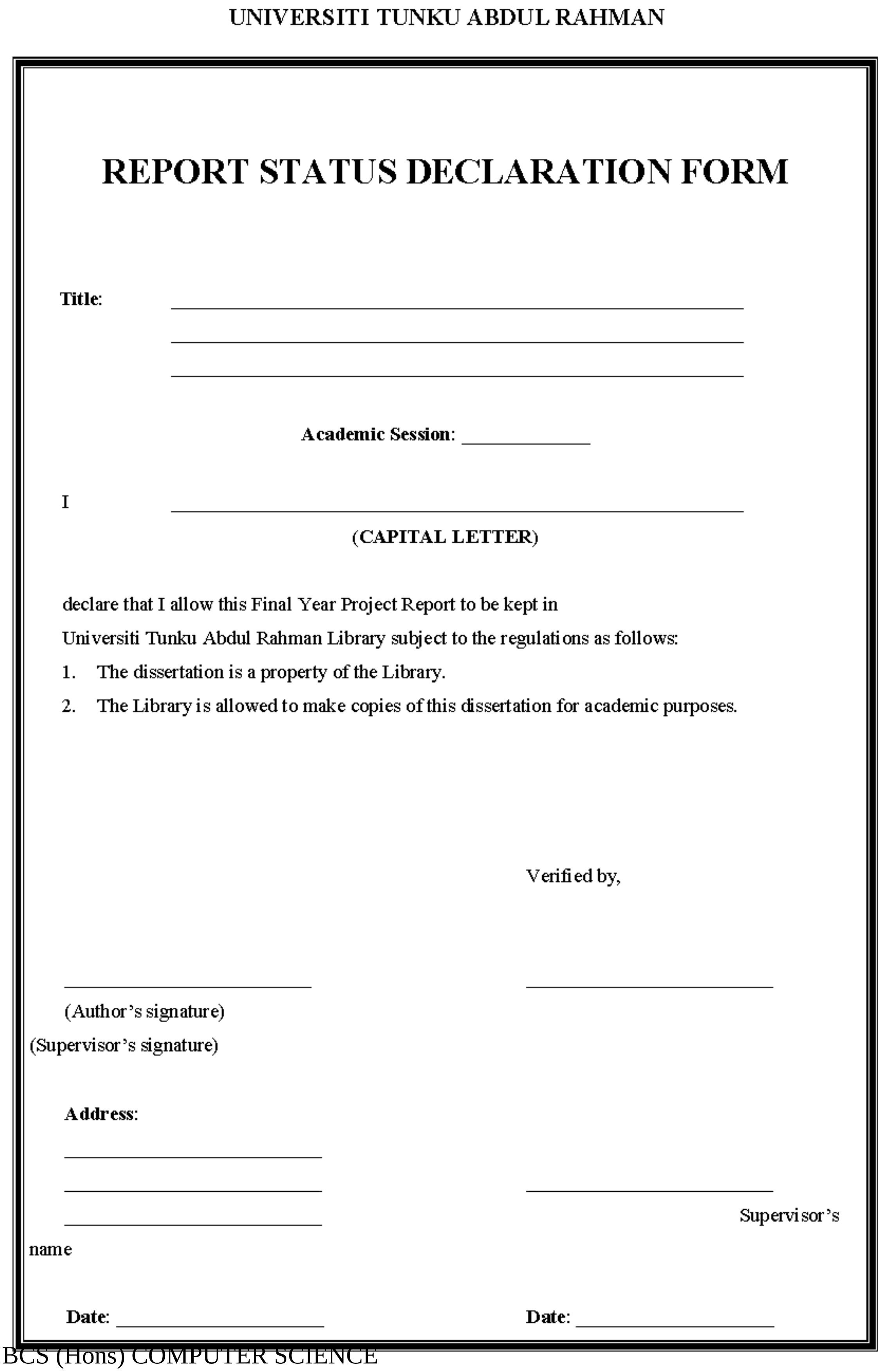
for the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONS)

COMPUTER SCIENCE

Faculty of Information and Communication Technology
(Kampar Campus)

MAY 2019



UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: ____

____

____

Academic Session: ____________

I ____

(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

_________________________ _________________________

(Author's signature)
(Supervisor's signature)

Address:

____

_________________________ _________________________

_________________________ Supervisor's
name

Date: ____________________ **Date**: ____________________

Pervasive Student Profiling Method for Team Matching Recommendation

By

Lean Wei Fong

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONS)

COMPUTER SCIENCE

Faculty of Information and Communication Technology
(Kampar Campus)

MAY 2019

DECLARATION OF ORIGINALITY

I declare that this report entitled “Pervasive Student Profiling Method for Team Matching Recommendation” is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : ___________________________

Name : ___________________________

Date : ___________________________

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisors, Dr Aun Yichet who has given me this bright opportunity to engage in an Machine Learning project. A million thank to you for guiding me throughout the whole project.

Finally, I must say thanks to my parents and my family for their love, support and continuous encouragement throughout the course.

ABSTRACT

Profiling is a process where analysis are done on an individual phycologically and behavior in order to predict or classify a group of similar individuals. In this paper, several method to profile student had been introduce, student are profiled with 5 defined attributes which are Learnability, Productivity, Leadership, Domain Skillsets and Exploring. Student profile are used to serve for one application, which is assignment team matching recommendation. This can be done by using machine learning technique to predict each profile respective assignment role's performance rating. There are 4 roles defined which are Planning, Writing, Idea and Technical. With these 4 roles being predicted for each profiled individual, the team matching recommendation can be done based on the rating of these 4 roles of a student. This is to help recommend a much more effective and balance assignment group.

With the help of this team matching recommending system, problem for right assignment group formation can be solved, as this is the most common problem that can be found in any university. Student can take the recommendation as a reference to form more effective assignment group.

Contents

List of Tables

List of Figure

List of Abbreviation

AI	Artificial intelligent
WOE	Weight Of Evidence
EEG	Electroencephalography
KNN	K- nearest neighbor
KDD	Knowledge discovery in database
CNN	Convolutional neural network
SVM	Support vector machine
MLFS	Machine learning feature selection
GPU	Graphic processing unit
LSTM	Long short-term memory
URL	Uniform resource locator

Chapter 1: Introduction

1.1 Problem Statement and Motivation

The most common problem many of the student had meet in their university life is forming a “right” group for assignment task. Being in a good or bad group to complete any task is always determined by the people in a group, and the productivity and efficiency of completing certain task is always determined by the group itself as well. Better group, better result; bad group where groupmates are not even compatible. As you can see having a compatible groupmates and be in a right group is extremely important. The most common way of forming assignment group is either people decided stay in the comfort zones where they team up with their friends or simply form a group in class with people that they don’t even recognize or know. Group formation is the foundation of effective group works, that’s why group formation itself is the first steps and the very crucial steps of group assignment in university, a lot of problems may arise when doing group work if this process didn’t handle well. According to a research done by W.Martin Davies (2009), motivation of participants is the most serious problem in groupworks. At some situation, some groupmates might be unnecessary in certain assessment task or may not fully committed to the goal of the group. In the research, W.Martin Davies (2009) discuss about 2 example of motivational issues that arises from the problem mention above, which are social loafing and free riding. Social loafing in the situation when certain groupmates have lesser contribution or put lesser effort as compares to other because of lacking attention by others in the group work. For free riding normally describe groupmate that didn’t do anything at all and taking advantages of others who contribute and work hard to accomplish the group work. However, problem might can be solved by using some technical approach method which will be discussed in this paper. A pervasive student profiling method for team matching recommendation. Understanding students personality characteristics can get us more information about their behavior and role’s performance on completing different job scope of certain assignment task. With the help of machine learning the method mention above will be more easier. By knowing their ability, we are able to use the data to performed more balanced team matching by on utilizing their skill and ability.

1.2 Background

Personality refers to characteristics, way of thinking, behavior and emotion pattern of an individuals. An individual may expose his/her personality to the others time to time if they get along long enough. However, with the advent of AI technology, it is possible to predict an individual personality or characteristics by using enough dataset about that individual. This paper will be discussing about a pervasive student profiling method for team matching recommendation which is to use generated student profile to help in team matching recommendation. Profiling is the process where analytic is done on an individual to know he/she psychologically or behavioural in order to predict capabilities or classified people. With characteristics profile of individuals, we can use it as datasets to perform all kind of prediction using machine learning. Machine learning techniques; subfield of Artificial Intelligent which is an application that helps to provide system the ability of learning automatically without human interruption or explicitly programmed. Machine Learning is more concern about develop a computer program that gathered enough data and study the data, finding pattern and eventually learn itself. There are several industries can get used to the help of machine learning techniques and solve a lot of problems with it. However, in this paper we will be focusing on education industries. Sometimes by knowing student personality or characteristics, we can get to know other behaviour of respective student as personality somehow determine their behaviour. For example, a student with high leadership personality will eventually lead others people and planning job scope in completing certain assignment and task. This is the core idea of this research paper, where few characteristics attributes will be defined in this paper then will all this attribute, a student profiling is done. Method of data collecting will be defined as well to collect different data from various data sources to determine the rating of these defined characteristics attributes. With the student characteristics profile generated, based on these profiles, we will eventually try to predict respective student’s performance on different job scope in completing an assignment. The prediction will be done using machine learning algorithms. By understanding student’s role performance on completing different jobs scope we can then use this as reference to help to performing team matching recommendation, where different individuals who are good at different aspect of job will be group together to form a more balanced assignment team. This whole idea can help in recommending a more effective team or so called "right" team to complete any task given.

1.3 Project Objectives

The objectives of this project:

I. **Identify a set of attributes that describes an individual's characteristics and use it as a profile of respective student**

Each attribute should be able to define a person in terms of education aspect. Then different data collecting method will be used on collecting data respective to each attribute. The data collected will then go through some calculation to determine to attributes value and eventually profiling student based on these attributes value.

II. **To predict Student's ability on performing different job scopes of an assignment task**

Used the generated profile of student to perform prediction of student's different role performance of completing a task. Machine learning algorithms will be used to learn and train the datasets to create a model for prediction.

III. **Perform Team matching recommendation**

Recommendation of different combination of groups of people will be done as team matching by using the predicted role performance of each student as the criteria to look at and form a much balanced team.

1.4 Proposed work

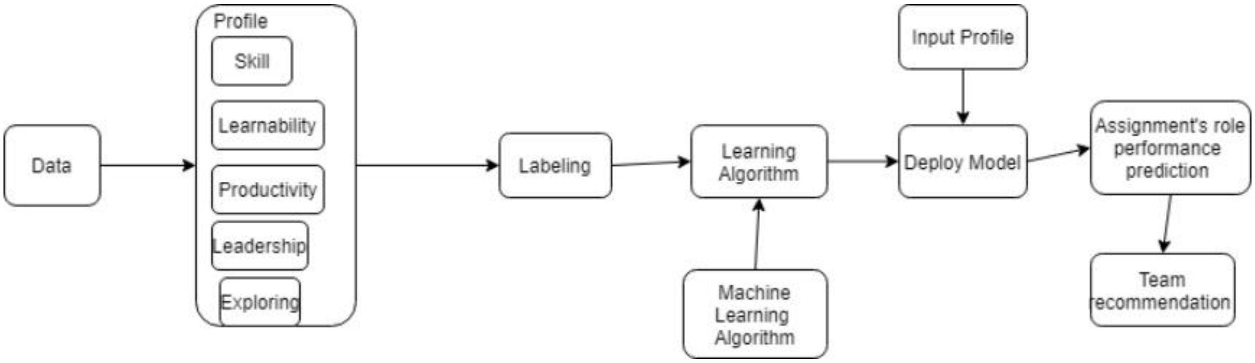


Diagram 1.4.1 shows the proposed system design of this research

Above diagram shows the proposed method for this research. The most crucial stage of all this will be the data collecting stage where all the data will be collected from various data source in order to determine the value of each attributes in the profiling process. All this profile will then label with respective student's performance in participating different role in completing assignment task. The labeled data will be used to fit into the machine learning algorithms for the purpose of training and eventually come up with a model to perform prediction. All this predicted data will act as a reference to perform team matching recommendation.

1.5 Background Information

Deep learning

Deep learning is a subfield of machine learning. Deep learning can use different algorithms that are stacked in a hierarchy which always increasing its complexity and abstraction that impersonate the human learning processes. The input of a deep learning is huge amounts of training datasets then the computer programs will use deep learning and go through iteration of deep learning process until a result with good accuracy is outputted. With the use of deep learning, the system can build the feature without any supervision. This is also called feature learning.

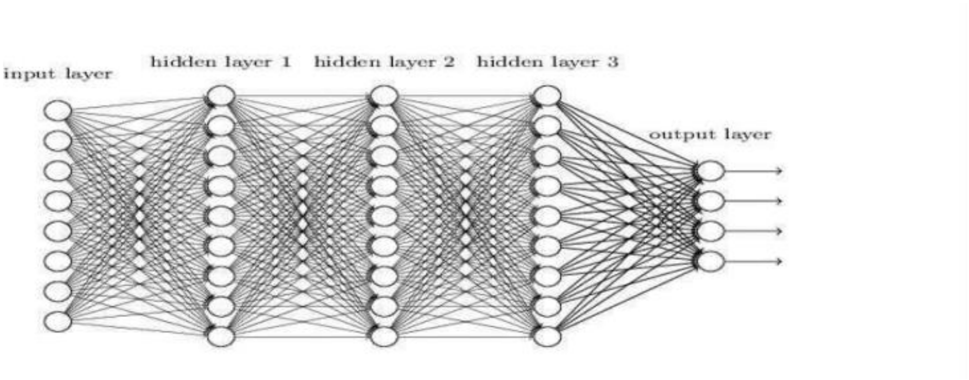


Diagram 1.5.1 Shows Deep Neural Network architecture. (Source from: https://stats.stackexchange.com/questions/182734/what-is-the-difference-between-a-neural-network-and-a-deep-neural-network-and-w)

Deep learning often called deep neural networks, this is because most deep learning methods use neural network architectures. Deep neural network can have hundreds of hidden layer in the network. Below is the diagram of deep neural network:

Machine learning process

Machine learning is used to recognizing undiscovered pattern with computer can learn by themselves without human programming. To do so, a lot of data are required for it to learn. First of all data collection need to be performed, quantity & quality of data determine the accuracy of the model. Then prepare the data by cleaning it, randomize it visualize it and split it into training set and testing set. After all data is set up, choose a right model for the task, then train the model. To know how well the model perform, evaluate the model by testing the model with unseen data. Then parameter tuning can also be performed to tune model parameters for performance improving. Last but not least, use the test set data to make the prediction on the task.

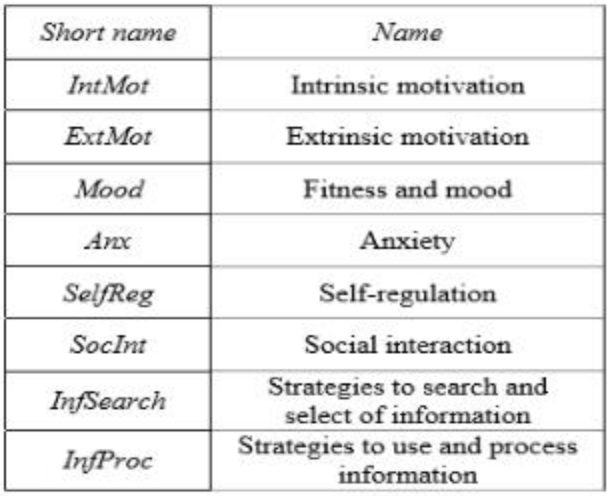
1.6 Impact, significance and contribution

By far there are many existing researches which research about profiling individual using AI then use the profile to predict the relationship between the profile with certain problem. However, most of this existing research, the way they perform profiling is just filling up questionnaire. Questionnaire sometime is not accurate enough as the person who fill up can be dishonest, unconscientious, bias or different in understand the question. It came up there are quite a lot of factor that might influence the accuracy of profiling itself. In this paper, the profiling method used is definitely not using questionnaire method, instead, the data to profile an individual came up from various data source, like student database, tracing student visited website and more. The whole profiling process is based on what we observed from the data gathered. Beside that, the process can be automated as well. So, compare to questionnaire, the method proposed in this paper is much more effective but a lot of work needed to be done compare to just designing question for people to fill up.

Chapter 2: Literature Review

2.1 Predict student academic performance based on self-regulating characteristics

This research is done by Julieta Noguez, Luis Neri, et.al (2016), who study to classify student in group according to their self-regulation skills (SR), learning strategies (LeSt) and affective strategies (AfSt). The cluster of this generated student profile will then use for studying the relationship between student academic outcomes and their relative cognitive and social profile.The method they used to profile student is using item that will help to identify relevant dimension and characterize student's profile. This items are grouped and 8 dimension were defined.



Short name	*Name*
IntMot	Intrinsic motivation
ExtMot	Extrinsic motivation
Mood	Fitness and mood
Anx	Anxiety
SelfReg	Self-regulation
SocInt	Social interaction
InfSearch	Strategies to search and select of information
InfProc	Strategies to use and process information

Diagram 2.1.1 8 dimensions that characterize a student profile. (Source from: Julieta Noguez, Luis Neri, et.al (2016))

Then they upload set of questionnaires to sample of 96 engineering student which enrolled in math, physics and computer science courses. The questionnaires consist of questions that will determine the value of 8 dimensions that characterize a student.

The answered questionnaires of each student will then calculate by averaging the values of each dimension. These averaged values of dimension will then be the student personal profile. The generate profile of each student will form in a spider web diagram.

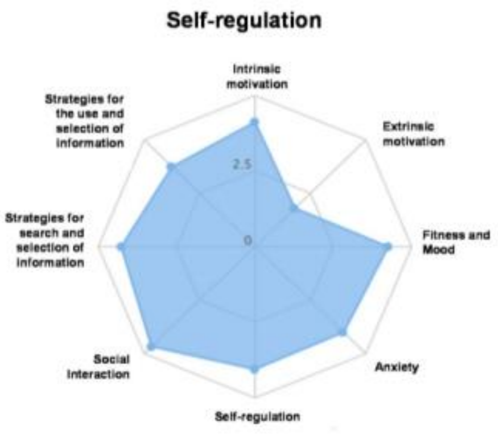
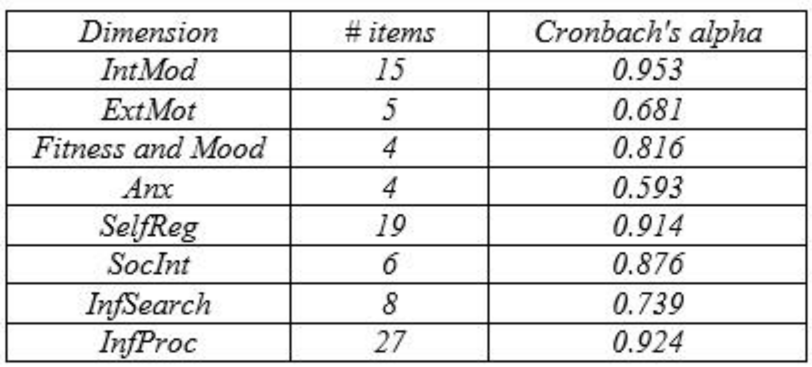


Diagram2.1.2 Spider web diagram of an individual student. (Source from: Julieta Noguez, Luis Neri, et.al (2016))

These profiles will then go through some reliability and social validation test for evaluation purpose. Below is the result of the reliability test using Cronbach's Alpha:



Dimension	# items	Cronbach's alpha
IntMod	15	0.953
ExtMot	5	0.681
Fitness and Mood	4	0.816
Anx	4	0.593
SelfReg	19	0.914
SocInt	6	0.876
InfSearch	8	0.739
InfProc	27	0.924

Diagram 2.1.3 result of the cronbach's Alpha for each dimensions. (Source from: Julieta Noguez, Luis Neri, et.al (2016))

According to Julieta Noguez, Luis Neri, et.al (2016), based on the result of the Cronbach's alphas, generally all dimension is properly characterized and differentiated, and is suitable for determining student profiles. But more exploration should be done to increase Anxiety and Extrinsic Cronbach's value. As for their social validation test, According to Julieta Noguez, Luis Neri, et.al (2016) the validation shows very good agreement between the profiles assigned by system and those perceived by students.

These show us that Profiling by using questionnaire can also use to understand an individual behavior, reliability test and social validation test should be conduct to evaluate the accuracy of the generated profile. However, data provide by questionnaire can still be dishonest, even with social validation test, the accuracy might not be honestly true.

2.2 Learning Human Behavioral Profiles in a Cyber Environment

Michael Forte, Christopher Hummel, et.al (2010) had done research on profiling online shopper's pattern of behavior using data mining techniques. There are two metrics for profiling behavior of online shoppers are:

I. Classified the data into sub-categories of profile, gender and major confidence. Then analyse these group based on statistical differences and time in order to find patterns in these difference demographic groups.

II. Create a regression model to determine the usefulness of the data for attributes application. 3 Models are created for different attribute of individual.

Three types of analysis method which are significance testing, regression modeling and Weight of evidence (WOE) are used to study the behavior pattern of online shoppers. Significance testing is analysis which look for significant differences. Regression model is used to accurately predict undiscovered subject's demographic information. 2 regression model is used as the first model take all collected subject data into account and the second set model involve only using randomly re-sampling from the entire data set as a model to testing on the rest of the data not selected. Lastly Weights of evidence (WOE), used to test the ability of accurate prediction of demographic characteristics of user using scoring system.

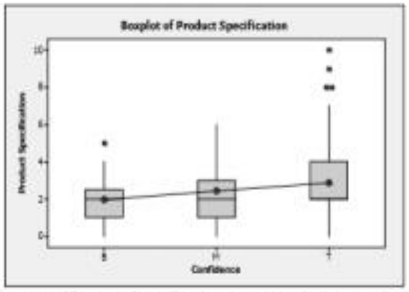


Fig 2. Boxplot of Product Specification (number of clicks) v. Confidence.

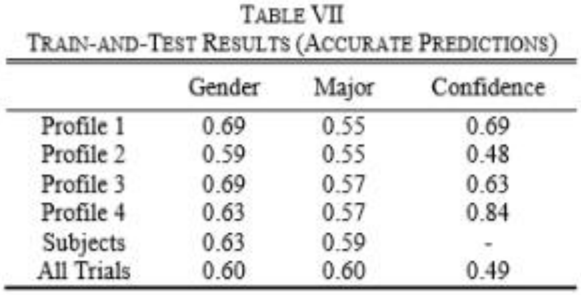


TABLE VII
TRAIN-AND-TEST RESULTS (ACCURATE PREDICTIONS)

	Gender	Major	Confidence
Profile 1	0.69	0.55	0.69
Profile 2	0.59	0.55	0.48
Profile 3	0.69	0.57	0.63
Profile 4	0.63	0.57	0.84
Subjects	0.63	0.59	-
All Trials	0.60	0.60	0.49

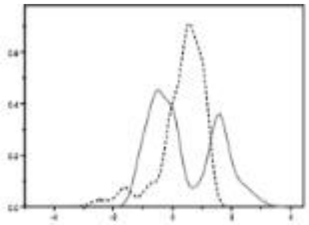


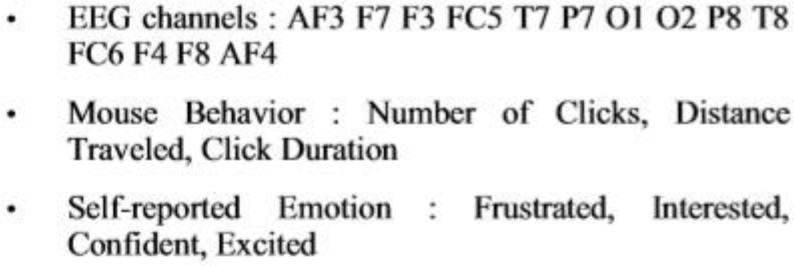
Fig. 4. Density plot based on weights of evidence for males (solid line) and females (dotted line).

Diagram 2.2.1 example of method of analysis (Source from: Michael Forte, Christopher Hummel, et.al (2010))

Logistic regression model is a good model for prediction, but in order to make it more accurate or allow much more robust analysis, larger sample size is needed. WOE techniques can be useful in credit scoring system to predict behavioral patterns.

2.3 Use of Personality Profile in Predicting Academic Emotion based on Brainwaves Signals and Mouse Behavior

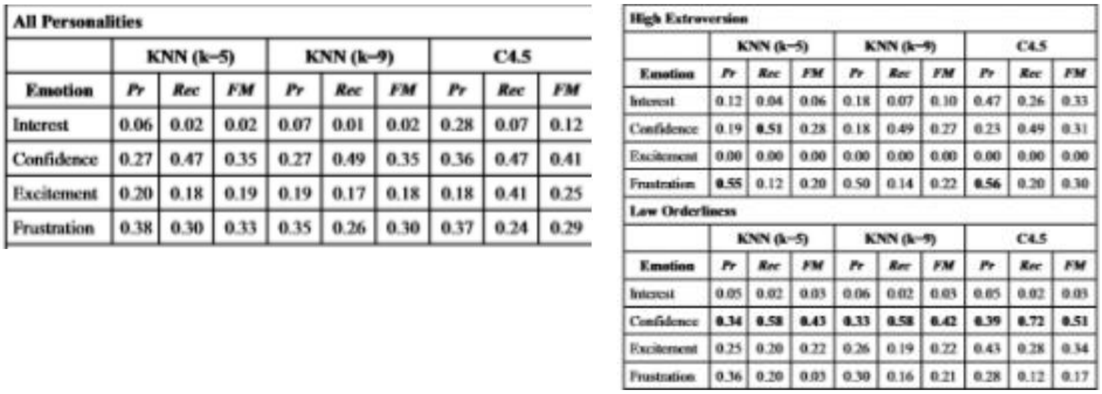
This study is done by Judith J.Azcarraga*, John Francis Ibanez Jr., et.al (2011), In this research, prediction of student academic emotion is done by collecting data through a learning software where the data is in the form of student's brainwaves signals and mouse-clicks activities. The brainwaves signals collected using an Emotiv EEG sensor while the mouse-clicks activities are the number of click, distance traveled by mouse and duration of each click. The EEG sensor is attached to head of student, then during the learning session EEG signals and mouse data is being observed. The features used for emotion classification are:



- EEG channels : AF3 F7 F3 FC5 T7 P7 O1 O2 P8 T8 FC6 F4 F8 AF4
- Mouse Behavior : Number of Clicks, Distance Traveled, Click Duration
- Self-reported Emotion : Frustrated, Interested, Confident, Excited

Before the observation, a questionnaire is also distributed for student to fill up to measure his/her personality traits which are extroversion, orderliness, emotional stability, accommodation, inquisitiveness.

KNN classifier and C4.5 machine learning algorithm were used to predict the emotion of different personality traits.



All Personalities

	KNN (k=5)			KNN (k=9)			C4.5		
Emotion	***Pr***	***Rec***	***FM***	***Pr***	***Rec***	***FM***	***Pr***	***Rec***	***FM***
Interest	0.06	0.02	0.02	0.07	0.01	0.02	0.28	0.07	0.12
Confidence	0.27	0.47	0.35	0.27	0.49	0.35	0.36	0.47	0.41
Excitement	0.20	0.18	0.19	0.19	0.17	0.18	0.18	0.41	0.25
Frustration	0.38	0.30	0.33	0.35	0.26	0.30	0.37	0.24	0.29

High Extroversion

	KNN (k=5)			KNN (k=9)			C4.5		
Emotion	***Pr***	***Rec***	***FM***	***Pr***	***Rec***	***FM***	***Pr***	***Rec***	***FM***
Interest	0.12	0.04	0.06	0.18	0.07	0.10	0.47	0.26	0.33
Confidence	0.19	**0.51**	0.28	0.18	0.49	0.27	0.23	0.49	0.31
Excitement	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Frustration	**0.55**	0.12	0.20	0.50	0.14	0.22	**0.56**	0.20	0.30

Low Orderliness

	KNN (k=5)			KNN (k=9)			C4.5		
Emotion	***Pr***	***Rec***	***FM***	***Pr***	***Rec***	***FM***	***Pr***	***Rec***	***FM***
Interest	0.05	0.02	0.03	0.06	0.02	0.03	0.05	0.02	0.03
Confidence	**0.34**	**0.58**	**0.43**	**0.33**	**0.58**	**0.42**	**0.39**	**0.72**	**0.51**
Excitement	0.25	0.20	0.22	0.26	0.19	0.22	0.43	0.28	0.34
Frustration	0.36	0.20	0.03	0.30	0.16	0.21	0.28	0.12	0.17

Diagram 2.3.1 Example of prediction of academic emotion of student based on EEG data and mouse click behavior. (Source from: Judith J. Azcarraga, John Francis Ibanez Jr., et.al (2011))*

Judith J. Azcarraga*, John Francis Ibanez Jr., et.al (2011) research shows that individual behavior pattern can also predicted using data collected from individual’s daily event activities, and there is also potential of combining physiological signals with these data to improve accuracy of prediction. However, tracking physiological signals might need expensive sensor for affect detection, adequate funding is need for this type of research.

2.4 Academic Advising System Using Data Mining Method for Decision Making Support

Riah F. Elcullada-Encarnacion(2018) had study is about developing a system that facilitates academic advising section and keep track of student academic performance. The system implement the data mining method that helps generating required data set for analysis and data modeling eventually support academic decision making for student.

A Modified KDD process were used to approach data mining. KDD is a process that transformed raw data to a pattern that helps discovered new knowledge.

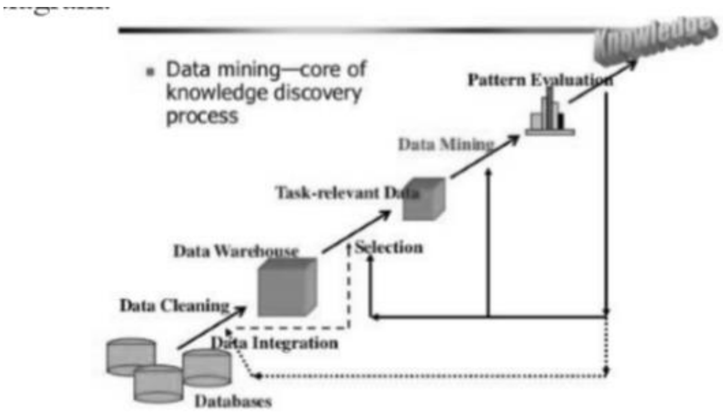
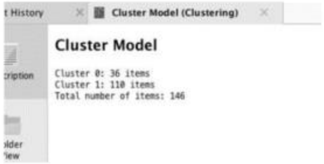


Diagram 2.4.1 shows the diagram of Modified KDD process. (Source from: Riah F. Elcullada-Encarnacion(2018))

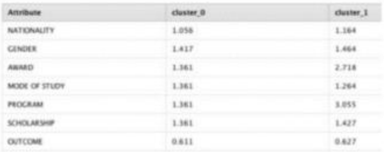
The algorithm of data mining used in this study is K-means clustering algorithm which is used to group student according to their similar attribute. The attribute consists of student background, achievements and academic courses.



Cluster Model

```
Cluster 0: 36 items
Cluster 1: 110 items
Total number of items: 146
```

Figure 5. Summary of Instances Result



Attribute	cluster_0	cluster_1
NATIONALITY	1.056	1.164
GENDER	1.417	1.464
AWARD	1.361	2.718
MODE OF STUDY	1.361	1.264
PROGRAM	1.361	3.055
SCHOLARSHIP	1.361	1.427
OUTCOME	0.611	0.627

Figure 6. The Final Cluster Centroids After K-means Clustering Process

Diagram 2.4.2 shows the output of K mean algorithms (Source from: Riah F. Elcullada-Encarnacion(2018))

From the result of cluster, the model for decision-making support is formed by analyzing statistically based on the result cluster.

The research shows that K-means clustering might be a good algorithm to help cluster up data and perform decision- making or recommendation. However, the it still require the analysis knowledge of human being, it would be the best if the whole process is automated including recommendation and decision support.

2.5 Teacher Assessment and Profiling using Fuzzy Rule based System and Apriori Algorithm

In this paper Atta-ur-Rahman (2013) design a system called Teacher Assessment and Profiling System (TAPS) which can be used to know more about a teacher’s strength and weakness so that the teacher can use it as reference for improvement.

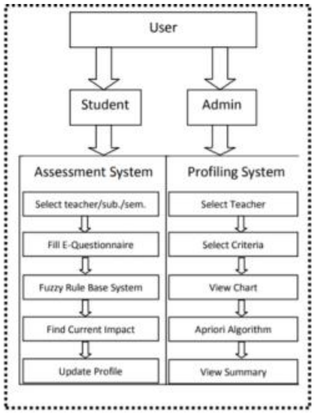


Diagram 2.5.1 shows the proposed system model (Source from: Atta-ur-Rahman (2013))

The diagram above is the flow diagram of the proposed system model where the user will be the student and they are able to go to the assessment system to fill up the questionnaire about the respective teacher they chose. Student feed back eventually will be feed into the fuzzy rule base system and profile the system based on the student quesntionnaire. All the profile will be store in the profiling system and user get to see the profile of the teacher they selected.

Beside that, Atta-ur-Rahman also proposed a method of using apriori algorithm to find out the relation between find association between teacher and subject, teacher and class and more.

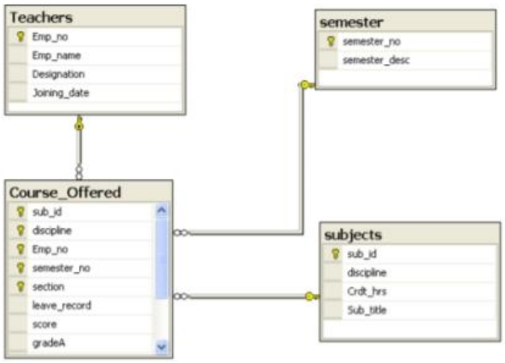
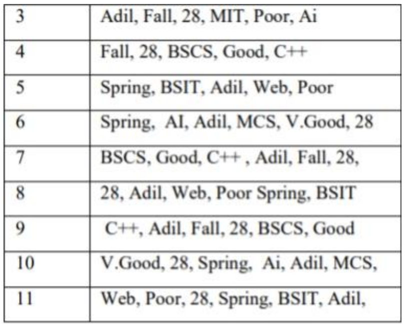


Diagram 2.5.2 shows the schema that will be used for apriori algorithm(Source from: Atta-ur-Rahman (2013))



3	Adil, Fall, 28, MIT, Poor, Ai
4	Fall, 28, BSCS, Good, C++
5	Spring, BSIT, Adil, Web, Poor
6	Spring, AI, Adil, MCS, V.Good, 28
7	BSCS, Good, C++ , Adil, Fall, 28,
8	28, Adil, Web, Poor Spring, BSIT
9	C++, Adil, Fall, 28, BSCS, Good
10	V.Good, 28, Spring, Ai, Adil, MCS,
11	Web, Poor, 28, Spring, BSIT, Adil,

Diagram 2.5.3 shows the result generated by the apriori algorithm

Above diagram shows the associations generated by the apriori algorithm.

From this research we can know that apropri method is a good way of finding association between different data and the relationships between them. However, some using apropri algorithm need a lot time especially when it need to produce large number of candidate sets.

2.6 A hybrid method based on MLFS approach to analyze students' academic achievement

Wei-Xiang Liu and Ching-Hsue Cheng (2016) had proposed a method of extracting and verifying critical feature by using combination of Machine Learning Feature Selection (MLFS) and Support Vector Machine (SVM). They first use MLFS to extract key features then use SVM to verify its accuracy. The whole process look like this:

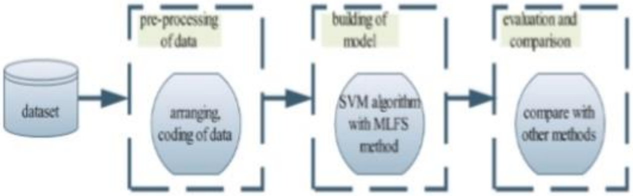
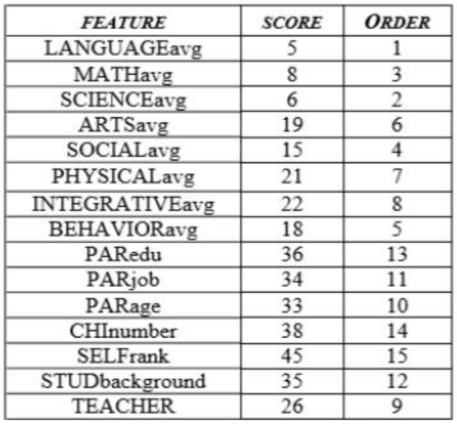

dataset
pre-processing of data
arranging, coding of data
building of model
SVM algorithm with MLFS method
evaluation and comparison
compare with other methods

Figure 2. Mode of research

Diagram 2.6.1 shows the 3 phase of this research (Source from: Wei-Xiang Liu and Ching-Hsue Cheng (2016))

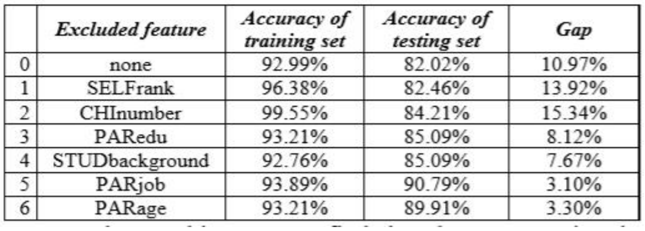
In this research they collected about 1300 student’s data in 40 classes for 1-6 grade. Then MLFS is used to be informed about importance of each feature by calculating the total ranking score of all defined feature.



FEATURE	SCORE	ORDER
LANGUAGEavg	5	1
MATHavg	8	3
SCIENCEavg	6	2
ARTSavg	19	6
SOCIALavg	15	4
PHYSICALavg	21	7
INTEGRATIVEavg	22	8
BEHAVIORavg	18	5
PARedu	36	13
PARjob	34	11
PARage	33	10
CHInumber	38	14
SELFrank	45	15
STUDbackground	35	12
TEACHER	26	9

Diagram 2.6.2 shows sum up ranking score of all feature (Source from: Wei-Xiang Liu and Ching-Hsue Cheng (2016))

According to Wei-Xiang Liu and Ching-Hsue Cheng (2016) the smaller the number of score the greater the influence, hence they eliminate feature with higher score one by one with SVM algorithm as model, then perform accuracy calculation each time a low influence feature is remove. The result under SVM algorithm is shown below.

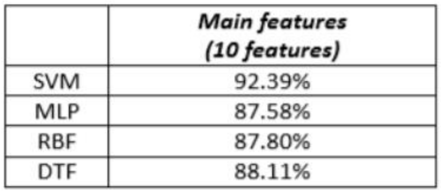


	Excluded feature	Accuracy of training set	Accuracy of testing set	Gap
0	none	92.99%	82.02%	10.97%
1	SELFrank	96.38%	82.46%	13.92%
2	CHInumber	99.55%	84.21%	15.34%
3	PARedu	93.21%	85.09%	8.12%
4	STUDbackground	92.76%	85.09%	7.67%
5	PARjob	93.89%	90.79%	3.10%
6	PARage	93.21%	89.91%	3.30%

Diagram 2.6.3 shows accuracy of deleting low influence feature gradually. (Source from: Wei-Xiang Liu and Ching-Hsue Cheng (2016))

Based on above table, the accuracy looks better and better when more and more low influence feature is removed.

The whole process is then repeated by using other classification methods to compare with SVM classification. The result is shown at Diagram 2.6.4.



	Main features (10 features)
SVM	92.39%
MLP	87.58%
RBF	87.80%
DTF	88.11%

Diagram 2.6.4 shows average accuracy of SVM vs other types of classification algorithm (Source from: Wei-Xiang Liu and Ching-Hsue Cheng (2016))

As state by Wei-Xiang Liu and Ching-Hsue Cheng (2016) this hybrid method with MLFS extract key features for enhancing accuracy and SVM algorithm classification came up with better quality and effectiveness.

2.7 Prediction of personality first impression with deep Bimodal LSTM

Karen Yang and Nao Glaser (2017) had done research on proposing a Depp Bimodal Regression LSTM model that extracts feature from video in term of temporally ordered visual and audio to predict person's first impression on five personality traits. The personality traits consist of openness, conscientiousness, extroversion, agreeableness and neuroticism (OCEAN). They first use a ResNet34 model to extract visual features from randomly selected video frame. ResNet34 model is a fine-tuned 34 layers residual network pretrained on ImageNet. Then Bi-Modal LSTM model will be used to encodes both visual and audio modalities with temporal modeling.

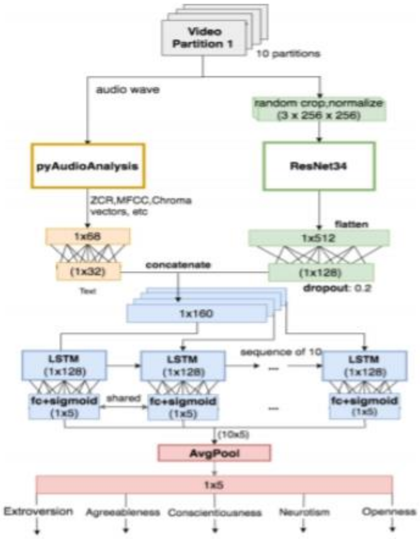
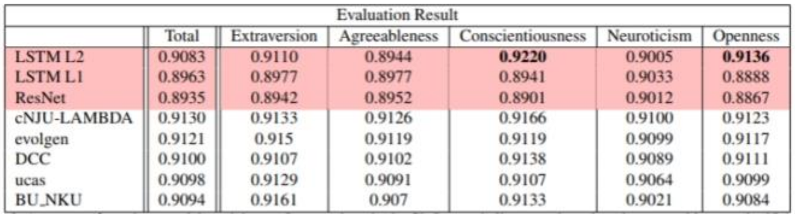


Diagram 2.7.1 shows the Deep Bi-modal LSTM model architecture (Source from: Karen Yang and Nao Glaser (2017))

Evaluation between LSTM and others competitor is tested, the result is show below.



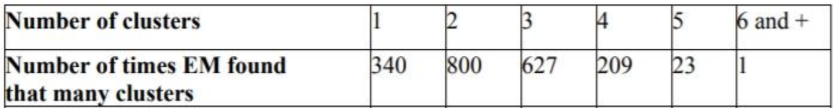
Evaluation Result						
	Total	Extraversion	Agreeableness	Conscientiousness	Neuroticism	Openness
LSTM L2	0.9083	0.9110	0.8944	**0.9220**	0.9005	**0.9136**
LSTM L1	0.8963	0.8977	0.8977	0.8941	0.9033	0.8888
ResNet	0.8935	0.8942	0.8952	0.8901	0.9012	0.8867
cNJU-LAMBDA	0.9130	0.9133	0.9126	0.9166	0.9100	0.9123
evolgen	0.9121	0.915	0.9119	0.9119	0.9099	0.9117
DCC	0.9100	0.9107	0.9102	0.9138	0.9089	0.9111
ucas	0.9098	0.9129	0.9091	0.9107	0.9064	0.9099
BU_NKU	0.9094	0.9161	0.907	0.9133	0.9021	0.9084

Diagram 2.7.2 shows the accuracy between 3 model used in this research and others competitors. architecture (Source from: Karen Yang and Nao Glaser (2017))

According to Karen Yang and Nao Glaser (2017) it is proved that LSTM is a good model for prediction of first impressions of apparent personality. The model also suggest that face is the most significant factor in this prediction as face of subject in the video influence the most.

2.8 Clustering and Profiling Students According to their interactions with an Intelligent tutoring System Fostering Self-regulated system.

In this research, J.M.Harley, G.J.Trevors, et.al (2013) use clustering algorithm to cluster on student who learn from intelligent tutoring system. All of this cluster of student will then be characterize into 3 distinct profile of student. The clustering algorithm they chose to use in this research is Expectation-Maximization (EM) algorithm.



Number of clusters	1	2	3	4	5	6 and +
Number of times EM found that many clusters	340	800	627	209	23	1

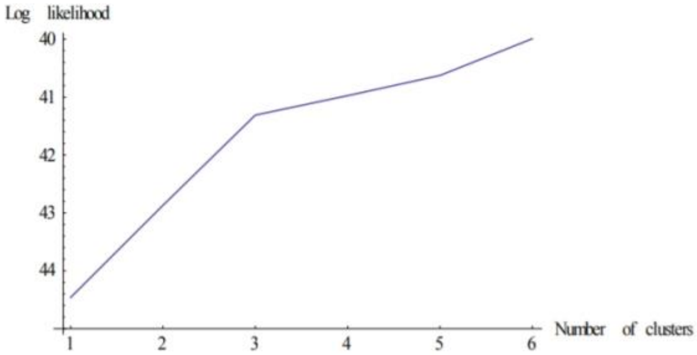
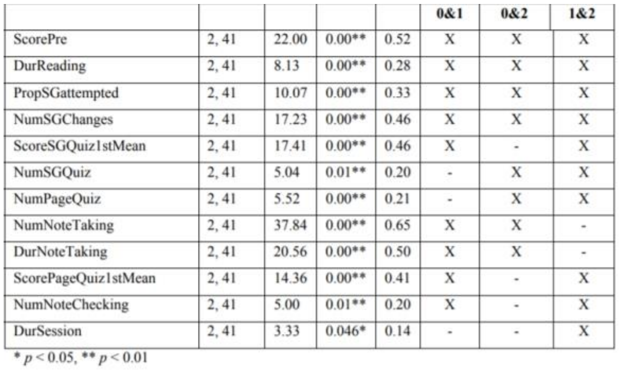


Diagram 2.8.1 shows the result of cluster applying EM algorithm with 2000 different seeds and a mean log-likelihood associated to each clustering when applying EM algorithm with 2000 different initial seeds (Source from: J.M.Harley, G.J.Trevors, et.al (2013))

For the cluster characterization and profile part, they wanted to characterize each cluster from the dominant partition with 3 clusters, so they use Multivariate statistics (MANOVA) to do it and displaying statistically differences of 12 variables used for cluster formation.

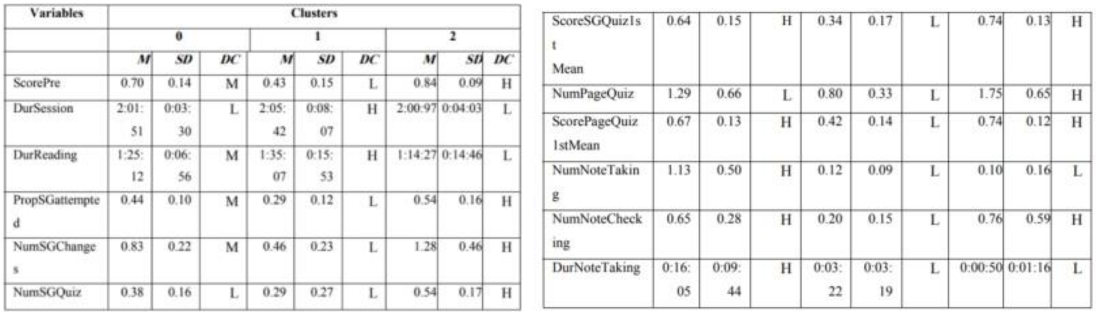


					0&1	0&2	1&2
ScorePre	2, 41	22.00	0.00**	0.52	X	X	X
DurReading	2, 41	8.13	0.00**	0.28	X	X	X
PropSGattempted	2, 41	10.07	0.00**	0.33	X	X	X
NumSGChanges	2, 41	17.23	0.00**	0.46	X	X	X
ScoreSGQuiz1stMean	2, 41	17.41	0.00**	0.46	X	-	X
NumSGQuiz	2, 41	5.04	0.01**	0.20	-	X	X
NumPageQuiz	2, 41	5.52	0.00**	0.21	-	X	X
NumNoteTaking	2, 41	37.84	0.00**	0.65	X	X	-
DurNoteTaking	2, 41	20.56	0.00**	0.50	X	X	-
ScorePageQuiz1stMean	2, 41	14.36	0.00**	0.41	X	-	X
NumNoteChecking	2, 41	5.00	0.01**	0.20	X	-	X
DurSession	2, 41	3.33	0.046*	0.14	-	-	X

* $p < 0.05$, ** $p < 0.01$

Diagram 2.8.2 shows the result of MANOVA for the 12 variables used in cluster formation and the pairwise differences for the 3 cluster. (Source from: J.M.Harley, G.J.Trevors, et.al (2013))

In order to profiles the cluster, they find the means and the standard deviations of each dependent variables to create profile for each cluster. Dummy code for mean value of High, Medium and Low is used to characterize different between cluster. With 3 cluster profile is created which help them to distinguish different type of learners based on the 12 variables.



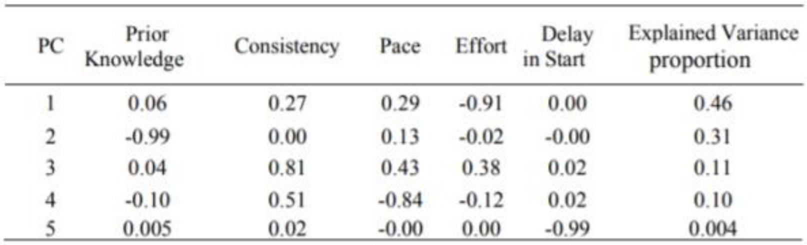
Variables	Clusters								
	0			1			2		
	M	*SD*	*DC*	*M*	*SD*	*DC*	*M*	*SD*	*DC*
ScorePre	0.70	0.14	M	0.43	0.15	L	0.84	0.09	H
DurSession	2:01:51	0:03:30	L	2:05:42	0:08:07	H	2:00:97	0:04:03	L
DurReading	1:25:12	0:06:56	M	1:35:07	0:15:53	H	1:14:27	0:14:46	L
PropSGattempted	0.44	0.10	M	0.29	0.12	L	0.54	0.16	H
NumSGChanges	0.83	0.22	M	0.46	0.23	L	1.28	0.46	H
NumSGQuiz	0.38	0.16	L	0.29	0.27	L	0.54	0.17	H
ScoreSGQuiz1st Mean	0.64	0.15	H	0.34	0.17	L	0.74	0.13	H
NumPageQuiz	1.29	0.66	L	0.80	0.33	L	1.75	0.65	H
ScorePageQuiz 1stMean	0.67	0.13	H	0.42	0.14	L	0.74	0.12	H
NumNoteTaking	1.13	0.50	H	0.12	0.09	L	0.10	0.16	L
NumNoteChecking	0.65	0.28	H	0.20	0.15	L	0.76	0.59	H
DurNoteTaking	0:16:05	0:09:44	H	0:03:22	0:03:19	L	0:00:50	0:01:16	L

Diagram 2.8.3 shows the table of 3 clustered with their respective mean and standard deviation on each variables and the dummy coded value. (Source from: J.M.Harley, G.J.Trevors, et.al (2013))

2.9 Data-driven learner profiling based on clustering student behaviors: Learning consistency, pace and effort

This research is done by S.Mojarad, A.Essa, et.al (2018), where they try to identify student with same academic and behavior characteristics. The method proposed in this paper is they collect data from a Web-based, adaptive assessment and learning system call ALEKS. All the student's are collected from ALEKS where they use this learning system to learn and student are groups into 6 cluster by using 6 defined characteristics. The 6 attributes are Prior Knowledge, consistency, Pace, Effort, Delay in Start, explain variance proportion.

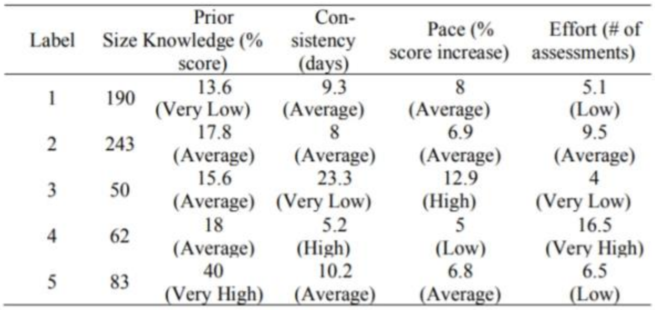
They first use PCA (Principal Component Analysis) to analysis the correlation between all these attributes.



PC	Prior Knowledge	Consistency	Pace	Effort	Delay in Start	Explained Variance proportion
1	0.06	0.27	0.29	-0.91	0.00	0.46
2	-0.99	0.00	0.13	-0.02	-0.00	0.31
3	0.04	0.81	0.43	0.38	0.02	0.11
4	-0.10	0.51	-0.84	-0.12	0.02	0.10
5	0.005	0.02	-0.00	0.00	-0.99	0.004

Diagram 2.9.2 shows the PCA of different attributes. (Source from: S.Mojarad, A.Essa, et.al (2018))

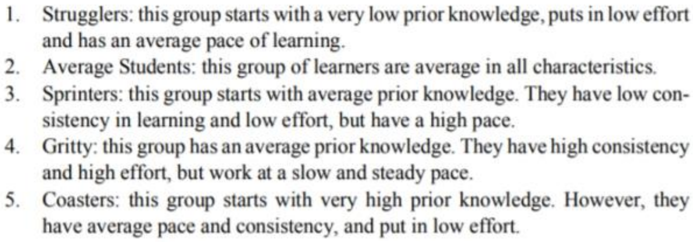
Then they use Mean shift clustering to identify 5 possible cluster in the data and set K-means clustering to find 5 different group of students. They find the average of attributes for each cluster in order to interpret the cluster.



Label	Size	Prior Knowledge (% score)	Con-sistency (days)	Pace (% score increase)	Effort (# of assessments)
1	190	13.6 (Very Low)	9.3 (Average)	8 (Average)	5.1 (Low)
2	243	17.8 (Average)	8 (Average)	6.9 (Average)	9.5 (Average)
3	50	15.6 (Average)	23.3 (Very Low)	12.9 (High)	4 (Very Low)
4	62	18 (Average)	5.2 (High)	5 (Low)	16.5 (Very High)
5	83	40 (Very High)	10.2 (Average)	6.8 (Average)	6.5 (Low)

Diagram 2.9.2 shows average attribute of 5 cluster. (Source from: S.Mojarad, A.Essa, et.al (2018))

After getting the cluster, they then define name for each cluster which best describe each cluster.



1. Strugglers: this group starts with a very low prior knowledge, puts in low effort and has an average pace of learning.
2. Average Students: this group of learners are average in all characteristics.
3. Sprinters: this group starts with average prior knowledge. They have low consistency in learning and low effort, but have a high pace.
4. Gritty: this group has an average prior knowledge. They have high consistency and high effort, but work at a slow and steady pace.
5. Coasters: this group starts with very high prior knowledge. However, they have average pace and consistency, and put in low effort.

Diagram 2.9.3 shows the name for each attribute. (Source from: S.Mojarad, A.Essa, et.al (2018))

Chapter 3: System Design

3.1 Overview

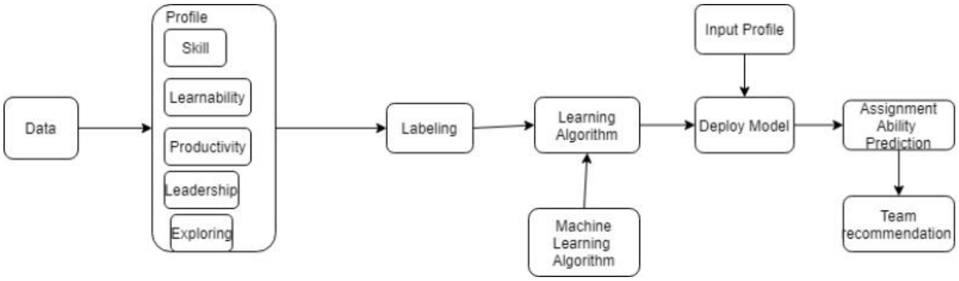


Diagram 3.1.1 show the overall system design of Team recommendation system

As you can see from diagram 3.1.1, the whole flow start with collecting data from various to perform profiling of student with defined attributes. All the profile will then go through labeling to label all the profile with respective assignment ability. The whole labeled profile will then go through supervised learning where the labeled datasets will be fit into a machine learning algorithm to start training and eventually come out with a model. This model then will be deployed and used for the application which is to perform team recommendation. The team recommendation is performed based on grouping a group of student together in a team according to their predicted student's role performance of completing an assignment.

3.2 Profiling

Below are the defined attributes which used to define a student personality.

Attributes	**Description**
Domain Skillsets	Any skills that an individual possess
Learnability	Ability to learn new stuff
Exploring	Initiative to explore and learn new stuff
Productivity	Rate of task accomplishment
Leadership	Ability to lead group of people in completing task

Table 3.1.1 Shows set of attribute that determine a student characteristics

The reason these attributes is used in this paper is these few attributes are suitable to be reviewed in educational aspect. These attributes will be calculated in terms of numerical value and the datasets collected from respective individual determine these attribute scores. Each attribute has their own way of collecting the respective data to determine its attribute value. In this research, UTAR will be the experimental ground for the research, and all these data sources are from UTAR as well.

For the domain skillsets attribute, the data collected will be student academic transcript which recorded grade for every course or subject student had took, the grade of the subject determine the skill that student possess. For example, if the student scored an A in Programming Concept, it is assume that student posses a skill in programming.

Learnability's data will be collected from student's academic transcript as well. Student's subject grade consistency will determine the value of learnability. According to UTAR program structure, there are subjects which are co-related to each other, for example, Analysis and Design of Information system subject is pre-requisite for the subject Object Oriented System Analysis and Design, which mean that both of this subject is related. Student's Grade on both of this subject will be the determinant of the value of learnability attribute.

Exploring attribute, the data is collected by capturing packet sent by student when surfing the internet. This is to know what website student normally visit, educational or not educational, which determine the exploring attribute.

Leadership attribute, the data is collected by accessing student UTAR portal to check on student's USSDC report. USSDC stand for Utar Soft Skills Development Certificate Programme, which is to recognize students' achievement and efforts to improve themselves in areas of communication, leadership, teamwork, strategy thinking, creativity and commitment to integrity. All of these involve student taking part of activities or event that is outside of the classroom. However for leadership attribute, we will be only focusing in leadership component of USSDC report. The data will be the leadership point that a student earn in university.

Productivity, the data will be collected from student accomplished assignment. Three important criteria will be used as the datasets needed to calculate student productivity, which are assignment grade, assignment due date and assignment submission date. All these component will then be used to calculate the productivity of a student in completing assignment.

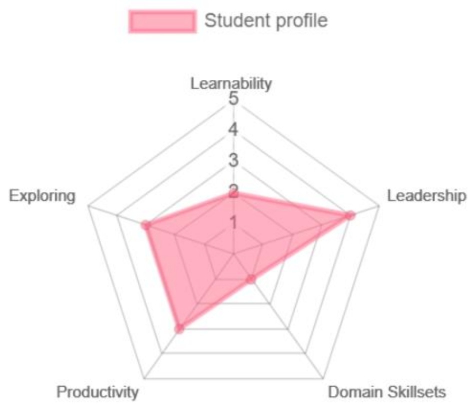


Diagram 3.1.2 shows the profile visualization of an individuals student

Each attribute of a student profile will be range from 1 to 5 which represent the rating of player characteristics in each attribute. Below is the method of how to get the rating of each attributes.

<u>Learnability</u>

Grade value of each subject

Grade ‘A+’:5.0

Grade ‘A’: 4.5

Grade ‘A-‘: 4.0

Grade ‘B+’: 3.5

Grade ‘B’ : 3

Grade ‘B-‘: 2.5

Grade ‘C+’: 2.0

Grade ‘C’: 1.0

Subject relation according to UTAR program structure of Bachelor Degree Computer Science

1) Programming Concept ≺ Data structure and Algorithm Problem Solving ≺

Ai Technique ≺ Deep learning for data science

2) Programming Concept ≺ Website Design ≺ Server-Side Web Applications Development
3) Programming Concept ≺ Data structure and Algorithm Problem Solving ≺ Parallel Programming
4) Programming Concept ≺ Data structure and Algorithm Problem Solving ≺ Fundamental of system programming
5) Programming Concept ≺ Object Oriented Programming Practices ≺ Graphic Programming for Mobile Platform
6) Analysis and Design of Information system ≺ Object Oriented System Analysis and Design ≺ Requirement Engineering
7) Basic Algebra, Advance calculus and application ≺ Numerical Method
8) Introduction to calculus and application ≺ Advance Calculus and application
9) Probability and Statistics for Programming ≺ Data Science
10) Data Communication and Networking ≺ Internetworking principle and practices
11) Database Development ≺ Database System
12) Fundamental Information Security ≺ Computer System and Network Security
13) Operating System ≺ Distributed Computer System
14) Software Engineering ≺ Software Testing

Learnability is all about consistency of the learning progress, so in this research, in order to obtain learnability rating of an individual, the grade of the related subject will be focus. The calculation part will be involving calculating the grade value difference between the related subject according to the list of subject relation rules shown above.

$$\text{Difference} = \underbrace{\text{grade of subject 1} - \text{grade of subject 2} - \text{grade of subject 3}}_{\text{Related subject according to subject relation rules}}$$

The difference of the grade will represent the consistency of the student learning.

Then average up all the differences between different field of subject, and the average determine the rating of learnability.

Below is the list of how rating is given in range of 1.0 to 5.0.

1.0: Average difference >0 and <1

2.0: Average difference >1 and <2

3.0: Average difference >2 and <3

4.0: Average difference >3 and <4

5.0: Average difference >4 and <5

<u>Exploring</u>

Exploring attribute is about initiative of an individuals to learn new things. As mentioned above, in this research the way to determine the rating of the exploring attribute will based on tracking websites that students normally visited at free time. Each website visited will be classified into 2 types which are education and non-education. Then based on all the websites collected, calculation will be performed to determine the rating of exploring attribute value. Below are the method.

The rules to determine the weight value for each educational website, according to the time spent on the website.

1.0 weight value :< 5 minutes (< 300s)

2.0 weight value :> 5 minutes && < 10 minutes (> 300s && <600s)

3.0 weight value :> 15 minute (> 900s)

Average weight value

$$\left(\frac{Sum\ of\ weight\ value\ of\ each\ educational\ website}{Total\ educational\ website\ visited\ \times 3(maximum\ weight\ value)}\right)$$

Percentage of education website among all website visited

$$\frac{Total\ educational\ website\ visited}{Total\ website\ visited}$$

$$\boldsymbol{Exploring\ Attribute\ Rating} \\ = \left(\frac{\boldsymbol{Average\ weight\ value + educational\ website\ per\ overall\ visited\ website}}{2}\right) \\ \times 5$$

0.0-1.0: not so exploring

1.0-2.0: slightly exploring

2.0-3.0: moderately exploring

3.0-4.0: exploring

4.0-5.0: very exploring

<u>Productivity</u>

Productivity attribute rating will be determined through student’s assignment start date, submission date and grade of the assignment to calculate the rate of accomplishment of certain task.

Number of days used to complete an assignment = Assignment submission date - Assignment start date

Then use the number of days and grade of each assignment to obtain the ratio of productivity.

$$Productivity\ Ratio = {}^{Assignment\ score}/_{Number\ of\ days}$$

With the productivity Ratio we can know the productivity rate of student completing assignment, however we do not know how well is the ratio, so we will use statistics method to determine how well is the ratio by finding mean and standard variation of productivity Ratio.

$$Mean = \frac{\sum Productivity\ ratio}{n}$$

$$Standard\ Deviation = \sqrt{\frac{\sum_{i=1}^{n}(x_i - mean)^2}{n}}$$

Below are the rules to determine the productivity value range from 1.0 to 5.0

1.0: productivity ratio < mean/2

2.0: productivity ratio < mean and >mean/2

3.0: productivity ratio = mean

4.0: productivity ratio >= mean and < standard deviation

5.0: productivity ratio >= standard deviation

The final productivity attribute rating will be the average of all productivity value obtained.

Domain Skillsets

To obtain domain skillsets of a student, the method proposed in this research is determine it by getting the grade of the subject. There are 6 domain field defined in this paper which are:

1. Programming
2. Networking
3. Software
4. Database
5. Artificial Intelligent

6. Theory, logic and design

In this paper, the subject from Bachelor Degree of Computer Science in UTAR will be used as the example. All the subject of Computer Science are being classified in to these 7 domain. Below are the few list of Classification of some subject in UTAR. All of this classification are based on ACM Computing classification system.

Domain Skillsets	**Keywords**
Programming	Introduction to Programming
Programming	Object-oriented programming
Programming	Website Development
Programming	Data Structure
Programming	Parallel programming languages
Networking	Internetworking
Networking	Information Security
Software	Software Testing
Software	Software infrastructure
Software	Analysis and Design
Database	Database Development
Database	Database System
Artificial Intelligent	Data Science
Artificial Intelligent	Deep Learning
Artificial Intelligent	AI Technique
Theory, Logic and Design	Introductory Discrete Mathematics for Computer Science
Theory, Logic and Design	Introduction to Computer Architecture
Theory, Logic and Design	Algorithm Analysis

Average grade of all the subject in these 7 domain will be calculated, and come out with average grade value of each domain. Below are the Grade value of subject.

Grade ‘A+’:5.0

Grade ‘A’: 4.5

Grade ‘A-‘: 4.0

Grade ‘B+’: 3.5

Grade ‘B’ : 3

Grade ‘B-‘: 2.5

Grade ‘C+’: 2.0

Grade ‘C’: 1.0

The final Domain Skillsets attribute rating will be the average value of each domain averaged grade value.

Each domain value = Mean (Grade of subjects)

Domain Skillsets attribute value = Mean (Each domain value)

Leadership

The leadership attribute value is obtained by accessing student USSDC portal and retrieve the total point in leaderships component. Below is the list of how rating of the leadership attribute value is defined.

1.0: total point >5 and <=10

2.0: total point >10 and <=20

3.0: total point >20 and <=30

4.0: total point >30 and <=40

5.0: total point >40

3.3 Assignment team roles

There are 4 different roles we defined in this research. These 4 roles will be the main criteria to look at when performing team matching recommendation.

Roles	**Description**
Planning	Leader who plan and distribute job scope on completing an assignment task.
Idea	Provide Idea in problem solving
Technical	Good in technical part of the task for example coding
Writing	Writing reports, drawing flowcharts and others

Table 3.2.1 Shows the different assignment roles defined in this paper.

3.4 Implementation Issues and challenge

There are few limitation found in this research as there is only limited amount of time to complete the research. The result created in this research might not be accurate and maybe have some flaws in it. Below are the limitation found.

<u>Collecting data</u>

The challenge faced in this research is collecting data itself. Each characteristics attributes are described differently which mean the data needed to be collected are all difference and serve with different purpose. From defining which data to be collected and where should we collect it from are all challenge found in this research.

<u>Defining method to determine attributes rating</u>

After data for an attribute have finished collected, how to deal with the data is another problem faced in this research. As stated above, one of the objective is to define and profile individuals with different attribute ratings, and the data collected are to serve for this purpose which is to determine to attribute ratings from it. Thus, a lot of mathematical method, formula, rules have be defined in order to determine the attribute ratings, however all of the method, rules are being defined and created in this research might not necessary correct or accurate as all of these are just based on ideas we had. No indicator or prove can be used to shows the method is correct.

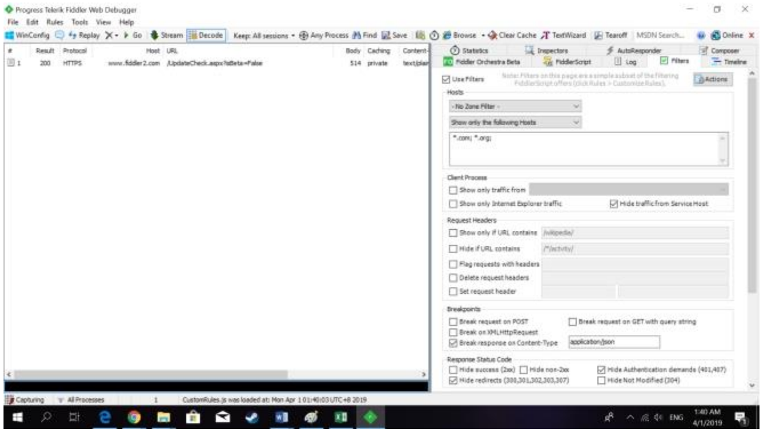
Chapter 4: Methodology

As discussed in chapter 3 system design, on what data to collect in order to determine the rating of 5 attributes, now this is the is the part where we discuss how to collect the data, what technique is used and tools that being used.

4.1 Exploring Attribute

As mention before, exploring attribute is all about initiative of an individuals to learn new stuff. Tracing websites visited by student might able to let us know what website that student usually visit, is that educational website? If it is that student might have exploring traits by using free time to explore the internet for new knowledge.

In order to trace website visited, packet tracer is one of the ways to do so. By capturing the packet and obtain the URL address and google searched word from the packets. In this paper, a software called Fiddler is used for capturing packets. Fiddler is a server application that debug HTTP. The reason Fiddler is chosen as software to capture packet is because recently most webpage had already used HTTPS secure protocol and all the information captured from this protocol is encrypted, however fiddler able to capture HTTPS packet and decrypt the packet so user can view the information in the packets. Diagram below shows interface of fiddler.


Progress Telerik Fiddler Web Debugger
File Edit Rules Tools View Help
WinConfig Replay Go Stream Decode Keep: All sessions Any Process Find Save Browse Clear Cache TextWizard Tearoff MSDN Search... Online
Statistics Inspectors AutoResponder Composer
Fiddler Orchestra Beta FiddlerScript Log Filters Timeline
Actions
ENG 1:40 AM 4/1/2019

#	Result	Protocol	Host	URL	Body	Caching	Content-
1	200	HTTPS	www.fiddler2.com	/UpdateCheck.aspx?isBeta=False	514	private	text/plain

☑ Use Filters

Note: Filters on this page are a simple subset of the filtering FiddlerScript offers (click Rules > Customize Rules).

Hosts

- No Zone Filter -

Show only the following Hosts

*.com; *.org;

Client Process

☐ Show only traffic from

☐ Show only Internet Explorer traffic ☑ Hide traffic from Service Host

Request Headers

☐ Show only if URL contains /wikipedia/

☐ Hide if URL contains /*/activity/

☐ Flag requests with headers

☐ Delete request headers

☐ Set request header

Breakpoints

☐ Break request on POST ☐ Break request on GET with query string

☐ Break on XMLHttpRequest

☑ Break response on Content-Type application/json

Response Status Code

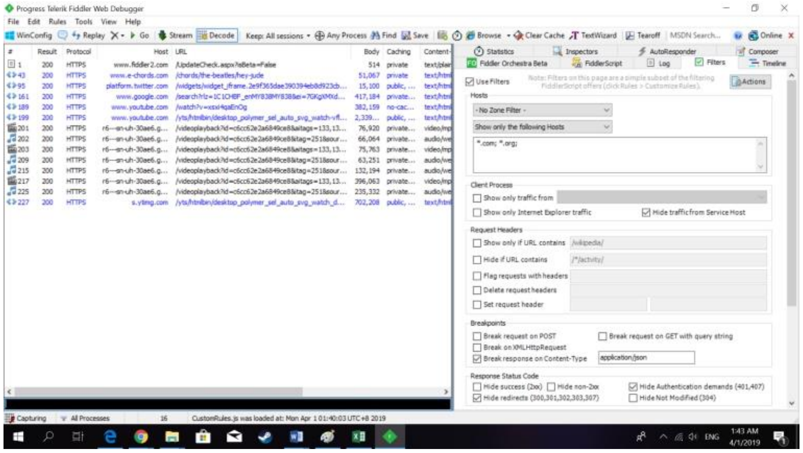
☐ Hide success (2xx) ☐ Hide non-2xx ☑ Hide Authentication demands (401,407)

☑ Hide redirects (300,301,302,303,307) ☐ Hide Not Modified (304)

Capturing | All Processes | 1 | CustomRules.js was loaded at: Mon Apr 1 01:40:03 UTC+8 2019

Diagram 4.1.1 Shows interface of fiddler.

Fiddler interface consists of a filter function which let user to filter out desired packets. In this case, only HTTPS protocol packets is filtered out by using the setting function.


Progress Telerik Fiddler Web Debugger
File Edit Rules Tools View Help
WinConfig Replay Go Stream Decode Keep: All sessions Any Process Find Save Browse Clear Cache TextWizard Tearoff MSDN Search... Online
Statistics Inspectors AutoResponder Composer
Fiddler Orchestra Beta FiddlerScript Log Filters Timeline
Actions

#	Result	Protocol	Host	URL	Body	Caching	Content-
1	200	HTTPS	www.fiddler2.com	/UpdateCheck.aspx?isBeta=False	514	private	text/plai
43	200	HTTPS	www.e-chords.com	/chords/the-beatles/hey-jude	51,067	private	text/html
95	200	HTTPS	platform.twitter.com	/widgets/widget_iframe.2e9f365dae390394eb8d923cb...	15,100	public, ...	text/html
161	200	HTTPS	www.google.com	/search?rlz=1C1CHBF_enMY838MY838&ei=7GKgXMXd...	417,184	private...	text/html
189	200	HTTPS	www.youtube.com	/watch?v=xsxl4qaEnOg	382,159	no-cac...	text/html
199	200	HTTPS	www.youtube.com	/yts/htmlbin/desktop_polymer_sel_auto_svg_watch-vfl...	2,339...	public, ...	text/html
201	200	HTTPS	r6---sn-uh-30ae6.g...	/videoplayback?id=c6cc62e2a6849ce8&aitags=133,13...	76,920	private...	video/mp
202	200	HTTPS	r6---sn-uh-30ae6.g...	/videoplayback?id=c6cc62e2a6849ce8&itag=251&sour...	66,064	private...	audio/we
203	200	HTTPS	r6---sn-uh-30ae6.g...	/videoplayback?id=c6cc62e2a6849ce8&aitags=133,13...	75,763	private...	video/mp
209	200	HTTPS	r6---sn-uh-30ae6.g...	/videoplayback?id=c6cc62e2a6849ce8&itag=251&sour...	63,251	private...	audio/we
215	200	HTTPS	r6---sn-uh-30ae6.g...	/videoplayback?id=c6cc62e2a6849ce8&itag=251&sour...	132,194	private...	audio/we
217	200	HTTPS	r6---sn-uh-30ae6.g...	/videoplayback?id=c6cc62e2a6849ce8&aitags=133,13...	396,063	private...	video/mp
225	200	HTTPS	r6---sn-uh-30ae6.g...	/videoplayback?id=c6cc62e2a6849ce8&itag=251&sour...	235,332	private...	audio/we
227	200	HTTPS	s.ytimg.com	/yts/htmlbin/desktop_polymer_sel_auto_svg_watch_d...	702,208	public, ...	text/html

☑ Use Filters

Note: Filters on this page are a simple subset of the filtering FiddlerScript offers (click Rules > Customize Rules).

Hosts

- No Zone Filter -

Show only the following Hosts

*.com; *.org;

Client Process

☐ Show only traffic from

☐ Show only Internet Explorer traffic ☑ Hide traffic from Service Host

Request Headers

☐ Show only if URL contains /wikipedia/

☐ Hide if URL contains /*/activity/

☐ Flag requests with headers

☐ Delete request headers

☐ Set request header

Breakpoints

☐ Break request on POST ☐ Break request on GET with query string

☐ Break on XMLHttpRequest

☑ Break response on Content-Type application/json

Response Status Code

☐ Hide success (2xx) ☐ Hide non-2xx ☑ Hide Authentication demands (401,407)

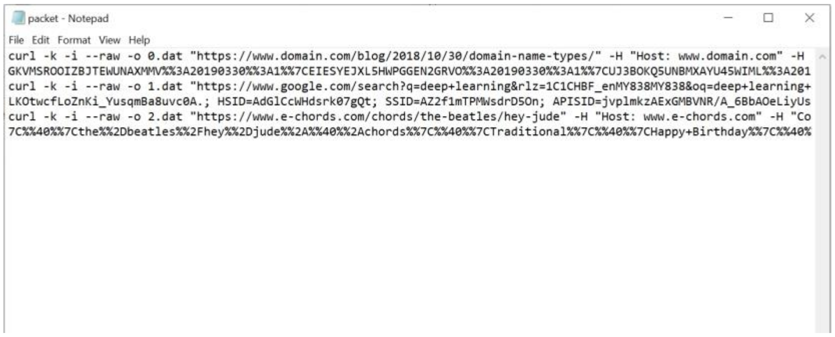
☑ Hide redirects (300,301,302,303,307) ☐ Hide Not Modified (304)

Capturing All Processes 16 CustomRules.js was loaded at: Mon Apr 1 01:40:03 UTC+8 2019

1:43 AM 4/1/2019

Diagram 4..1.2 Shows Fiddler had capture few packets with filter setting on

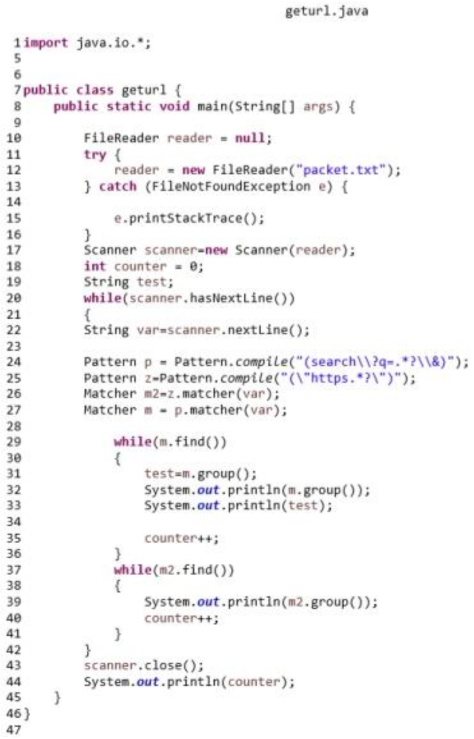
The packets are then being export and diagram below show a exported packet.



```
curl -k -i --raw -o 0.dat "https://www.domain.com/blog/2018/10/30/domain-name-types/" -H "Host: www.domain.com" -H
GKVMSROOIZBJTEWUNAXMMV%%3A20190330%%3A1%%7CEIESYEJXL5HWPGGEN2GRVO%%3A20190330%%3A1%%7CUJ3BOKQ5UNBMXAYU45WIML%%3A201
curl -k -i --raw -o 1.dat "https://www.google.com/search?q=deep+learning&rlz=1C1CHBF_enMY838MY838&oq=deep+learning+
LKOtwcfLoZnKi_YusqmBa8uvc0A.; HSID=AdGlCcWHdsrk07gQt; SSID=AZ2f1mTPMWsdrD5On; APISID=jvplmkzAExGMBVNR/A_6BbAOeLiyUs
curl -k -i --raw -o 2.dat "https://www.e-chords.com/chords/the-beatles/hey-jude" -H "Host: www.e-chords.com" -H "Co
7C%%40%%7Cthe%%2Dbeatles%%2Fhey%%2Djude%%2A%%40%%2Achords%%7C%%40%%7CTraditional%%7C%%40%%7CHappy+Birthday%%7C%%40%
```

Diagram 4.1.3 shows the file of the export packets

As you can see at the above diagram, all packet’s information is exported out. But not all is what we need, we just need the URL of website visited and perhaps some Google searched word. Thus, a java program is written to filter out the URL link.

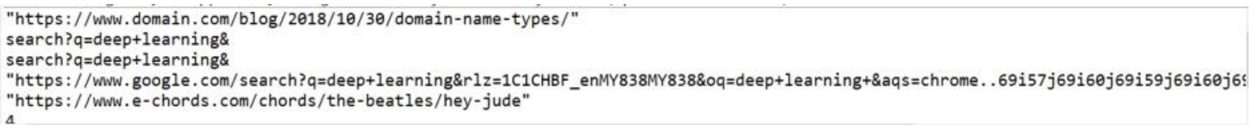


geturl.java

```java
import java.io.*;


public class geturl {
    public static void main(String[] args) {

        FileReader reader = null;
        try {
            reader = new FileReader("packet.txt");
        } catch (FileNotFoundException e) {

            e.printStackTrace();
        }
        Scanner scanner=new Scanner(reader);
        int counter = 0;
        String test;
        while(scanner.hasNextLine())
        {
        String var=scanner.nextLine();

        Pattern p = Pattern.compile("(search\\?q=.*?\\&)");
        Pattern z=Pattern.compile("(\"https.*?\")");
        Matcher m2=z.matcher(var);
        Matcher m = p.matcher(var);

            while(m.find())
            {
                test=m.group();
                System.out.println(m.group());
                System.out.println(test);

                counter++;
            }
            while(m2.find())
            {
                System.out.println(m2.group());
                counter++;
            }
        }
        scanner.close();
        System.out.println(counter);
    }
}

```

Diagram 4.1.4 Shows the java code of filtering URL out of

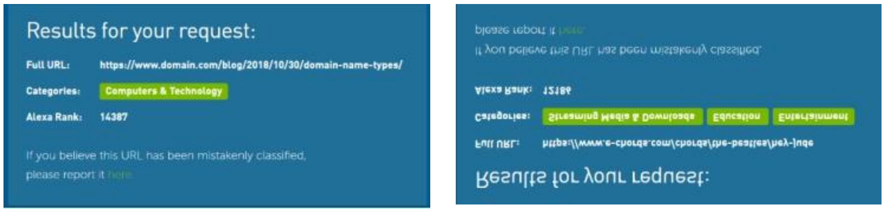
The URL obtained from the packet and Google searched information is shown below.



```
"https://www.domain.com/blog/2018/10/30/domain-name-types/"
search?q=deep+learning&
search?q=deep+learning&
"https://www.google.com/search?q=deep+learning&rlz=1C1CHBF_enMY838MY838&oq=deep+learning+&aqs=chrome..69i57j69i60j69i59j69i60j6
"https://www.e-chords.com/chords/the-beatles/hey-jude"
4
```

Diagram 4.1.5 shows the result of debugging the code.

Now that we had traced the packets, based on the data collected, classification can be performed. For now, classification is performed sorely using online URL category check.



Results for your request:

Full URL: https://www.domain.com/blog/2018/10/30/domain-name-types/

Categories: Computers & Technology

Alexa Rank: 14387

If you believe this URL has been mistakenly classified, please report it here.

Results for your request:

Full URL: https://www.e-chords.com/chords/the-beatles/hey-jude

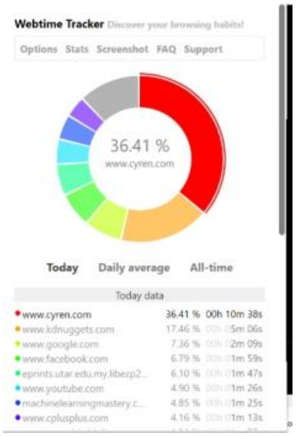
Categories: Streaming Media & Downloads Education Entertainment

Alexa Rank: 12184

If you believe this URL has been mistakenly classified, please report it here.

Diagram 4.1.6 shows the result of online URL classification

The Website which category fall under “Education” and “Technology” both will considered as educational websites, since the research will be taking student from faculty of FICT (faculty of information and communication technology) website under category “Technology” should be their study material as well. Time spent on a particular website will also be determined as weighted value. The time spent of each website will be tracked using a google extension called WebTime tracker.



www.cyren.com	36.41 %	00h 10m 38s
www.kdnuggets.com	17.46 %	00h 05m 06s
www.google.com	7.36 %	00h 02m 09s
www.facebook.com	6.79 %	00h 01m 59s
eprints.utar.edu.my.liberp2...	6.10 %	00h 01m 47s
www.youtube.com	4.90 %	00h 01m 26s
machinelearningmastery.c...	4.85 %	00h 01m 25s
www.cplusplus.com	4.16 %	00h 01m 13s

Diagram 4.1.7 shows Webtime track by google.

Based on the classified result generated, calculation can be performed to determine attribute rating of exploring.

I. First calculate the average weight value of each classified educational website.

II. Then determine how many website is considered as education among all visited website in terms of percentage.

III. Calculated the score by averaging value determined between percentage of education website and average weight value

IV. Lastly multiple the score by 5. (Because the scale of attributes value is set between 1-5).

Using the formula mentioned in chapter 3 system design

Average weight value

$$\left(\frac{Sum\ of\ weight\ value\ of\ each\ educational\ website}{Total\ educational\ website\ visited\ \times 3(maximum\ weight\ value)}\right)$$

Percentage of education website among all website visited

$$\frac{Total\ educational\ website\ visited}{Total\ website\ visited}$$

$$Educational\ Attribute\ value = \left(\frac{\boldsymbol{Average\ weight\ value + educational\ website\ per\ overall\ visited\ website}}{2}\right) \times 5$$

The rules to determine the weight value for each educational website.

If the user time spent of the website is:

< 5 minutes (< 300s): 1.0 weight value

> 5 minutes && < 10 minutes (> 300s && <600s): 2.0 weight value

> 15 minute (> 900s): 3.0 weight value

Exploring Attributes

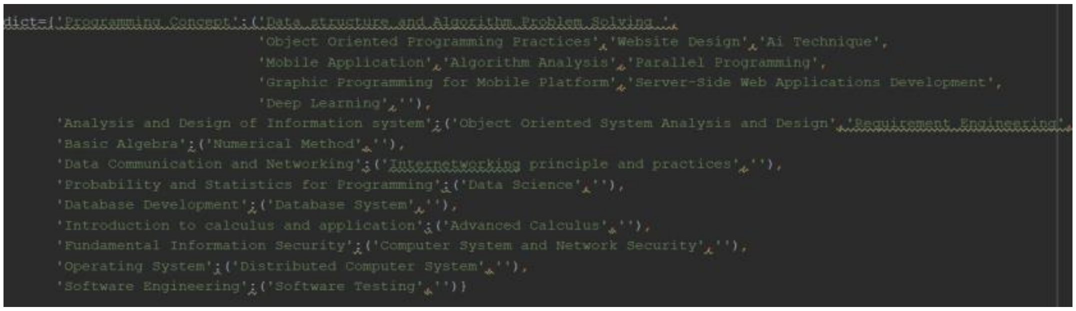
0.0 – 1.0: not so exploring
1.0 – 2.0: slightly exploring
2.0 – 3.0: moderately exploring
3.0 – 4.0: exploring
4.0 – 5.0: very exploring

Above rules show the characteristics of each range of attributes value and that’s how the exploring attribute is obtained

4.2 Learnability Attribute

As mentioned in chapter 3 system design, learnability attribute rating will be determined based on subject grade consistency. To achieve this, we will use Pycharm, a python language IDE to complete the work. Below is the chunk of code written in python language to determine the Learnability Attribute rating.

Dictionary of Subject relation

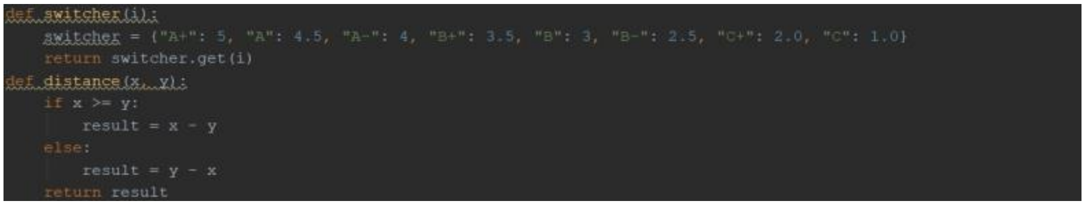


```
dict={'Programming Concept':('Data structure and Algorithm Problem Solving ',
                             'Object Oriented Programming Practices','Website Design','Ai Technique',
                             'Mobile Application','Algorithm Analysis','Parallel Programming',
                             'Graphic Programming for Mobile Platform','Server-Side Web Applications Development',
                             'Deep Learning',''),
      'Analysis and Design of Information system':('Object Oriented System Analysis and Design','Requirement Engineering',
      'Basic Algebra':('Numerical Method',''),
      'Data Communication and Networking':('Internetworking principle and practices',''),
      'Probability and Statistics for Programming':('Data Science',''),
      'Database Development':('Database System',''),
      'Introduction to calculus and application':('Advanced Calculus',''),
      'Fundamental Information Security':('Computer System and Network Security',''),
      'Operating System':('Distributed Computer System',''),
      'Software Engineering':('Software Testing','')}
```

Diagram 4.2.1 shows the python code to create dictionary for subject relation

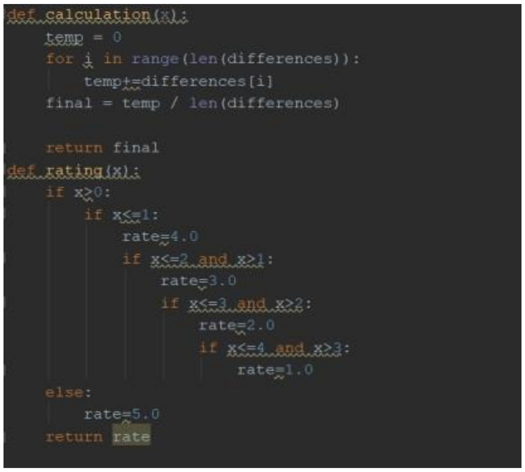
Dictionary created to define the relation rule between each subject according to the subject relation rule mentioned in chapter 3, which are all based on the program structure of UTAR Computer Science.

Function



```
def switcher(i):
    switcher = {"A+": 5, "A": 4.5, "A-": 4, "B+": 3.5, "B": 3, "B-": 2.5, "C+": 2.0, "C": 1.0}
    return switcher.get(i)
def distance(x, y):
    if x >= y:
        result = x - y
    else:
        result = y - x
    return result
```

Diagram 4.2.2 shows function defined



```
def calculation(x):
    temp = 0
    for i in range(len(differences)):
        temp+=differences[i]
    final = temp / len(differences)

    return final
def rating(x):
    if x>0:
        if x<=1:
            rate=4.0
            if x<=2 and x>1:
                rate=3.0
                if x<=3 and x>2:
                    rate=2.0
                    if x<=4 and x>3:
                        rate=1.0
    else:
        rate=5.0
    return rate
```

Diagram 4.2.3 shows function defined

There are 4 function created which are rating(), calculation(), switcher(), and distance().

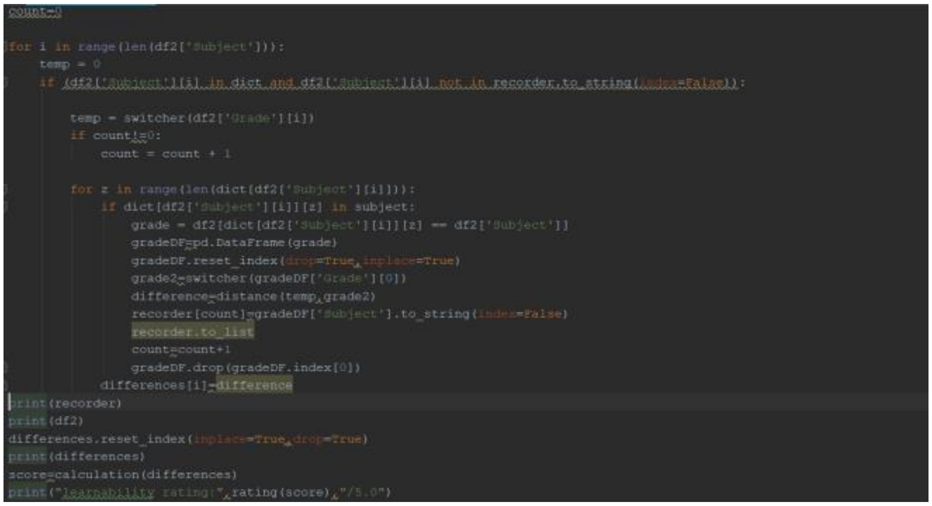
Swticher(): This function is defined to perform switch case functionality where it consists of case of different grade for A+ to C and their respective grade value.

Distance(): To calculate the differences between 2 passed reference.

Calculation(): To calculate the average value of all calculated subject grade differences.

Rating(): To return the final rating of Learnability rating.

Main function

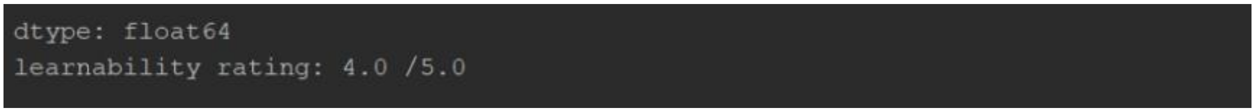


```
count=0

for i in range(len(df2['Subject'])):
    temp = 0
    if (df2['Subject'][i] in dict and df2['Subject'][i] not in recorder.to_string(index=False)):

        temp = switcher(df2['Grade'][i])
        if count!=0:
            count = count + 1

        for z in range(len(dict[df2['Subject'][i]])):
            if dict[df2['Subject'][i]][z] in subject:
                grade = df2[dict[df2['Subject'][i]][z] == df2['Subject']]
                gradeDF=pd.DataFrame(grade)
                gradeDF.reset_index(drop=True,inplace=True)
                grade2=switcher(gradeDF['Grade'][0])
                difference=distance(temp,grade2)
                recorder[count]=gradeDF['Subject'].to_string(index=False)
                recorder.to_list
                count=count+1
                gradeDF.drop(gradeDF.index[0])
            differences[i]=difference
print(recorder)
print(df2)
differences.reset_index(inplace=True,drop=True)
print(differences)
score=calculation(differences)
print("learnability rating:",rating(score),"/5.0")
```

Diagram 4.2.4 shows the main function

The main function first will compare the subject in datasets with the subject stated in subject relation rules then find its next succeeding subject and eventually call out distance() function to find the differences between them. With each subject distance are found, then will call calculation() method to find the average value, and pass the average value into rating() method to obtain the final learnability attribute rating value.

Sample Output



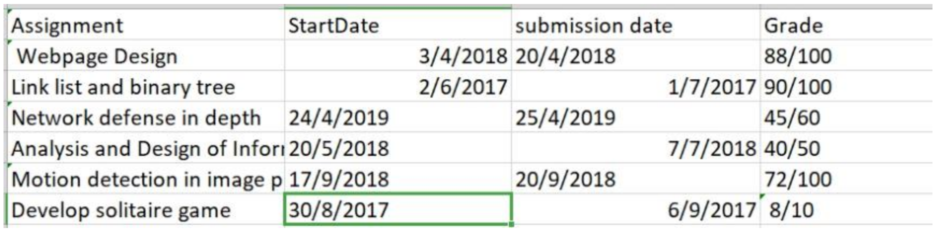
```
dtype: float64
learnability rating: 4.0 /5.0
```

Diagram 4.2.5 shows the sample output

4.3 Productivity Attribute

For productivity attribute, we will determine it by finding productivity of student in completing an assignment. The Tools used to do this will be Pycharm, a python IDE.

Datasets

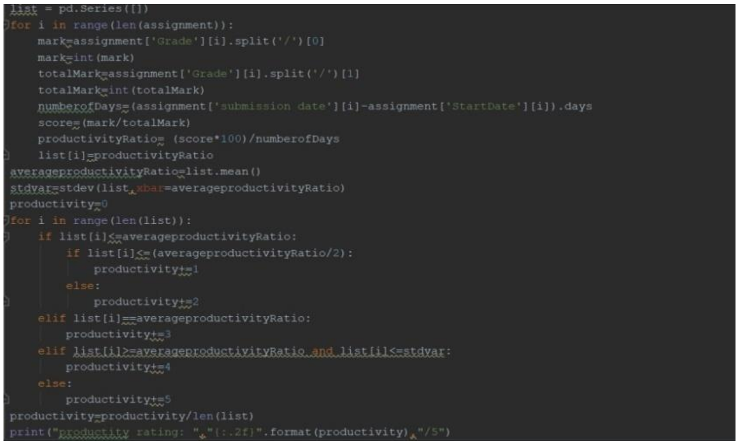


Assignment	StartDate	submission date	Grade
Webpage Design	3/4/2018	20/4/2018	88/100
Link list and binary tree	2/6/2017	1/7/2017	90/100
Network defense in depth	24/4/2019	25/4/2019	45/60
Analysis and Design of Infor	20/5/2018	7/7/2018	40/50
Motion detection in image p	17/9/2018	20/9/2018	72/100
Develop solitaire game	30/8/2017	6/9/2017	8/10

Diagram 4.3.1 shows the sample synthetic datasets used

Above diagram show the synthetic datasets that is created in this research for experimental purpose.

Main Function

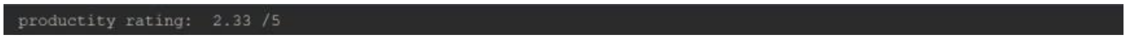


```
list = pd.Series([])
for i in range(len(assignment)):
    mark=assignment['Grade'][i].split('/')[0]
    mark=int(mark)
    totalMark=assignment['Grade'][i].split('/')[1]
    totalMark=int(totalMark)
    numberofDays=(assignment['submission date'][i]-assignment['StartDate'][i]).days
    score=(mark/totalMark)
    productivityRatio= (score*100)/numberofDays
    list[i]=productivityRatio
averageproductivityRatio=list.mean()
stdvar=stdev(list,xbar=averageproductivityRatio)
productivity=0
for i in range(len(list)):
    if list[i]<=averageproductivityRatio:
        if list[i]<=(averageproductivityRatio/2):
            productivity+=1
        else:
            productivity+=2
    elif list[i]==averageproductivityRatio:
        productivity+=3
    elif list[i]>=averageproductivityRatio and list[i]<=stdvar:
        productivity+=4
    else:
        productivity+=5
productivity=productivity/len(list)
print("productity rating: ","{:.2f}".format(productivity),"/5")
```

Diagram 4.3.2 shows the main function

The main function is programmed to perform some calculation mentioned in chapter 3 like finding standard deviation, mean and productivity ratio. After finding the productivity ratio of each assignment, it will then go the for loop to compared with mean and standard deviation of productivity ratio in order to rate the productive value ranged from 1 to 5. Each rating will eventually summed up and find the average value.

Sample output



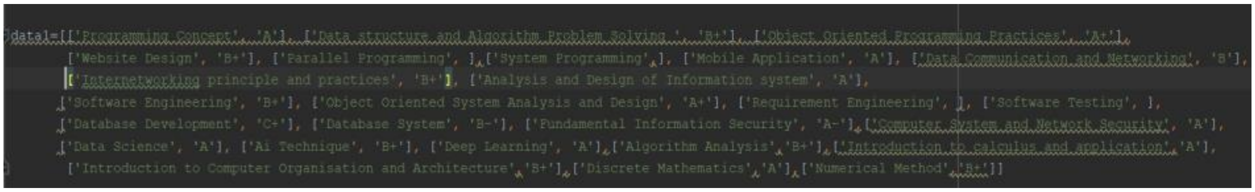
```
productity rating:  2.33 /5
```

Diagram 4.3.3 shows the sample output of the program

4.4 Domain Skillsets

Domain skillsets is all about determine averaged subject grade of each domain. As mentioned in chapter 3 there are 6 domains defined which are programming, networking, software, database, artificial intelligent and theory, logic and design. The tools used in for this attribute will be Pycharm, a python language IDE.

Datasets



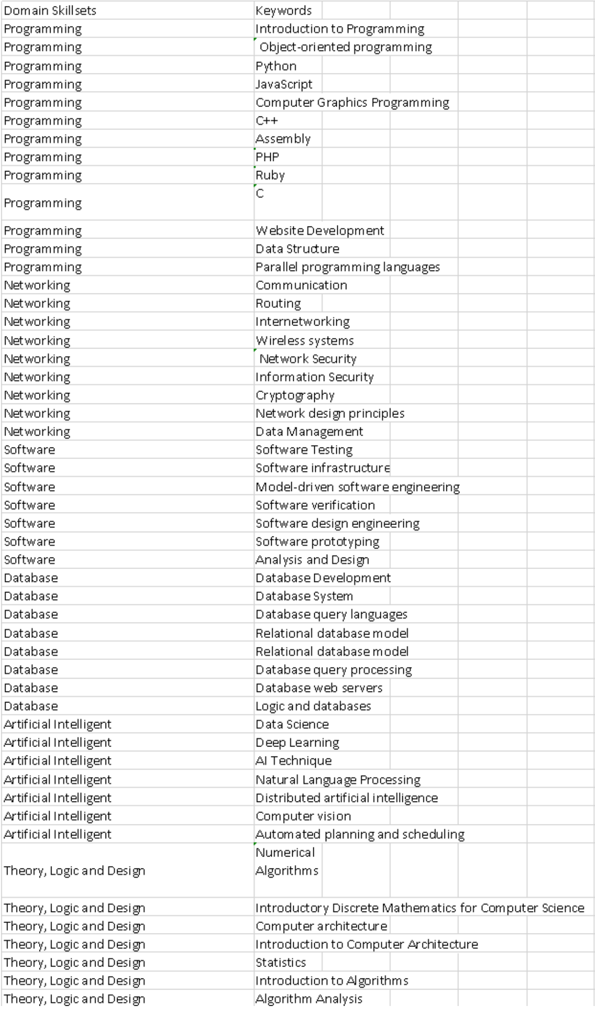
```
data1=[['Programming Concept', 'A'], ['Data structure and Algorithm Problem Solving ', 'B+'], ['Object Oriented Programming Practices', 'A+'],
       ['Website Design', 'B+'], ['Parallel Programming', ],['System Programming',], ['Mobile Application', 'A'], ['Data Communication and Networking', 'B'],
       ['Internetworking principle and practices', 'B+'], ['Analysis and Design of Information system', 'A'],
      ['Software Engineering', 'B+'], ['Object Oriented System Analysis and Design', 'A+'], ['Requirement Engineering', ], ['Software Testing', ],
      ['Database Development', 'C+'], ['Database System', 'B-'], ['Fundamental Information Security', 'A-'],['Computer System and Network Security', 'A'],
      ['Data Science', 'A'], ['Ai Technique', 'B+'], ['Deep Learning', 'A'],['Algorithm Analysis','B+'],['Introduction to calculus and application','A'],
       ['Introduction to Computer Organisation and Architecture','B+'],['Discrete Mathematics','A'],['Numerical Method','B+']]
```

Diagram 4.4.1 shows the datasets of subject and its grade

Above diagram shows the datasets of subject found in UTAR Computer Science program structure and its grade. However, all these subject are not yet being classified into the 6 different domain field. In order to classified them, machine learning text classifier technique is used to do it.

Before classify the datasets above, training is needed to be done first. Below are the labeled datasets that we used to train it to create classifier model. The labelled datasets are based on the ACM Computing classification (https://www.acm.org/publications/class-2012)

Labelled datasets for training

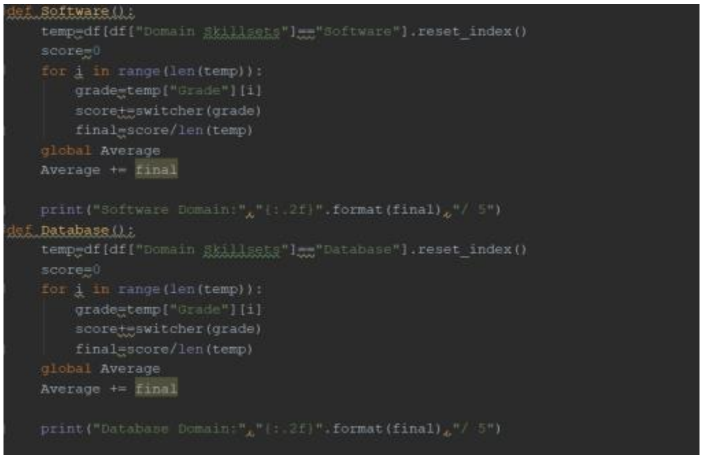


Domain Skillsets	Keywords
Programming	Introduction to Programming
Programming	Object-oriented programming
Programming	Python
Programming	JavaScript
Programming	Computer Graphics Programming
Programming	C++
Programming	Assembly
Programming	PHP
Programming	Ruby
Programming	C
Programming	Website Development
Programming	Data Structure
Programming	Parallel programming languages
Networking	Communication
Networking	Routing
Networking	Internetworking
Networking	Wireless systems
Networking	Network Security
Networking	Information Security
Networking	Cryptography
Networking	Network design principles
Networking	Data Management
Software	Software Testing
Software	Software infrastructure
Software	Model-driven software engineering
Software	Software verification
Software	Software design engineering
Software	Software prototyping
Software	Analysis and Design
Database	Database Development
Database	Database System
Database	Database query languages
Database	Relational database model
Database	Relational database model
Database	Database query processing
Database	Database web servers
Database	Logic and databases
Artificial Intelligent	Data Science
Artificial Intelligent	Deep Learning
Artificial Intelligent	AI Technique
Artificial Intelligent	Natural Language Processing
Artificial Intelligent	Distributed artificial intelligence
Artificial Intelligent	Computer vision
Artificial Intelligent	Automated planning and scheduling
Theory, Logic and Design	Numerical Algorithms
Theory, Logic and Design	Introductory Discrete Mathematics for Computer Science
Theory, Logic and Design	Computer architecture
Theory, Logic and Design	Introduction to Computer Architecture
Theory, Logic and Design	Statistics
Theory, Logic and Design	Introduction to Algorithms
Theory, Logic and Design	Algorithm Analysis

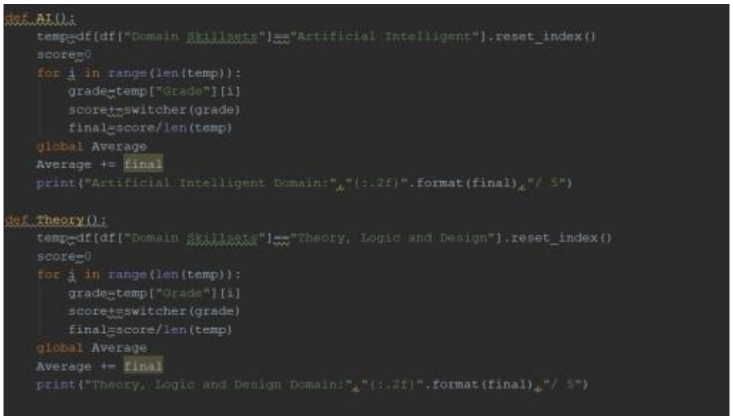
Diagram 4.4.2 shows the labelled datasets that will be used for text classifier training

With the labelled datasets is prepared, now we can train the classifier model by using a machine learning algorithm. In this research, LinearSVC will be used as the machine learning classifier to classified the subject.

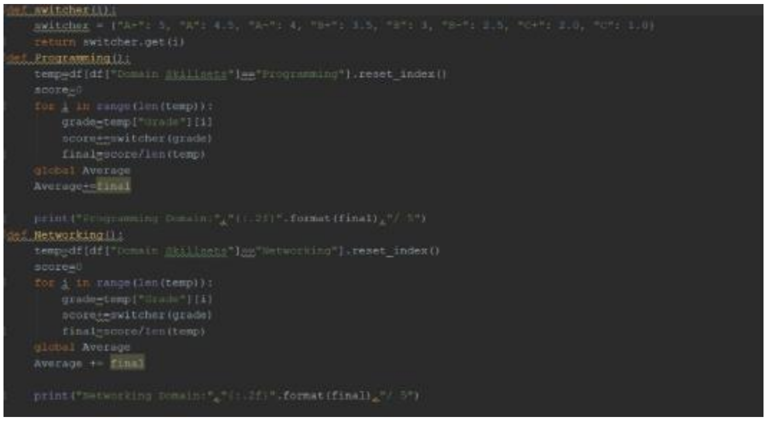
Defined Function



```
def Software():
    temp=df[df["Domain Skillsets"]=="Software"].reset_index()
    score=0
    for i in range(len(temp)):
        grade=temp["Grade"][i]
        score+=switcher(grade)
        final=score/len(temp)
    global Average
    Average += final

    print("Software Domain:","{:.2f}".format(final),"/ 5")
def Database():
    temp=df[df["Domain Skillsets"]=="Database"].reset_index()
    score=0
    for i in range(len(temp)):
        grade=temp["Grade"][i]
        score+=switcher(grade)
        final=score/len(temp)
    global Average
    Average += final

    print("Database Domain:","{:.2f}".format(final),"/ 5")
```



```
def AI():
    temp=df[df["Domain Skillsets"]=="Artificial Intelligent"].reset_index()
    score=0
    for i in range(len(temp)):
        grade=temp["Grade"][i]
        score+=switcher(grade)
        final=score/len(temp)
    global Average
    Average += final
    print("Artificial Intelligent Domain:","{:.2f}".format(final),"/ 5")

def Theory():
    temp=df[df["Domain Skillsets"]=="Theory, Logic and Design"].reset_index()
    score=0
    for i in range(len(temp)):
        grade=temp["Grade"][i]
        score+=switcher(grade)
        final=score/len(temp)
    global Average
    Average += final
    print("Theory, Logic and Design Domain:","{:.2f}".format(final),"/ 5")
```



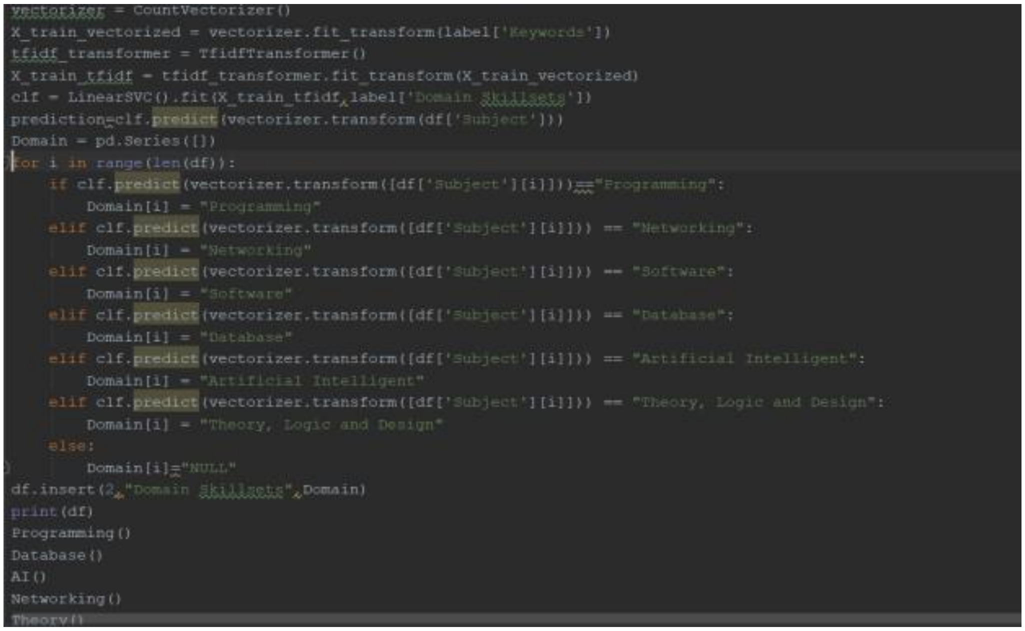
```
def switcher(i):
    switcher = {"A+": 5, "A": 4.5, "A-": 4, "B+": 3.5, "B": 3, "B-": 2.5, "C+": 2.0, "C": 1.0}
    return switcher.get(i)
def Programming():
    temp=df[df["Domain Skillsets"]=="Programming"].reset_index()
    score=0
    for i in range(len(temp)):
        grade=temp["Grade"][i]
        score+=switcher(grade)
        final=score/len(temp)
    global Average
    Average+=final

    print("Programming Domain:","{:.2f}".format(final),"/ 5")
def Networking():
    temp=df[df["Domain Skillsets"]=="Networking"].reset_index()
    score=0
    for i in range(len(temp)):
        grade=temp["Grade"][i]
        score+=switcher(grade)
        final=score/len(temp)
    global Average
    Average += final

    print("Networking Domain:","{:.2f}".format(final),"/ 5")
```

Diagram 4.4.3 shows the function defined in the program

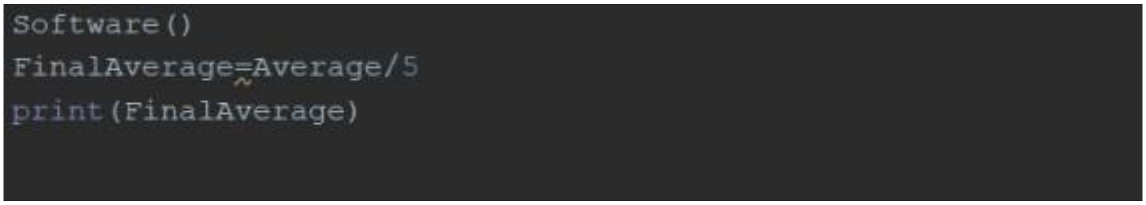
Swticher(): This function is defined to perform switch case functionality where it consists of case of different grade for A+ to C and their respective grade value.

Programming(), Network(),Theory(), AI(), Database(), Software(): All these function are defined to calculate respective average grade value of subject which belongs to respective domain.

Main Function



```
vectorizer = CountVectorizer()
X_train_vectorized = vectorizer.fit_transform(label['Keywords'])
tfidf_transformer = TfidfTransformer()
X_train_tfidf = tfidf_transformer.fit_transform(X_train_vectorized)
clf = LinearSVC().fit(X_train_tfidf,label['Domain Skillsets'])
prediction=clf.predict(vectorizer.transform(df['Subject']))
Domain = pd.Series([])
for i in range(len(df)):
    if clf.predict(vectorizer.transform([df['Subject'][i]]))=="Programming":
        Domain[i] = "Programming"
    elif clf.predict(vectorizer.transform([df['Subject'][i]])) == "Networking":
        Domain[i] = "Networking"
    elif clf.predict(vectorizer.transform([df['Subject'][i]])) == "Software":
        Domain[i] = "Software"
    elif clf.predict(vectorizer.transform([df['Subject'][i]])) == "Database":
        Domain[i] = "Database"
    elif clf.predict(vectorizer.transform([df['Subject'][i]])) == "Artificial Intelligent":
        Domain[i] = "Artificial Intelligent"
    elif clf.predict(vectorizer.transform([df['Subject'][i]])) == "Theory, Logic and Design":
        Domain[i] = "Theory, Logic and Design"
    else:
        Domain[i]="NULL"
df.insert(2,"Domain Skillsets",Domain)
print(df)
Programming()
Database()
AI()
Networking()
Theory()
```

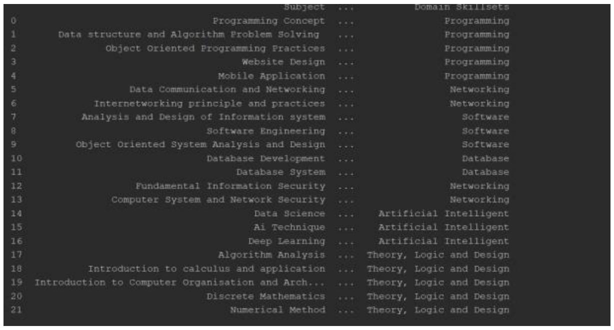


```
Software()
FinalAverage=Average/5
print(FinalAverage)
```

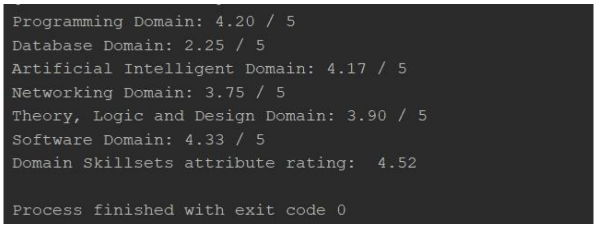
Diagram 4.4.4 shows the main function of the program

In the main function, the labelled datasets is preprocessed by vectorizing it and will be used to fit into the LinearSVC model for training. After the model is being created, the Subject and grade datasets will be used to fit into the model and start to classify all the subject in the dataset.

Sample output



```
                                              Subject  ...          Domain Skillsets
0                                 Programming Concept  ...               Programming
1       Data structure and Algorithm Problem Solving   ...               Programming
2              Object Oriented Programming Practices  ...               Programming
3                                      Website Design  ...               Programming
4                                  Mobile Application  ...               Programming
5                 Data Communication and Networking  ...                Networking
6             Internetworking principle and practices  ...                Networking
7           Analysis and Design of Information system  ...                  Software
8                                Software Engineering  ...                  Software
9          Object Oriented System Analysis and Design  ...                  Software
10                               Database Development  ...                  Database
11                                    Database System  ...                  Database
12                  Fundamental Information Security  ...                Networking
13                Computer System and Network Security  ...                Networking
14                                       Data Science  ...   Artificial Intelligent
15                                       Ai Technique  ...   Artificial Intelligent
16                                      Deep Learning  ...   Artificial Intelligent
17                                 Algorithm Analysis  ...  Theory, Logic and Design
18           Introduction to calculus and application  ...  Theory, Logic and Design
19  Introduction to Computer Organisation and Arch...  ...  Theory, Logic and Design
20                               Discrete Mathematics  ...  Theory, Logic and Design
21                                   Numerical Method  ...  Theory, Logic and Design
```



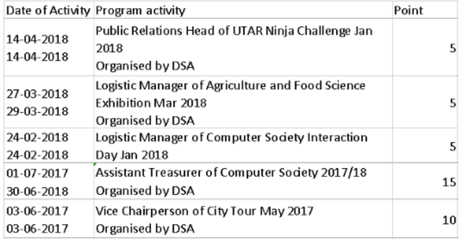
```
Programming Domain: 4.20 / 5
Database Domain: 2.25 / 5
Artificial Intelligent Domain: 4.17 / 5
Networking Domain: 3.75 / 5
Theory, Logic and Design Domain: 3.90 / 5
Software Domain: 4.33 / 5
Domain Skillsets attribute rating:  4.52

Process finished with exit code 0
```

Diagram 4.4.5 shows the Sample output of classification and Domain Skillsets attribute rating

4.5 Leadership Attribute

Leadership Attribute will be determined through summing up all the leadership point of student USSDC leadership component and based on the total to determine the leadership rating. The tools used for this will be pycharm, a python language IDE.

Dataset

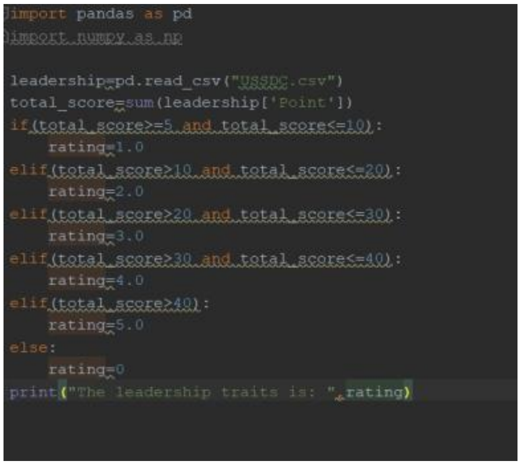


Date of Activity	Program activity	Point
14-04-2018 14-04-2018	Public Relations Head of UTAR Ninja Challenge Jan 2018 Organised by DSA	5
27-03-2018 29-03-2018	Logistic Manager of Agriculture and Food Science Exhibition Mar 2018 Organised by DSA	5
24-02-2018 24-02-2018	Logistic Manager of Computer Society Interaction Day Jan 2018	5
01-07-2017 30-06-2018	Assistant Treasurer of Computer Society 2017/18 Organised by DSA	15
03-06-2017 03-06-2017	Vice Chairperson of City Tour May 2017 Organised by DSA	10

Diagram 4.5.1 shows the dataset for this section

Above dataset records USSDC point obtain from leadership component of a student.

Main function



```
import pandas as pd
import numpy as np

leadership=pd.read_csv("USSDC.csv")
total_score=sum(leadership['Point'])
if(total_score>=5 and total_score<=10):
    rating=1.0
elif(total_score>10 and total_score<=20):
    rating=2.0
elif(total_score>20 and total_score<=30):
    rating=3.0
elif(total_score>30 and total_score<=40):
    rating=4.0
elif(total_score>40):
    rating=5.0
else:
    rating=0
print("The leadership traits is: ",rating)
```

Diagram 4.5.2 shows the main function of the program

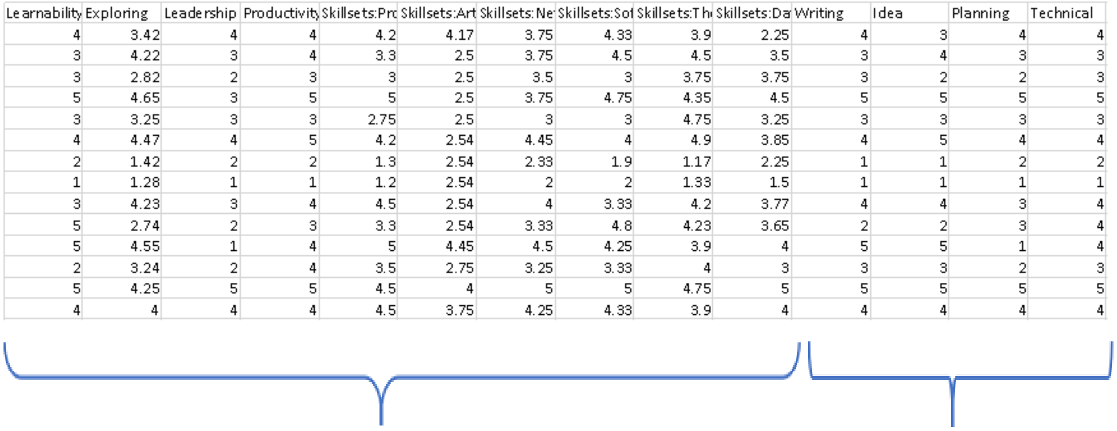
The main function programmed to read the datasets and total up the USSDC leadership point and compare the total with the rules to determine the final leadership rating.

4.6 Machine learning

Machine learning is used for the purpose of pattern recognition that allow the program to learn itself without explicitly program to perform specific task.

In this research, machine learning will be used to perform team matching recommendation by predicting student's performance in different role of a team based on different personality. In order to do so, firstly we need a dataset to perform training. The dataset will be the profile created using the pervasive profiling method mentioned earlier which consist of 5 different characteristics attribute. All these different profile of different student will be labeled with the 4 role performance rating as mentioned earlier in chapter 3. Labelling is done in order to perform supervised machine learning technique. In this paper, the dataset used is a synthetic datasets which all the dataset is artificially prepared for the purpose of experimentally train the machine learning model.

Labelled datasets



Learnability	Exploring	Leadership	Productivity	Skillsets:Pr	Skillsets:Art	Skillsets:Ne	Skillsets:Sof	Skillsets:Th	Skillsets:Da	Writing	Idea	Planning	Technical
4	3.42	4	4	4.2	4.17	3.75	4.33	3.9	2.25	4	3	4	4
3	4.22	3	4	3.3	2.5	3.75	4.5	4.5	3.5	3	4	3	3
3	2.82	2	3	3	2.5	3.5	3	3.75	3.75	3	2	2	3
5	4.65	3	5	5	2.5	3.75	4.75	4.35	4.5	5	5	5	5
3	3.25	3	3	2.75	2.5	3	3	4.75	3.25	3	3	3	3
4	4.47	4	5	4.2	2.54	4.45	4	4.9	3.85	4	5	4	4
2	1.42	2	2	1.3	2.54	2.33	1.9	1.17	2.25	1	1	2	2
1	1.28	1	1	1.2	2.54	2	2	1.33	1.5	1	1	1	1
3	4.23	3	4	4.5	2.54	4	3.33	4.2	3.77	4	4	3	4
5	2.74	2	3	3.3	2.54	3.33	4.8	4.23	3.65	2	2	3	4
5	4.55	1	4	5	4.45	4.5	4.25	3.9	4	5	5	1	4
2	3.24	2	4	3.5	2.75	3.25	3.33	4	3	3	3	2	3
5	4.25	5	5	4.5	4	5	5	4.75	5	5	5	5	5
4	4	4	4	4.5	3.75	4.25	4.33	3.9	4	4	4	4	4

Profile Roles

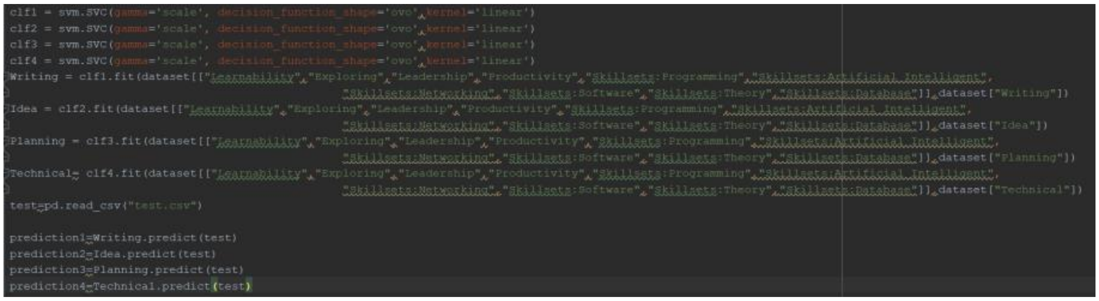
Diagram 4.6.1 shows the datasets used to train the machine learning model

Above diagram shows the synthetic dataset being created to train the machine learning model. Each row represents individual's characteristics profile labelled with 4 roles.

Machine Learning Algorithms

The machine learning algorithms that had chosen to be used to train the datasets and perform prediction is SVC from the SVM(Support vector machine). The reason that SVM is chosen is because it can perform multi-class classification and according to Wei-Xiang Liu and Ching-Hsue Cheng (2016) research mentioned in chapter 2 literature, SVM is a much more effective and accurate algorithms to solve classification problem.

Main function

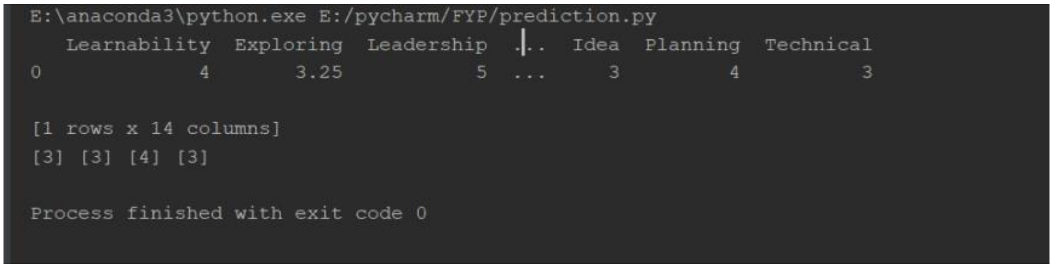


```
clf1 = svm.SVC(gamma='scale', decision_function_shape='ovo',kernel='linear')
clf2 = svm.SVC(gamma='scale', decision_function_shape='ovo',kernel='linear')
clf3 = svm.SVC(gamma='scale', decision_function_shape='ovo',kernel='linear')
clf4 = svm.SVC(gamma='scale', decision_function_shape='ovo',kernel='linear')
Writing = clf1.fit(dataset[["Learnability","Exploring","Leadership","Productivity","Skillsets:Programming","Skillsets:Artificial Intelligent",
                                              "Skillsets:Networking","Skillsets:Software","Skillsets:Theory","Skillsets:Database"]],dataset["Writing"])
Idea = clf2.fit(dataset[["Learnability","Exploring","Leadership","Productivity","Skillsets:Programming","Skillsets:Artificial Intelligent",
                                              "Skillsets:Networking","Skillsets:Software","Skillsets:Theory","Skillsets:Database"]],dataset["Idea"])
Planning = clf3.fit(dataset[["Learnability","Exploring","Leadership","Productivity","Skillsets:Programming","Skillsets:Artificial Intelligent",
                                              "Skillsets:Networking","Skillsets:Software","Skillsets:Theory","Skillsets:Database"]],dataset["Planning"])
Technical= clf4.fit(dataset[["Learnability","Exploring","Leadership","Productivity","Skillsets:Programming","Skillsets:Artificial Intelligent",
                                              "Skillsets:Networking","Skillsets:Software","Skillsets:Theory","Skillsets:Database"]],dataset["Technical"])
test=pd.read_csv("test.csv")

prediction1=Writing.predict(test)
prediction2=Idea.predict(test)
prediction3=Planning.predict(test)
prediction4=Technical.predict(test)
```

Diagram 4.6.2 shows the main function of the program

Above diagram shows the main function of the program of machine learning. As you can see 4 SVM model are created to predict all 4 different roles which are writing, idea, planning and technical. The input of training is the profile of students.

Sample output



```
E:\anaconda3\python.exe E:/pycharm/FYP/prediction.py
   Learnability  Exploring  Leadership  ...  Idea  Planning  Technical
0             4       3.25           5  ...     3         4          3

[1 rows x 14 columns]
[3] [3] [4] [3]

Process finished with exit code 0
```

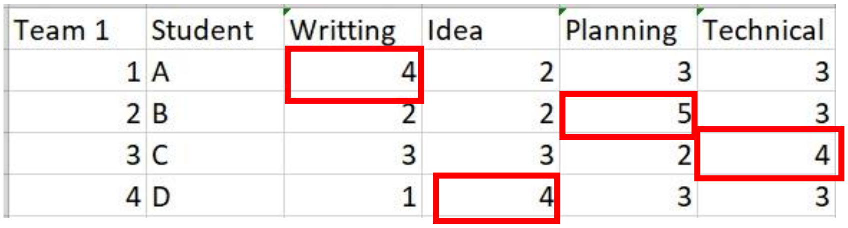
Diagram 4.6.3 above shows the output of the program

Above diagram shows the output of the prediction which is the performance rating of each roles.

4.7 Team matching recommendation

In order to do team matching recommendation, here in this paper we archive this by make use of the predicted performance rating of each task's role. The method of recommending a much effective team matching is by having a much more balanced team formation. A balanced team means that a team should have each member at least good at one of the defined roles and 4 of the roles should be fulfilled.

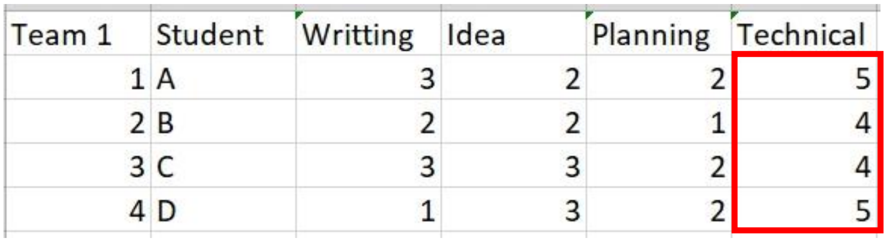
Below is the example of a balanced and effective team matching recommendation.



Team 1	Student	Writting	Idea	Planning	Technical
1	A	4	2	3	3
2	B	2	2	5	3
3	C	3	3	2	4
4	D	1	4	3	3

Diagram 4.7.1 shows a example of recommended effective assignment team

As you can see from the diagram above, the recommended assignment team consist of 4 student which each of them had higher rating of at least one of a roles which is a much balanced and effective team.



Team 1	Student	Writting	Idea	Planning	Technical
1	A	3	2	2	5
2	B	2	2	1	4
3	C	3	3	2	4
4	D	1	3	2	5

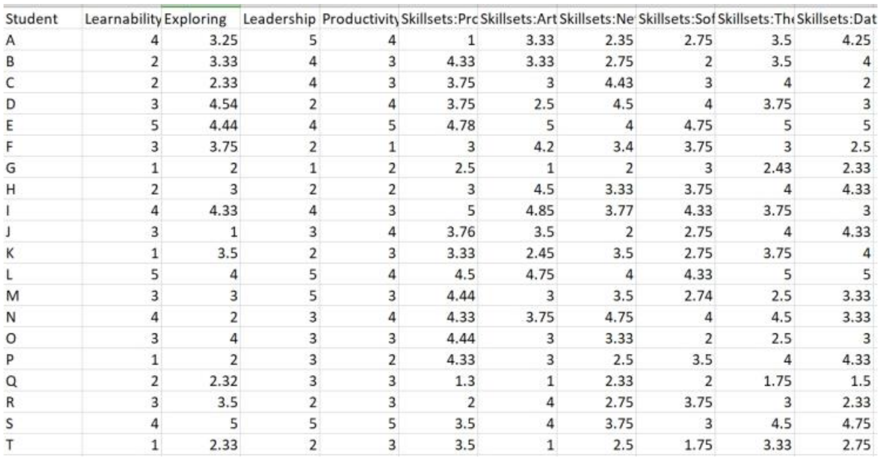
Diagram 4.7.2 shows an example of not balanced assignment team

Above diagram shows another example of an assignment group which consists of unbalanced group member with all of them good at technical and all of them are not really good at planning which mean that this group doesn’t have a real leader to help lead the group.

Thus, this is the method defined in this research to recommend a much more effective team formation by balancing out and utilizing each defined assignment roles.

Chapter 5 Implementation and Testing

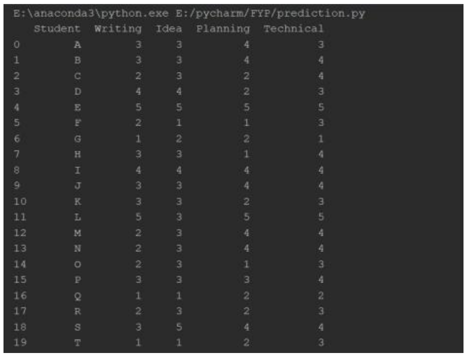
In order to test this out we can use this method and implement into a simulated class of 20 students who are trying to form group for assignment. First of all, all the desired data will be collected from them to obtain their respective profile. Below is the diagram of synthetic profile of all the student in the class.



Student	Learnability	Exploring	Leadership	Productivity	Skillsets:Prc	Skillsets:Art	Skillsets:Ne	Skillsets:Sof	Skillsets:Th	Skillsets:Dat
A	4	3.25	5	4	1	3.33	2.35	2.75	3.5	4.25
B	2	3.33	4	3	4.33	3.33	2.75	2	3.5	4
C	2	2.33	4	3	3.75	3	4.43	3	4	2
D	3	4.54	2	4	3.75	2.5	4.5	4	3.75	3
E	5	4.44	4	5	4.78	5	4	4.75	5	5
F	3	3.75	2	1	3	4.2	3.4	3.75	3	2.5
G	1	2	1	2	2.5	1	2	3	2.43	2.33
H	2	3	2	2	3	4.5	3.33	3.75	4	4.33
I	4	4.33	4	3	5	4.85	3.77	4.33	3.75	3
J	3	1	3	4	3.76	3.5	2	2.75	4	4.33
K	1	3.5	2	3	3.33	2.45	3.5	2.75	3.75	4
L	5	4	5	4	4.5	4.75	4	4.33	5	5
M	3	3	5	3	4.44	3	3.5	2.74	2.5	3.33
N	4	2	3	4	4.33	3.75	4.75	4	4.5	3.33
O	3	4	3	3	4.44	3	3.33	2	2.5	3
P	1	2	3	2	4.33	3	2.5	3.5	4	4.33
Q	2	2.32	3	3	1.3	1	2.33	2	1.75	1.5
R	3	3.5	2	3	2	4	2.75	3.75	3	2.33
S	4	5	5	5	3.5	4	3.75	3	4.5	4.75
T	1	2.33	2	3	3.5	1	2.5	1.75	3.33	2.75

Diagram 5.1 Shows the profile of 20 students in a class

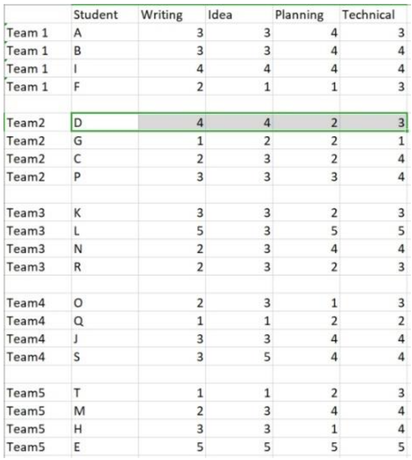
All this profile will be fit into the SVM model for prediction to determine their respective assignment role's rating.



```
E:\anaconda3\python.exe E:/pycharm/FYP/prediction.py
   Student  Writing  Idea  Planning  Technical
0        A        3     3         4          3
1        B        3     3         4          4
2        C        2     3         2          4
3        D        4     4         2          3
4        E        5     5         5          5
5        F        2     1         1          3
6        G        1     2         2          1
7        H        3     3         1          4
8        I        4     4         4          4
9        J        3     3         4          4
10       K        3     3         2          3
11       L        5     3         5          5
12       M        2     3         4          4
13       N        2     3         4          4
14       O        2     3         1          3
15       P        3     3         3          4
16       Q        1     1         2          2
17       R        2     3         2          3
18       S        3     5         4          4
19       T        1     1         2          3
```

Diagram 5.2 shows the predicted assignment role's rating of 20 students in class

Then team matching recommendation can be done by utilizing each student's role rating to form a much more balanced group.



	Student	Writing	Idea	Planning	Technical
Team 1	A	3	3	4	3
Team 1	B	3	3	4	4
Team 1	I	4	4	4	4
Team 1	F	2	1	1	3
Team2	D	4	4	2	3
Team2	G	1	2	2	1
Team2	C	2	3	2	4
Team2	P	3	3	3	4
Team3	K	3	3	2	3
Team3	L	5	3	5	5
Team3	N	2	3	4	4
Team3	R	2	3	2	3
Team4	O	2	3	1	3
Team4	Q	1	1	2	2
Team4	J	3	3	4	4
Team4	S	3	5	4	4
Team5	T	1	1	2	3
Team5	M	2	3	4	4
Team5	H	3	3	1	4
Team5	E	5	5	5	5

Diagram 5.3 shows the Team matched recommendation of 5 assignment group

As you can see above assignment team recommendation, each group are balanced based on their assignment role’s rating

Chapter 6: Conclusion

Profiling is a process where data is collected, analysis is done to a person psychological and behavioral in order to predict the person capabilities or to identifying categories of people. This research had proposed several methods to profile a student with 5 characteristics which are Leadership, Productivity, Learnability, Domain Skillsets, and Exploring. Each attribute has their own different method to collect the data required in order to determine their respective rating.

With profiling method being defined, we use the student profile to predict student assignment role's capabilities. There are 4 assignment roles define in this research as well which are Planning, Idea, Writing and Technical. With the use of machine learning technique, it will eventually learn the pattern by fitting dataset consists of different student profile labeled with respective assignment roles rating into the machine learning algorithm to create a model for prediction. In this paper, SVM classifier is chosen as the machine learning algorithm and supervise learning is the technique used. The SVM model will then used to predict student's assignment roles capabilities. However, the prediction in this research might not be accurate, as the dataset used to train the model is synthetic dataset, as it is used for experimental purposed.

Team matching recommendation is done based on the predicted assignment role rating. The recommendation is done by forming a much balance group with each role at least have one group member good at it and might able to help cover it.

The problem encountered in this research is the profiling part where different methods have to be designed in order to determine student attribute value which best describe their characteristics. To determine student attribute value, data has to be collected in order to do so, however the biggest problem is what data is required to do this, different attributes are describing differently in term of characteristics, thus each of them required different data as well. Even if the data needed to determine the attribute rating is successfully define, how to collected those data? How to make use of the data to determine the attribute value is always a question. The method proposed in this paper are all experimental, all of the method that had been performed are just an idea, it might not be accurate.

The idea proposed in this research can be the possible solution to help solving problem of assignment group formation in university. With the help of this research method, a much more effective and balanced team can be recommended to be formed.

There are some further improvement that can be made as well for this research. Currently, the method proposed to perform team matching recommendation is artificially match by referencing the student predicted assignment roles capabilities. This whole process can all be automated without involving human, which is using deep learning technique to do it. With the help of deep learning, the whole team matching process can all be automated and predicted by it. RNN (Recurrent Neural Networks) might be the choice of algorithm to use in team matching because RNN can capture relationship across input meaningfully.

Bibliography

1) Julieta Noguez, Luis Neri, et.al, 2016. Characteristics of self-regulation of engineering students to predict and improve their academic performance. 2016 IEEE Frontiers in Education Conference (FIE), pp. 1-4.
2) Michael Forte, Christopher Hummel, et.al, 2010. Learning Human Behavioral Profiles in a Cyber Environment. 2010 IEEE Systems and Information Engineering Design Symposium, pp. 182-185.
3) Judith J. Azcarraga*, John Francis Ibanez Jr., et.al, 2011. Use of Personality Profile in Predicting Academic Emotion based on Brainwaves Signals and Mouse Behavior, 2011 Third International Conference on Knowledge and Systems Engineering, pp. 239-243.
4) Riah F. Elcullada-Encarnacion, 2018. Academic Advising System Using Data Mining Method for Decision Making Support. 2018 4th International Conference on Computer and Technology Applications, pp. 29-32.
5) Atta-ur-Rahman, 2013. Teacher Assessment and Profiling using Fuzzy Rule based System and Apriori Algorithm. Available from: https://pdfs.semanticscholar.org/7b41/2ea18edf1519baff36b427d26944024d8df4.pdf
6) Wei-Xiang Liu and Ching-Hsue Cheng, 2016. A hybrid method based on MLFS approach to analyze students' academic achievement. 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), pp.1625- pp.1630.
7) Karen Yang and Nao Glaser, 2017. Prediction of personality first impression with deep Bimodal LSTM. Available from: https://www.semanticscholar.org/paper/Prediction-of-Personality-First-Impressions-With-Stanford-Mall/8871e6f4e3876e5fcf3a5a445d7636abdcb91574. [2017]

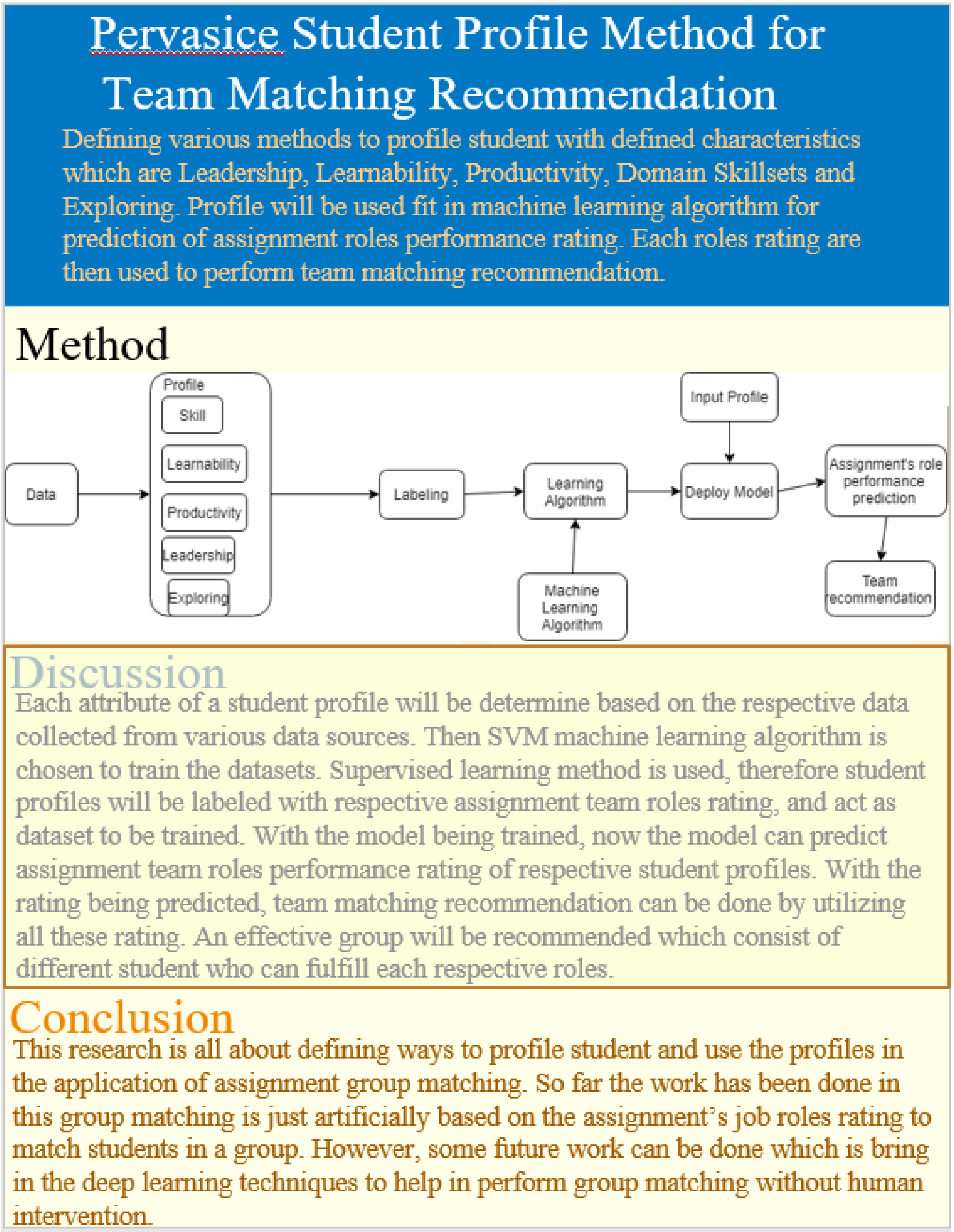
8) Matthew Mayo, 2018.Frameworks for Approaching the Machine Learning Process. Available from: https://www.kdnuggets.com/2018/05/general-approaches-machine-learning-process.html

9) J.M.Harley, G.J.Trevors, et.al, 2013. Clustering and Profiling Students According to their Interactions with an Intelligent Tutoring System Fostering Self-Regulated Learning. Available from: https://pdfs.semanticscholar.org/0f43/45d8d809f3952c4b6104f7cba462b398b7d3.pdf

10) S.Mojarad, A.Essa, et.al, 2018. Data-driven learner profiling based on clustering student behaviors: Learning consistency, pace and effort. Available from: http://www.upenn.edu/learninganalytics/ryanbaker/ITS_2018_Shirin_final.pdf

11) Jason Brownlee, 2016. What is deep learning. Available from: https://machinelearningmastery.com/what-is-deep-learning/.

Poster



Pervasice Student Profile Method for Team Matching Recommendation

Defining various methods to profile student with defined characteristics which are Leadership, Learnability, Productivity, Domain Skillsets and Exploring. Profile will be used fit in machine learning algorithm for prediction of assignment roles performance rating. Each roles rating are then used to perform team matching recommendation.

Method

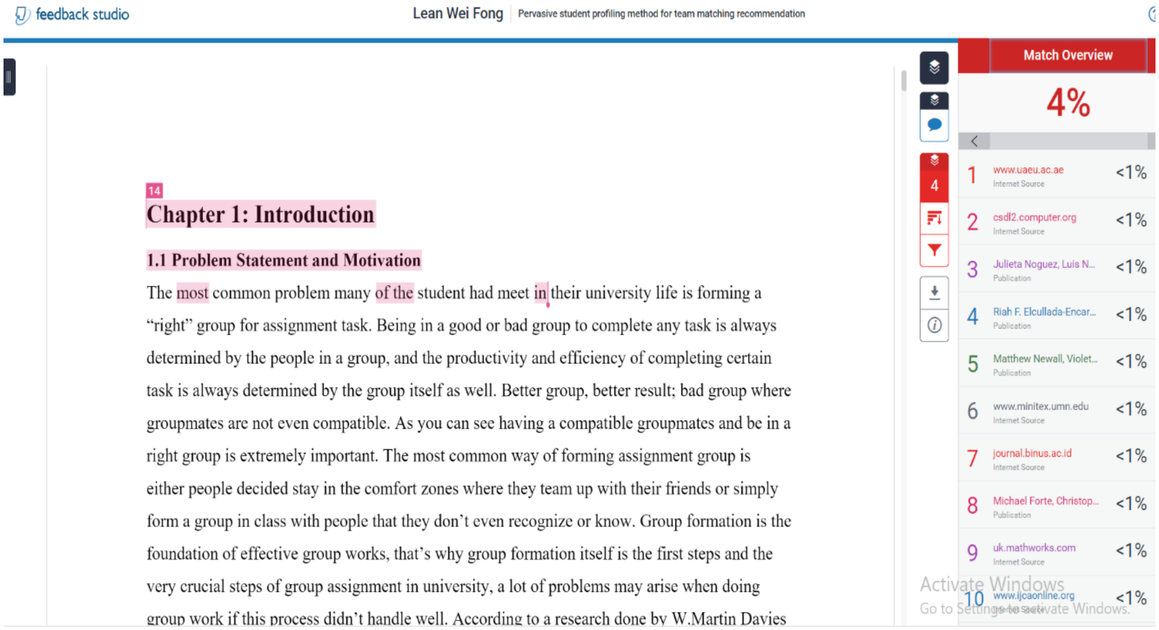
Discussion

Each attribute of a student profile will be determine based on the respective data collected from various data sources. Then SVM machine learning algorithm is chosen to train the datasets. Supervised learning method is used, therefore student profiles will be labeled with respective assignment team roles rating, and act as dataset to be trained. With the model being trained, now the model can predict assignment team roles performance rating of respective student profiles. With the rating being predicted, team matching recommendation can be done by utilizing all these rating. An effective group will be recommended which consist of different student who can fulfill each respective roles.

Conclusion

This research is all about defining ways to profile student and use the profiles in the application of assignment group matching. So far the work has been done in this group matching is just artificially based on the assignment's job roles rating to match students in a group. However, some future work can be done which is bring in the deep learning techniques to help in perform group matching without human intervention.

Plagarism Check Result



feedback studio

Lean Wei Fong | Pervasive student profiling method for team matching recommendation

14

Chapter 1: Introduction

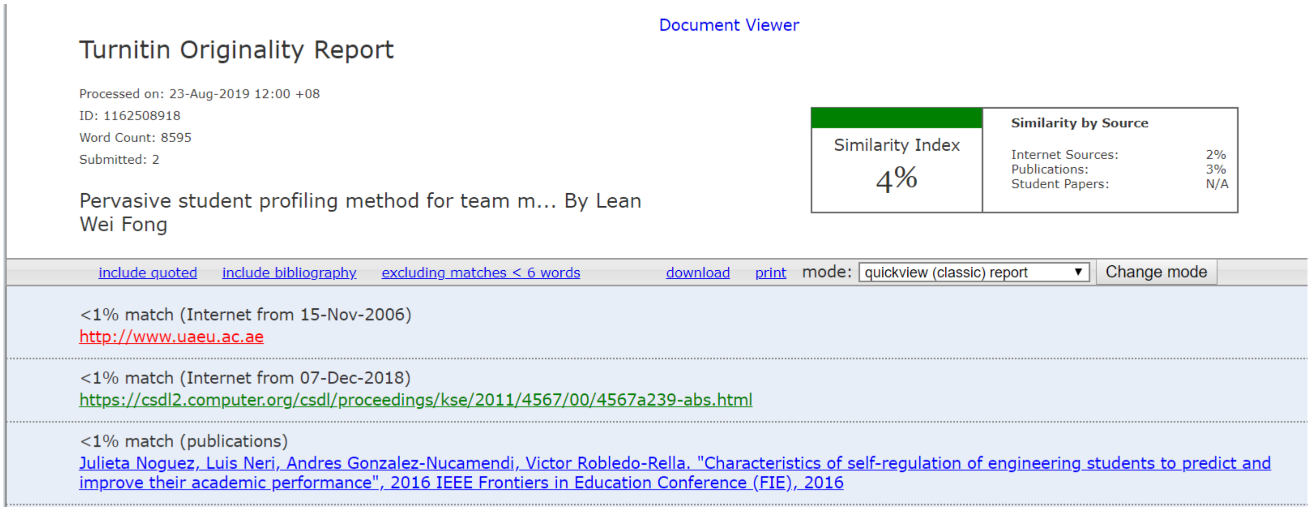
1.1 Problem Statement and Motivation

The most common problem many of the student had meet in their university life is forming a "right" group for assignment task. Being in a good or bad group to complete any task is always determined by the people in a group, and the productivity and efficiency of completing certain task is always determined by the group itself as well. Better group, better result; bad group where groupmates are not even compatible. As you can see having a compatible groupmates and be in a right group is extremely important. The most common way of forming assignment group is either people decided stay in the comfort zones where they team up with their friends or simply form a group in class with people that they don't even recognize or know. Group formation is the foundation of effective group works, that's why group formation itself is the first steps and the very crucial steps of group assignment in university, a lot of problems may arise when doing group work if this process didn't handle well. According to a research done by W.Martin Davies

Match Overview

4%

1	www.uaeu.ac.ae Internet Source	<1%
2	csdl2.computer.org Internet Source	<1%
3	Julieta Noguez, Luis N... Publication	<1%
4	Riah F. Elcullada-Encar... Publication	<1%
5	Matthew Newall, Violet... Publication	<1%
6	www.minitex.umn.edu Internet Source	<1%
7	journal.binus.ac.id Internet Source	<1%
8	Michael Forte, Christop... Publication	<1%
9	uk.mathworks.com Internet Source	<1%
10	www.ijcaonline.org Internet Source	<1%



Document Viewer

Turnitin Originality Report

Processed on: 23-Aug-2019 12:00 +08
ID: 1162508918
Word Count: 8595
Submitted: 2

Pervasive student profiling method for team m... By Lean Wei Fong

Similarity Index
4%

Similarity by Source

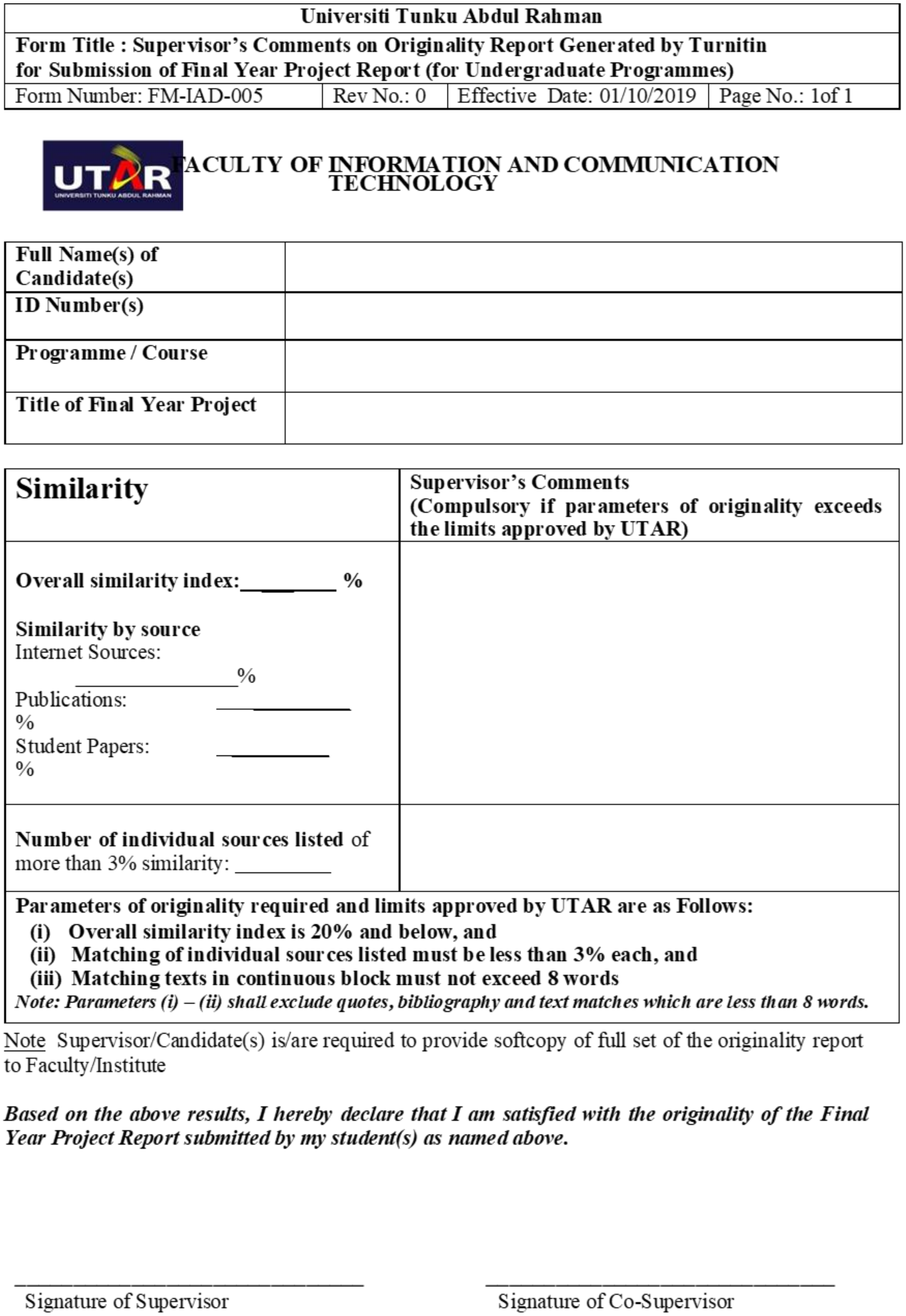
Internet Sources:	2%
Publications:	3%
Student Papers:	N/A

include quoted
include bibliography
excluding matches < 6 words
download
print
mode: quickview (classic) report ▾
Change mode

<1% match (Internet from 15-Nov-2006)
http://www.uaeu.ac.ae

<1% match (Internet from 07-Dec-2018)
https://csdl2.computer.org/csdl/proceedings/kse/2011/4567/00/4567a239-abs.html

<1% match (publications)
Julieta Noguez, Luis Neri, Andres Gonzalez-Nucamendi, Victor Robledo-Rella. "Characteristics of self-regulation of engineering students to predict and improve their academic performance", 2016 IEEE Frontiers in Education Conference (FIE), 2016



Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2019	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	
ID Number(s)	
Programme / Course	
Title of Final Year Project	

Similarity	**Supervisor's Comments** **(Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index:________ % **Similarity by source** Internet Sources: _______________% Publications: __________ % Student Papers: __________ %	
Number of individual sources listed of more than 3% similarity: _________	
Parameters of originality required and limits approved by UTAR are as Follows: **(i) Overall similarity index is 20% and below, and** **(ii) Matching of individual sources listed must be less than 3% each, and** **(iii) Matching texts in continuous block must not exceed 8 words** ***Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.***	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

____
Signature of Supervisor

____
Signature of Co-Supervisor

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	
Student Name	
Supervisor Name	

TICK (√)	**DOCUMENT ITEMS** Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
	Front Cover
	Signed Report Status Declaration Form
	Title Page
	Signed form of the Declaration of Originality
	Acknowledgement
	Abstract
	Table of Contents
	List of Figures (if applicable)
	List of Tables (if applicable)
	List of Symbols (if applicable)
	List of Abbreviations (if applicable)
	Chapters / Content
	Bibliography (or References)
	All references in bibliography are cited in the thesis, especially in the chapter of literature review
	Appendices (if applicable)
	Poster
	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report. ________________________ (Signature of Student) Date:	Supervisor verification. Report with incorrect format can get 5 mark (1 grade) reduction. ________________________ (Signature of Supervisor) Date: